Applied Text Analytics for Blogs

Gilad Mishne

Applied Text Analytics for Blogs

Academisch Proefschrift

ter verkrijging van de graad van doctor aan de
Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof.dr. J.W. Zwemmer
ten overstaan van een door het college voor
promoties ingestelde commissie, in het openbaar
te verdedigen in de Aula der Universiteit
op vrijdag 27 april 2007, te 10.00 uur

door

Gilad Avraham Mishne

geboren te Haifa, Israël.

Promotor: Prof.dr. Maarten de Rijke

Committee:
Prof.dr. Ricardo Baeza-Yates
Dr. Natalie Glance
Prof.dr. Simon Jones
Dr. Maarten Marx

Faculteit der Natuurwetenschappen, Wiskunde en Informatica
Universiteit van Amsterdam

SIKS Dissertation Series No. 2007-06
The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

The investigations were supported by the Netherlands Organization for Scientific Research (NWO) under project number 220-80-001.

Printed and bound by PrintPartners Ipskamp, Enschede

ISBN-13: 978–90–5776–163–8

Contents

Acknowledgments

I am indebted to my advisor, Professor Maarten de Rijke, who opened the door of scientific research for me and guided my walk through a path that was at times unclear. Despite an incredibly busy schedule, Maarten always found time to discuss any issue, or to suggest final improvements to a paper; he helped me in many ways, professional and other, and gave me a much-appreciated free hand to explore new areas my way. Many thanks also to all members of the ILPS group at the University of Amsterdam for providing a great environment through many inspirational discussions and a diverse range of interests; special thanks goes to Maarten Marx, for introducing me to the group in the first place.

Research on text analysis depends on access to large amounts of real-life data; I am grateful to Natalie Glance from Blogpulse.com, Greg Gershman from Blogdigger.com, and Nils Rooijmans from Ilse Media BV for providing such data.

I wish to thank Susan Dumais and Doug Oard for coaching me at the SIGIR 2005 Doctoral Consortium, providing much needed context for the planned work.

I was fortunate to be hosted as an intern twice during the past years, working on interesting projects with great people. Thanks Yoelle Maarek, Aya Soffer and David Carmel for arranging my stay at the IBM Research Lab in Haifa during 2004, and Natalie Glance and Matthew Hurst for hosting me at the Intelliseek Applied Research Center (now Nielsen BuzzMetrics) in 2005.

I am grateful to the Netherlands Organization for Scientific Research (NWO) for providing the financial support for my work, and to Ricardo Baeza-Yates, Natalie Glance, Simon Jones, and Maarten Marx for serving on my PhD committee. Thanks also to Christel Lutz for help with Dutch translations.

My biggest thanks goes to my parents, Nava and David, who taught me (among many other things) the value of education, and to Lotta, for support in good and in bad.

Finally, this dissertation would not have been possible without the writings of millions of individuals worldwide documenting their thoughts, experiences, and commentary online. Thank you bloggers!

Gilad Mishne
Amsterdam, February 2007

Chapter 1

Introduction

The World Wide Web affects numerous aspects of our life, from the way we work to how we spend our free time and interact with others. Countless words have been written about the changes that came about with the popularity of the web, and it no longer seems required to state the web's importance, or the vast amount of information we can access through it.

In recent years, however, the content of the web is changing. Advances in technology and easy access to the internet from everywhere, anytime, combined with a generation that grew up with the web around it, and that is used to living online, have pushed forward a phenomenon known as *user-generated content*: web pages and content created and published by users of websites, rather than website owners. No longer is web content generated by professionals: everyone is a contributor. People use the web to publish and share photos, videos, audio, artwork, and, of course, various forms of written material. This is happening on a massive scale: according to a 2006 survey, more than a third of internet users create content; among young people, numbers are substantially higher [120]. As a result, large websites—from giants like the BBC and Amazon to small community pages—are changing too, focusing on offering facilities for everyone to add and share their content.

A particular type of user-generated content is the *blog*, a form of web page containing periodic updates by an author, and displaying these updates in a chronologically sorted order. Over the few years of their existence, blogs have evolved into public diaries of a sort, maintained by millions of individuals worldwide. Keeping a diary is nothing new by itself; but the recent blogging phenomenon is unique in that the diaries are publicly available, and distributed in a format easily processed by machines. The personal lives of millions are now accessible to all through an unmediated channel.

The Blogspace as a Corpus

From an information access point of view, the blogspace—the collection of all blogs—differs from other collections of documents on three levels: *content*, *structure*, and *timeline*. In terms of content, the text found in blogs is often of a personal nature, containing descriptions of a blogger's life and surroundings, as well as thoughts, emotions and commentary on various topics. This type of content is rare in other publicly available corpora; on a massive scale, it can only be found in blogs. The structure of blogs also differs from other domains: it is a dense social network of people, with an abundance of communities and a rapid flow of information. Finally, blogs are timelined: every blog post, comment, trackback and other entity in the blogspace has a detailed timestamp attached, often to the minute.

Of these three ways in which the blogspace differs from other collections of documents, this dissertation focuses mainly on the content of blogs, exploring ways in which this type of text can be used to offer bloggers and blog readers beneficial ways to access the information in the blogspace. In particular, we utilize various *text analytics* approaches for these tasks.

Text Analytics for Blogs

Text analytics is a broad term encompassing a set of methods for discovering knowledge in unstructured text. Data mining and machine learning techniques are combined with computational linguistics, information retrieval and information extraction, to identify concepts in text, determine relations between them, and support higher-level exploratory analyses of the data. For example, analytics techniques are used to locate domain experts in enterprise collections [57]; build knowledge taxonomies from collections of unstructured data [259]; or extract genes and relations between genes from biomedical information [148].

Text analytics—also referred to as text mining, knowledge discovery in text, intelligent information access, and a range of other terms—attempts to address the *information overload* problem. As technology advances, enabling creation and storage of more and more data, the amount of text and other information we are surrounded by reaches new levels: we are no longer able to digest all material we have access to, and need mechanisms for searching, categorizing, summarizing—and simply understanding—these large amounts of data. Search engines, which became popular on the web as it grew in size, and more recently are becoming the norm also for access to personal information archives such as the local filesystem of personal computers, are only starting to address the problem. Search has an important role in identifying specific information sources in a large collection, but falls short of extracting actual knowledge from the retrieved information. More importantly, search does not mine and combine information from multiple sources, but focuses instead on delivering single entities at a time.

Within this analytics framework, we apply a range of methods to mine information from blogs; we use both known text analysis methods and novel ones to do this. In some cases, we aim to extract knowledge which is not particular to blogs from this new domain; examples are blog categories (or tags), or searcher behavior. In other cases, we define new tasks which can benefit from text analytics: e.g., tasks related to the sentiment expressed in blogs, or to particular types of spam that exist in the blogspace. We refer to the work presented here as *applied* text analytics, as every algorithm proposed is put to the test in a real-life setting, with a concrete task, real data, and (where possible), compared to the state-of-the-art.

1.1 Research Questions

The main question guiding this thesis is this: *how can the characteristics of blogs be utilized to effectively mine useful knowledge from the blogspace?* More specifically, we aim to address three questions derived from this main one.

1. What types of information can be mined from the personal, informal content found in blogs? How can text analysis be used to extract this knowledge, both at the individual blog level and at the aggregate level?

We have already mentioned that the content of blogs differs from that of other corpora used for text analysis: as a collection of public diaries, it is rich in personal, unedited commentary, opinions, and thoughts. This type of content suggests new types of knowledge that can be accessed through blogs, and new tasks that text analytics may help to address, such as identifying the opinions expressed, classifying and summarizing them. One goal we set ourselves is to identify the types of information that can be mined from single blogs or from collections of blogs, and show how text analysis can be used to extract it.

2. How do existing text analytics methods perform when applied to blog content? How can known approaches benefit from properties of blogs?

While blog content is substantially different from the content of other types of corpora, there are tasks which are shared between this and other domains, e.g., topical categorization of content. For this type of tasks, blogs are both a challenge and an opportunity. On one hand, blog content may prove complicated for these tasks, which are often sensitive to unfocused or ungrammatical text such as that found in blogs. On the other hand, some aspects of the blogspace—for example, the dense, community-like structure of it—may be used to enhance existing approaches. In a number of cases, we will address a traditional information-seeking task in blogs; we will evaluate the effectiveness of existing text analysis methods for it, as well as use the properties of the blogspace to improve results or enrich the task otherwise.

3. How does blog search differ from web search? What are the differences and similarities in the types of information needs and the user behavior? What factors contribute to the effectiveness of blog retrieval, and, in particular, to the performance on those search tasks which are characteristic of the blogspace?

The sharp increase in the amount of information available online has positioned search as a prominent way in which people access this information; and as the volume of the blogspace grows, search is becoming increasingly important in this domain too. For this reason, we dedicate a separate part of this thesis to explore blog search—an area which, at the time this work was done, was largely unchartered. First, we identify the similarities and differences between search in the blogspace and other types of web search, both in terms of user needs and user behavior. Then, we address those needs which are special to blogs, examine how known retrieval approaches perform on it, and develop additional ones tailored to the task and the data.

Methodology

As described earlier, the general term "text analytics" refers to a collection of many techniques and approaches, and we utilize a number of them to answer the questions we have just listed. In particular, we focus on two methods of analytics in this thesis: *statistical language models* and *text classification*.

Language models, which are described in detail in Chapter 3, are statistical models that capture properties of language usage, expressing them with probabilities; in their simplest form, they are a probability distributions over strings, listing the likelihood of any term to appear in a language. Language models can be used to decide how likely it is that a given segment of text belongs to some language, or how strongly related two different texts are. The language analyzed can be the language used in a large domain (e.g., all blogs), but also the language used in a very specific context, such as a single entry in a blog; in this dissertation, we use both language types, to address tasks such as profiling the bloggers, or searching the blog content.

Text classification, or text categorization, is aimed at assigning one or more categories—usually predefined ones—to a given text, to facilitate meaningful distinction between types of text. We model many of the tasks presented in this thesis as text classification tasks, and use a supervised machine learning approach to address them. In the supervised framework, an automated learner is provided with categorized training examples and a set of features which can be extracted from each; the learner then uses the features to construct a model of the different classes. We concentrate mainly on the types of features that can be identified within the text of blogs for this process, and their relative contribution to the learning process.

1.2 Organization of the Thesis

This thesis is organized in three parts, each focused on a different set of information access tasks for blogs. Before these parts begin, however, a background chapter—Chapter 2—introduces blogs and the blogspace, and provides context for the rest of the thesis. In addition to defining the terminology used later and providing an overview of related work, it includes some novel analysis of the properties of the blogspace, particularly those related to content, and, to a lesser extent, to link structure.

The three parts following this background chapter are as follows.

- Part I begins the analysis of blog contents at the level of single blogs, and, sometimes, single blog posts, identifying information that can be mined at the blog level and evaluating methods for the extraction. The Part consists of three chapters: Chapter 3 takes a closer look at the blogger, attempting to learn information about her from her blog: for example, we are interested in what products a blogger or her readers are likely to appreciate. This is followed by Chapter 4 which applies traditional text classification methods to extract new types of information from blogs: both content-oriented information such as tags, as well as information about the style and sentiment expressed in the blog . Finally, Chapter 5 tackles one form of spam which occurs mostly in blogs—comment spam—by analyzing the content of the blog.

- Part II moves from the single blog to multiple blogs, offering methods for extracting aggregated knowledge; some of these are extensions to methods used in Part I, while others are new altogether. In particular, the part contains two chapters. Chapter 6 focuses on the high level of sentiment often found in blog posts, showing that mining the combined level of this sentiment is useful, and—building on work from Chapter 4—introduces new tasks in this domain. Chapter 7 analyzes a form of online communication which is particular to blogs—commenting—identifying types of knowledge which can be found in this domain.

- Part III focuses on blog search; as we mentioned, we view the importance of search as a method for accessing information on the web as justifying a separate part of the thesis dedicated to this domain. The part includes two chapters: Chapter 8 studies search behavior in the blogspace, comparing it to search behavior in other web areas; Chapter 9 presents and analyzes ways to apply text analysis for addressing a search task which is unique to the blogspace: locating opinionated content, or people's thoughts and comments, in blog posts.

Chapter 10 concludes this thesis, listing its contributions and suggesting future work which builds on the work presented here.

Finally, the nature of the work presented in this thesis—analytics of large-scale corpora—call for a substantial investment in engineering and system development. Some of the tools that have been developed for performing the experiments described in later chapters are described in more details in two appendices.

Although a single story is told in this thesis, the three parts of it can be read separately; in particular, the last part, which discusses search in the blogspace, does not depend on knowledge introduced in the first two.

1.3 Foundations

This work builds on intermediate results made public over the past years. Early versions of some of the work presented in Part I were published as

- "Experiments with Mood Classification in Blog Posts" [194],
- "AutoTag: A Collaborative Approach to Automated Tag Assignment for Weblog Posts" [195],
- "Blocking Blog Spam with Language Model Disagreement" [199],
- "Deriving Wishlists from Blogs" [202], and
- "Language Model Mixtures for Contextual Ad Placement in Personal Blogs" [203].

Part II builds on work presented in

- "Predicting Movie Sales from Blogger Sentiment" [208],
- "Capturing Global Mood Levels using Blog Posts" [201],
- "Why Are They Excited? Identifying and Explaining Spikes in Blog Mood Levels" [23], and
- "Leave a Reply: An Analysis of Weblog Comments" [207].

Part of the material appearing in Part III was published in

- "A Study of Blog Search" [205],
- "Boosting Web Retrieval through Query Operations" [200],
- "Multiple Ranking Strategies for Opinion Retrieval in Blogs" [197], and
- "Using Blog Properties to Improve Retrieval" [198].

Some of the ideas discussed throughout the thesis appeared in "Information Access Challenges in the Blogspace" [196]. Appendix B describes the internal workings of the system presented in "MoodViews: Tools for Blog Mood Analysis" [204] and made public at www.moodviews.com since mid-2005.

Chapter 2

Background: Blogs and the Blogspace

This chapter introduces blogs and the blogspace, defining the terminology used in later chapters. Additionally, it surveys blog-related research, grouping it into high-level categories.

The unique properties of blogs attract researchers from many schools. In particular, a large body of blog research deals with anthropological and ethnographic issues; other studies are journalism-oriented. As the scope of this thesis is computational access to information in blogs, we leave out most of this non-computational research, briefly touching on it when needed to clarify an issue; due to this restriction, we skip some excellent ethnographic work such as that of Nardi et al. [218], as well as research concerning privacy implications of the public exposure of individuals—particularly young ones—through their blogs, e.g. [301].

We begin by introducing blogs, reviewing their structure and format, and providng the historical context for the blogging phenomenon. This is followed by a description of the *blogspace*, the collection of all blogs: its internal structure, development, and demographics. Finally, we survey major works in computational analysis of blogs, grouped into different domains. Throughout this chapter, we will emphasize the unique characteristics of the blogspace as a corpus: the content type, structure, and temporal aspects.

2.1 Blogs

2.1.1 What is a Blog?

Coming up with a formal definition of a blog is an elusive task; indeed, understanding what it means was so sought-after that the word "blog" became the most-looked-up-word in the Merriam-Webster dictionary during 2004 [190]. The definition given by Merriam-Webster was "a web site that contains an online personal journal with reflections, comments, and often hyperlinks provided by the writer"—but it is only one of several possible definitions.

Jorn Barger first coined the term "weblog" in 1997 for his personal site, Robot Wisdom;[1] the original term referred to a "web page where a weblogger 'logs' all the other webpages she finds interesting" [27]. This definition was used in the early stages of blogging, during the late 1990s, when many of them indeed functioned as logs of pages the authors visited on the web. A short form, "blog," was proposed in 1999, and soon became more popular than the original term.

As the blogging phenomenon developed, so did its definition. The lists of relatively short descriptions of web pages found in early blogs were replaced by longer comments and reflections. Blog entries sometimes did not directly relate to a particular web page, focusing instead on current events or other topics, discussing them from a personal perspective. Towards the end of the century, this new style of blogs rapidly outnumbered the early, log-like ones [38]. These personal-oriented blogs were frequently updated, and contained entries which reflected the writer's thoughts and feelings at the time of writing, resembling online diaries. The authors of such blogs—the *bloggers*—used them both for personal record keeping and documentation (as with traditional diaries), and as a way of sharing their experiences and thoughts with friends and with others; while the vast majority maintained their blog as a hobby [165], some went as far as defining themselves through their blog [265]. The emergence of personal blogs was a gradual process, but it was particularly shaped and accelerated by the development of specialized platforms and software to assist bloggers such as Blogger[2] and Movable Type;[3] these platforms substantially reduced the amount of technical knowledge required by a blogger and brought the ease of blog publishing to the masses.

Some argue that blogs do not differ substantially from previous forms of self-publishing, other than an increased social aspect [184, 112]—but most blog studies address them as a separate medium. A commonly used definition for blogs is that given by Walker: a "frequently updated website consisting of dated entries arranged in reverse chronological order so the most recent post appears first" [304]. Walker, and others (e.g., Gill [85], Winer [313]), refer to an array of additional features shared by many blogs; some of these are so tightly associated with blogs that they are occasionally used do define them, or to decide whether a web site is indeed a blog. The features are summarized in Table 2.1.

But while many blogs share similar format and layout, the content type, style, and goals may differ substantially. A less technical approach to defining a blog is taken by Winer [313]. While he also lists characteristic features of blogs as well as technical implementation aspects, his definition is centered on the person behind the blog. A blog is, according to Winer, a web page voicing the unedited, unregulated voice of an individual; its structure and format are just means for

[1] http://robotwisdom.com

[2] http://blogger.com

[3] http://www.sixapart.com/movabletype/

Basic Features
• Web page containing time-stamped entries
• Entries are sorted in reverse chronological order, latest one first
Additional Features: Content/Style
• Authored by individuals, or—in a minority of the cases—by small communities
• Written in an informal, unedited, personal style
• Containing mostly text, although other multimedia formats exist
• Updated regularly
Additional Features: Structure/Format
• Published using a relatively simple, template-based platform; bloggers need not know the specifics of web authoring
• Containing archives of content, organized chronologically and/or thematically
• Enabling content syndication
• Enabling reader feedback
• Linking to additional, related blogs

Table 2.1: Defining features of blogs.

expressing the individual's views in an easier, more appealing, more interactive manner.[4]

Finally, a different approach to defining a blog is taken by Halavais [104]: according to this view, neither the contents nor the technical aspects of blogs are their defining characteristics, but rather the way they link to other blogs. In his words, "the only seemingly vital element of weblogging is a public forum (the World Wide Web) in which bloggers are able to associate and self-assemble into groups. The attraction to weblogging has less to do with the software involved and more to do with the kinds of social groups that emerge from interactions among weblogs and their authors."

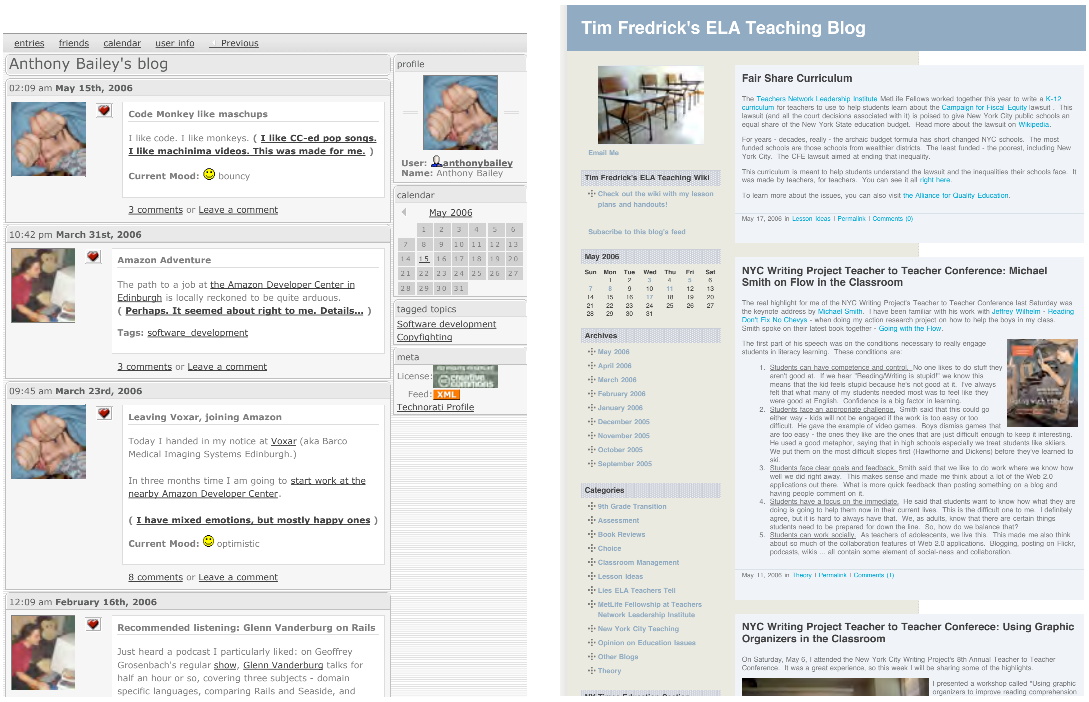
In this work, we adopt Winer's definition of a blog: a combination of a technical description of their building blocks with the notion of an underlying "voice of an individual."

[4] Winer does not exclude group blogs—blogs authored by more than one person—from his definition; examining one borderline case, he concludes that the important feature is that "the personalities of the writers come through." [313]

2.1.2 Terminology

We now turn to more concrete definitions and examples of the components and features of blogs and blogging. Rather than using a single blog as a running example and analyzing its components, different blogs are used as examples for different features. This way, different blogging styles and practices are demonstrated: blogs come in a variety of forms, and a diverse sample is required to develop an intuition of the common characteristics.[5]

Blog Post. A blog is made up of individual entries, or articles, called "posts." A post typically includes a title, a date, and contents (or body); usually, the author's name or another signature is provided too. Most blogs display, in their main page, the recently added posts, sorted in reverse chronological order (i.e., the latest post is shown at the top of the page). Examples of front pages of blogs containing a number of posts appear in Figure 2.1.



Anthony Bailey's blog

02:09 am **May 15th, 2006**

Code Monkey like maschups

I like code. I like monkeys. (**I like CC-ed pop songs. I like machinima videos. This was made for me.**)

Current Mood: bouncy

3 comments or Leave a comment

10:42 pm **March 31st, 2006**

Amazon Adventure

The path to a job at the Amazon Developer Center in Edinburgh is locally reckoned to be quite arduous.
(**Perhaps. It seemed about right to me. Details...**)

Tags: software_development

3 comments or Leave a comment

09:45 am **March 23rd, 2006**

Leaving Voxar, joining Amazon

Today I handed in my notice at Voxar (aka Barco Medical Imaging Systems Edinburgh.)

In three months time I am going to start work at the nearby Amazon Developer Center.

(**I have mixed emotions, but mostly happy ones**)

Current Mood: optimistic

8 comments or Leave a comment

12:09 am **February 16th, 2006**

Recommended listening: Glenn Vanderburg on Rails

Just heard a podcast I particularly liked: on Geoffrey Grosenbach's regular show, Glenn Vanderburg talks for half an hour or so, covering three subjects - domain specific languages, comparing Rails and Seaside, and

User: anthonybailey
Name: Anthony Bailey

May 2006

	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Software development
Copyfighting

Technorati Profile

Tim Fredrick's ELA Teaching Blog

Email Me

Tim Fredrick's ELA Teaching Wiki

Check out the wiki with my lesson plans and handouts!

Subscribe to this blog's feed

May 2006

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Archives

- May 2006
- April 2006
- March 2006
- February 2006
- January 2006
- December 2005
- November 2005
- October 2005
- September 2005

Categories

- 9th Grade Transition
- Assessment
- Book Reviews
- Choice
- Classroom Management
- Lesson Ideas
- Lies ELA Teachers Tell
- MetLife Fellowship at Teachers Network Leadership Institute
- New York City Teaching
- Opinion on Education Issues
- Other Blogs
- Theory

Fair Share Curriculum

The Teachers Network Leadership Institute MetLife Fellows worked together this year to write a K-12 curriculum for teachers to use to help students learn about the Campaign for Fiscal Equity lawsuit . This lawsuit (and all the court decisions associated with it) is poised to give New York City public schools an equal share of the New York State education budget. Read more about the lawsuit on Wikipedia.

For years - decades, really - the archaic budget formula has short changed NYC schools. The most funded schools are those schools from wealthier districts. The least funded - the poorest, including New York City. The CFE lawsuit aimed at ending that inequality.

This curriculum is meant to help students understand the lawsuit and the inequalities their schools face. It was made by teachers, for teachers. You can see it all right here.

To learn more about the issues, you can also visit the Alliance for Quality Education.

May 17, 2006 in Lesson Ideas | Permalink | Comments (0)

NYC Writing Project Teacher to Teacher Conference: Michael Smith on Flow in the Classroom

The real highlight for me of the NYC Writing Project's Teacher to Teacher Conference last Saturday was the keynote address by Michael Smith. I have been familiar with his work with Jeffrey Wilhelm - Reading Don't Fix No Chevys - when doing my action research project on how to help the boys in my class. Smith spoke on their latest book together - Going with the Flow.

The first part of his speech was on the conditions necessary to really engage students in literacy learning. These conditions are:

1. Students can have competence and control. No one likes to do stuff they aren't good at. If we hear "Reading/Writing is stupid!" we know this means that the kid feels stupid because he's not good at it. I've always felt that what many of my students needed most was to feel like they were good at English. Confidence is a big factor in learning.
2. Students face an appropriate challenge. Smith said that this could go either way - kids will not be engaged if the work is too easy or too difficult. He gave the example of video games. Boys dismiss games that are too easy - the ones they like are the ones that are just difficult enough to keep it interesting. He used a good metaphor, saying that in high schools especially we treat students like skiiers. We put them on the most difficult slopes first (Hawthorne and Dickens) before they've learned to ski.
3. Students face clear goals and feedback. Smith said that we like to do work where we know how well we did right away. This makes sense and made me think about a lot of the Web 2.0 applications out there. What is more quick feedback than posting something on a blog and having people comment on it.
4. Students have a focus on the immediate. He said that students want to know how what they are doing is going to help them now in their current lives. This is the difficult one to me. I definitely agree, but it is hard to always have that. We, as adults, know that there are certain things students need to be prepared for down the line. So, how do we balance that?
5. Students can work socially. As teachers of adolescents, we live this. This made me also think about so much of the collaboration features of Web 2.0 applications. Blogging, posting on Flickr, podcasts, wikis ... all contain some element of social-ness and collaboration.

May 11, 2006 in Theory | Permalink | Comments (1)

NYC Writing Project Teacher to Teacher Conferece: Using Graphic Organizers in the Classroom

On Saturday, May 6, I attended the New York City Writing Project's 8th Annual Teacher to Teacher Conference. It was a great experience, so this week I will be sharing some of the highlights.

I presented a workshop called "Using graphic organizers to improve reading comprehension

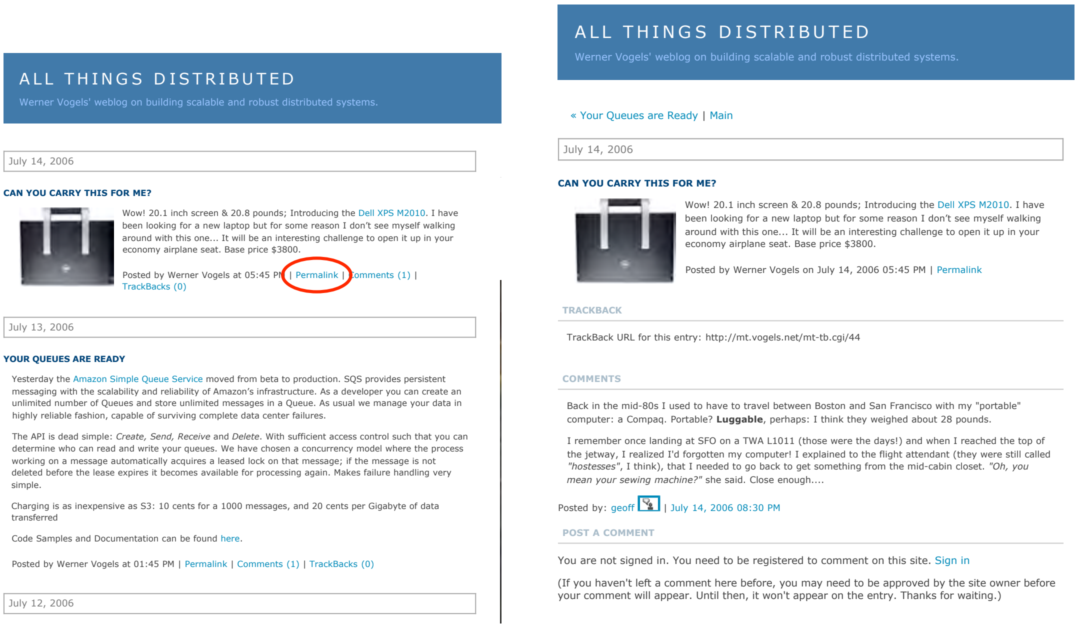
Figure 2.1: Sample blog posts on blog front pages. Left: http://anthonybailey.livejournal.com/; Right: http://timfredrick.typepad.com/.

Permalink. Permalinks, short for permanent links, are URLs referring to a specific blog post within a blog. The pages reached by these URLs are, at least

[5] The examples used here are taken from the subset of blogs licensed under a Creative Commons license.

theoretically, guaranteed to be “permanent”—i.e., the same post is always reachable given its permalink (while the contents of the blog page itself change over time, as entries are added). Permalinks first appeared in 2000, as the amount of blog content increased, and a mechanism for archiving and navigating older content was required [55]; currently, they are a standard feature in virtually all blogs. Sometimes, the full content of a blog post is shown only at its permalink page, and the blog front page contains just a summary of it.

Figure 2.2 highlights the permanent link indicator next to a blog post in a blog home page, and the blog post page when the permalink is followed. Note that the permalink page contains additional information not displayed on the front page (e.g., the comments appearing after the post).



ALL THINGS DISTRIBUTED

Werner Vogels' weblog on building scalable and robust distributed systems.

July 14, 2006

CAN YOU CARRY THIS FOR ME?

Wow! 20.1 inch screen & 20.8 pounds; Introducing the Dell XPS M2010. I have been looking for a new laptop but for some reason I don’t see myself walking around with this one... It will be an interesting challenge to open it up in your economy airplane seat. Base price $3800.

Posted by Werner Vogels at 05:45 PM | Permalink | Comments (1) | TrackBacks (0)

July 13, 2006

YOUR QUEUES ARE READY

Yesterday the Amazon Simple Queue Service moved from beta to production. SQS provides persistent messaging with the scalability and reliability of Amazon’s infrastructure. As a developer you can create an unlimited number of Queues and store unlimited messages in a Queue. As usual we manage your data in highly reliable fashion, capable of surviving complete data center failures.

The API is dead simple: *Create, Send, Receive* and *Delete*. With sufficient access control such that you can determine who can read and write your queues. We have chosen a concurrency model where the process working on a message automatically acquires a leased lock on that message; if the message is not deleted before the lease expires it becomes available for processing again. Makes failure handling very simple.

Charging is as inexpensive as S3: 10 cents for a 1000 messages, and 20 cents per Gigabyte of data transferred

Code Samples and Documentation can be found here.

Posted by Werner Vogels at 01:45 PM | Permalink | Comments (1) | TrackBacks (0)

July 12, 2006

ALL THINGS DISTRIBUTED

Werner Vogels' weblog on building scalable and robust distributed systems.

« Your Queues are Ready | Main

July 14, 2006

CAN YOU CARRY THIS FOR ME?

Wow! 20.1 inch screen & 20.8 pounds; Introducing the Dell XPS M2010. I have been looking for a new laptop but for some reason I don’t see myself walking around with this one... It will be an interesting challenge to open it up in your economy airplane seat. Base price $3800.

Posted by Werner Vogels on July 14, 2006 05:45 PM | Permalink

TRACKBACK

TrackBack URL for this entry: http://mt.vogels.net/mt-tb.cgi/44

COMMENTS

Back in the mid-80s I used to have to travel between Boston and San Francisco with my "portable" computer: a Compaq. Portable? **Luggable**, perhaps: I think they weighed about 28 pounds.

I remember once landing at SFO on a TWA L1011 (those were the days!) and when I reached the top of the jetway, I realized I'd forgotten my computer! I explained to the flight attendant (they were still called *"hostesses"*, I think), that I needed to go back to get something from the mid-cabin closet. *"Oh, you mean your sewing machine?"* she said. Close enough....

Posted by: geoff | July 14, 2006 08:30 PM

POST A COMMENT

You are not signed in. You need to be registered to comment on this site. Sign in

(If you haven't left a comment here before, you may need to be approved by the site owner before your comment will appear. Until then, it won't appear on the entry. Thanks for waiting.)

Figure 2.2: Left: a permalink (highlighted) appearing at the end of a post, on the main page of a blog. Right: the post as displayed when the permalink is followed. Source: http://allthingsdistributed.com.

Archive/Calendar. Being a timeline-oriented medium, most blogs offer a mechanism for browsing the blog content by date: an archive, often visualized as a calendar. A random sample of blogs analyzed in mid-2003 shows 74% of them had archives [112]—and this figure has most likely increased since then, with the enhancements in blogging platforms. Examples of calendar-like archives appear in the blog posts in Figure 2.1.

Comments. Many blogging platforms allow readers to react to a blog post by writing a comment, which is then displayed following the post itself. This two-sided communication mechanism is one of the properties distinguishing blog content from other web content. Like blog posts themselves, comments usually appear in chronological order (but, unlike the posts, the ordering is not reversed—the most recent comment appears last); in some platforms, they are grouped in conversational threads, in a style resembling online discussion boards. To prevent abuse, some commenting mechanisms require some sort of user authorization, or are moderated by the blogger; all in all, in mid-2006 the vast majority of bloggers (87%) allowed comments on their blog [165]. A sample comment appears in Figure 2.2 (right side).

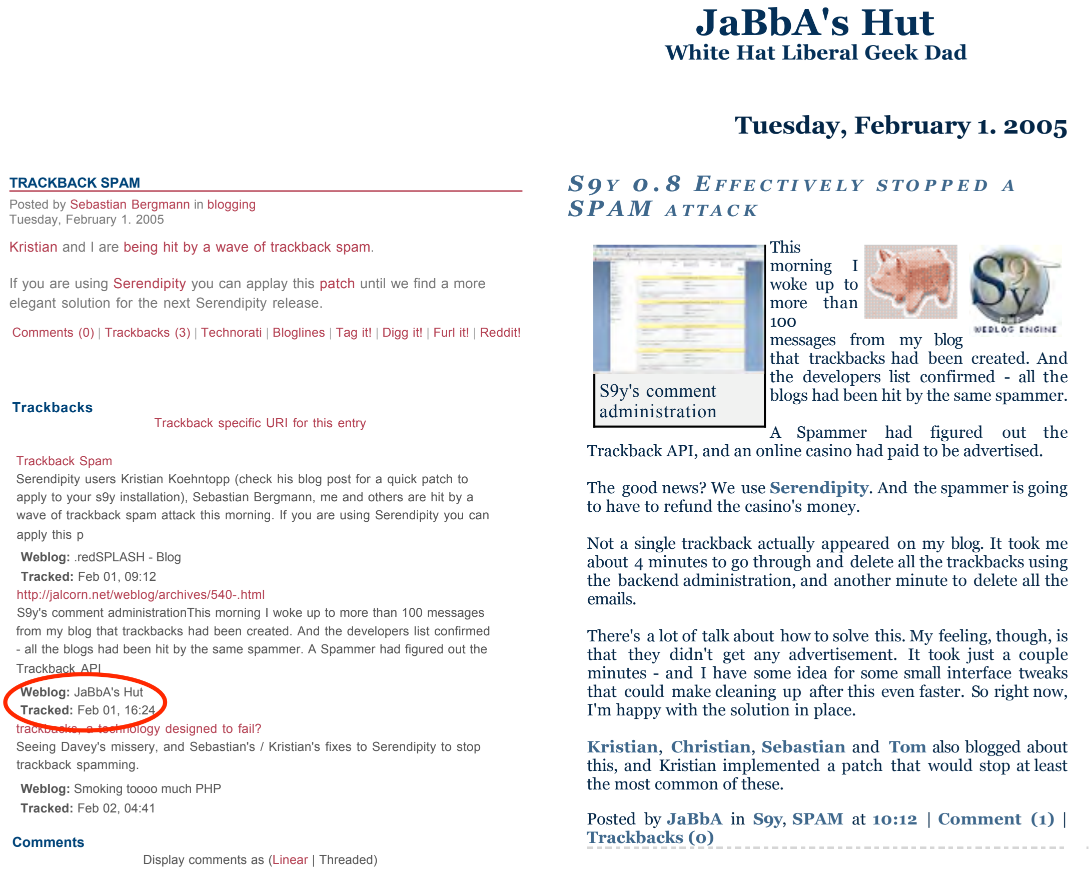
Trackbacks. Linkbacks are a family of protocols that are used to notify a web site about a page linking to it; the trackback protocol is the most common of them, and is often used to refer to the technique as a whole. The main usage of trackbacks in the blogspace is linking related blog posts. A blogger referencing a post in another blog can use the trackback mechanisms to notify the referenced blog; the referenced post will then usually include a short excerpt from the reference in the original blogs and link back to it. These excerpts, also referred to as "trackbacks," typically appear at the end of the post, with the comments to the post.

Figure 2.3 shows a blog post having several trackbacks, and one of the posts that refers to it through the trackback mechanism.

Web feeds and Syndication. Web feeds are documents consisting of structured, usually timestamped, items; they are often used to distribute content in a format which is easy for computers to parse. Feeds are almost always XML-formatted; two specific XML formats called RSS (RDF Site Summary or Really Simple Syndication) and Atom are currently used for the vast majority of feeds.[6] Syndication, in the context of blogs, is the process of distributing blog content in standardized format through feeds. Most blog authoring tools support various syndication options. Blog readers can choose whether they access the blog through its HTML interface, or read the contents of the web feed. In the latter case, readers use specialized software to view the feed (possibly, together with other feeds of interest—hence the name "aggregators" sometimes used to refer to this software). Typically, readers of a blog who choose to view its content through its web feed "subscribe" to the blog's feed, meaning that their software regularly checks for feed updates. The usefulness of syndication and its popularity with internet users led online newspapers and other dynamic web sites to adopt it [86], and it is currently prevalent in many non-blog pages.

Links on the HTML page of a blog referring the reader to feeds of the blog are shown in Figure 2.4 (top); below it is an example of the syndicated content

[6] For an overview of the history and specifications of RSS and Atom see http://en.wikipedia.org/wiki/RSS and http://en.wikipedia.org/wiki/Atom_(standard)



TRACKBACK SPAM

Posted by Sebastian Bergmann in blogging
Tuesday, February 1. 2005

Kristian and I are being hit by a wave of trackback spam.

If you are using Serendipity you can applay this patch until we find a more elegant solution for the next Serendipity release.

Comments (0) | Trackbacks (3) | Technorati | Bloglines | Tag it! | Digg it! | Furl it! | Reddit!

Trackbacks

Trackback specific URI for this entry

Trackback Spam
Serendipity users Kristian Koehntopp (check his blog post for a quick patch to apply to your s9y installation), Sebastian Bergmann, me and others are hit by a wave of trackback spam attack this morning. If you are using Serendipity you can apply this p

Weblog: .redSPLASH - Blog
Tracked: Feb 01, 09:12

http://jalcorn.net/weblog/archives/540-.html
S9y's comment administrationThis morning I woke up to more than 100 messages from my blog that trackbacks had been created. And the developers list confirmed - all the blogs had been hit by the same spammer. A Spammer had figured out the Trackback API

Weblog: JaBbA's Hut
Tracked: Feb 01, 16:24

trackbacks, a technology designed to fail?
Seeing Davey's missery, and Sebastian's / Kristian's fixes to Serendipity to stop trackback spamming.

Weblog: Smoking toooo much PHP
Tracked: Feb 02, 04:41

Comments

Display comments as (Linear | Threaded)

JaBbA's Hut
White Hat Liberal Geek Dad

Tuesday, February 1. 2005

S9y 0.8 Effectively stopped a SPAM attack

S9y's comment administration

This morning I woke up to more than 100 messages from my blog that trackbacks had been created. And the developers list confirmed - all the blogs had been hit by the same spammer.

A Spammer had figured out the Trackback API, and an online casino had paid to be advertised.

The good news? We use Serendipity. And the spammer is going to have to refund the casino's money.

Not a single trackback actually appeared on my blog. It took me about 4 minutes to go through and delete all the trackbacks using the backend administration, and another minute to delete all the emails.

There's a lot of talk about how to solve this. My feeling, though, is that they didn't get any advertisement. It took just a couple minutes - and I have some idea for some small interface tweaks that could make cleaning up after this even faster. So right now, I'm happy with the solution in place.

Kristian, Christian, Sebastian and Tom also blogged about this, and Kristian implemented a patch that would stop at least the most common of these.

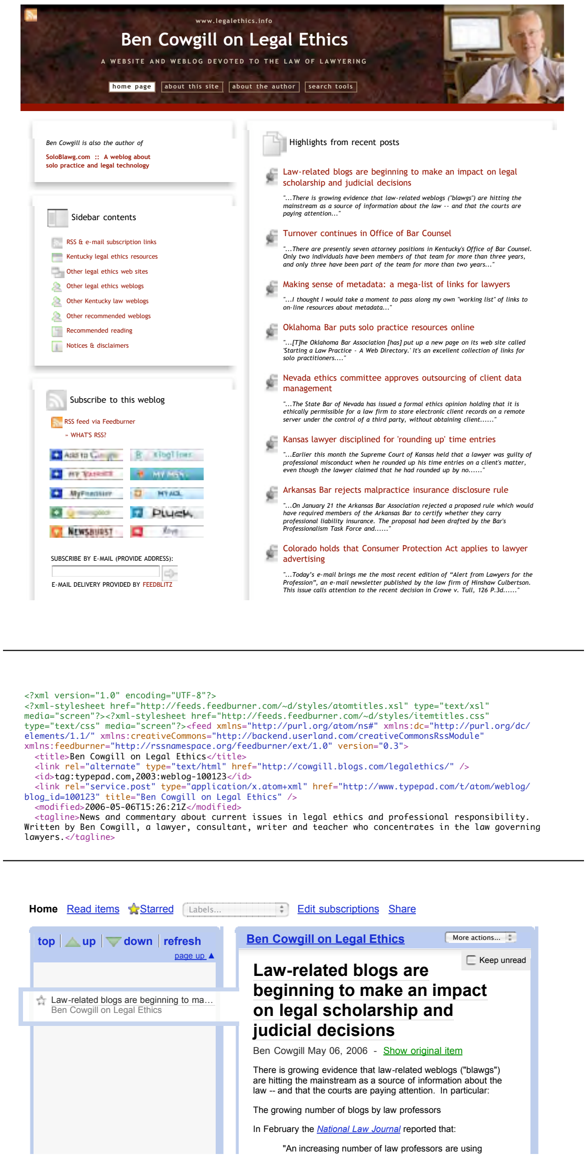
Posted by JaBbA in S9y, SPAM at 10:12 | Comment (1) | Trackbacks (0)

Figure 2.3: Left: a blog post with several trackbacks, displaying summaries of the linking posts. Source: http://sebastian-bergmann.de/blog. Right: one of the posts linking to the post on the left (the highlighted trackback). Source: http://jalcorn.net/weblog.

distributed and displayed in a feed aggregator (in this case—Google Reader).

Ping Server. When a blogger has completed writing a post (or updating an existing one), she is usually interested in notifying the blog's current and potential readers about the update. Many blogging tools offer a standardized, computer-readable form of such a notification, called a "ping." Ping servers are services set up to receive such pings, possibly redistributing them to interested parties—for example, blog search engines that want to index the new post, or subscribers of the given blog who are interested in knowing about new content posted to it.

Figure 2.5 shows example pings as redistributed from a public ping server, blo.gs.



Ben Cowgill on Legal Ethics

A WEBSITE AND WEBLOG DEVOTED TO THE LAW OF LAWYERING

Ben Cowgill is also the author of

SoloBlawg.com :: A weblog about solo practice and legal technology

Sidebar contents

Subscribe to this weblog

SUBSCRIBE BY E-MAIL (PROVIDE ADDRESS):

E-MAIL DELIVERY PROVIDED BY FEEDBLITZ

Highlights from recent posts

Law-related blogs are beginning to make an impact on legal scholarship and judicial decisions

"...There is growing evidence that law-related weblogs ("blawgs") are hitting the mainstream as a source of information about the law -- and that the courts are paying attention..."

Turnover continues in Office of Bar Counsel

"...There are presently seven attorney positions in Kentucky's Office of Bar Counsel. Only two individuals have been members of that team for more than three years, and only three have been part of the team for more than two years..."

Making sense of metadata: a mega-list of links for lawyers

"...I thought I would take a moment to pass along my own "working list" of links to on-line resources about metadata..."

Oklahoma Bar puts solo practice resources online

"...[T]he Oklahoma Bar Association [has] put up a new page on its web site called 'Starting a Law Practice - A Web Directory.' It's an excellent collection of links for solo practitioners...."

Nevada ethics committee approves outsourcing of client data management

"...The State Bar of Nevada has issued a formal ethics opinion holding that it is ethically permissible for a law firm to store electronic client records on a remote server under the control of a third party, without obtaining client......"

Kansas lawyer disciplined for 'rounding up' time entries

"...Earlier this month the Supreme Court of Kansas held that a lawyer was guilty of professional misconduct when he rounded up his time entries on a client's matter, even though the lawyer claimed that he had rounded up by no......"

Arkansas Bar rejects malpractice insurance disclosure rule

"...On January 21 the Arkansas Bar Association rejected a proposed rule which would have required members of the Arkansas Bar to certify whether they carry professional liability insurance. The proposal had been drafted by the Bar's Professionalism Task Force and......"

Colorado holds that Consumer Protection Act applies to lawyer advertising

"...Today's e-mail brings me the most recent edition of "Alert from Lawyers for the Profession", an e-mail newsletter published by the law firm of Hinshaw Culbertson. This issue calls attention to the recent decision in Crowe v. Tull, 126 P.3d......"

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet href="http://feeds.feedburner.com/~d/styles/atomtitles.xsl" type="text/xsl"
media="screen"?><?xml-stylesheet href="http://feeds.feedburner.com/~d/styles/itemtitles.css"
type="text/css" media="screen"?><feed xmlns="http://purl.org/atom/ns#" xmlns:dc="http://purl.org/dc/
elements/1.1/" xmlns:creativeCommons="http://backend.userland.com/creativeCommonsRssModule"
xmlns:feedburner="http://rssnamespace.org/feedburner/ext/1.0" version="0.3">
  <title>Ben Cowgill on Legal Ethics</title>
  <link rel="alternate" type="text/html" href="http://cowgill.blogs.com/legalethics/" />
  <id>tag:typepad.com,2003:weblog-100123</id>
  <link rel="service.post" type="application/x.atom+xml" href="http://www.typepad.com/t/atom/weblog/
blog_id=100123" title="Ben Cowgill on Legal Ethics" />
  <modified>2006-05-06T15:26:21Z</modified>
  <tagline>News and commentary about current issues in legal ethics and professional responsibility.
Written by Ben Cowgill, a lawyer, consultant, writer and teacher who concentrates in the law governing
lawyers.</tagline>
```

Ben Cowgill on Legal Ethics

Law-related blogs are beginning to ma…
Ben Cowgill on Legal Ethics

Law-related blogs are beginning to make an impact on legal scholarship and judicial decisions

Ben Cowgill May 06, 2006 - Show original item

There is growing evidence that law-related weblogs ("blawgs") are hitting the mainstream as a source of information about the law -- and that the courts are paying attention. In particular:

The growing number of blogs by law professors

In February the *National Law Journal* reported that:

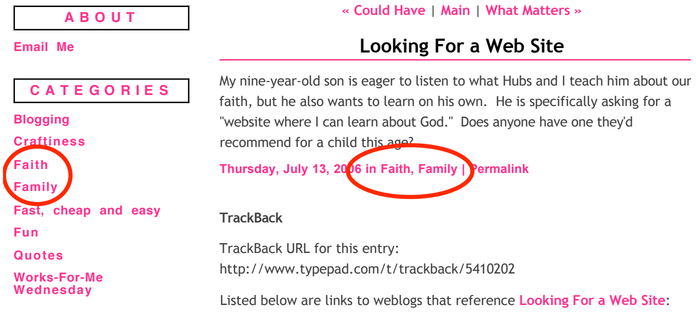
"An increasing number of law professors are using

Figure 2.4: The syndication mechanism: a link to a blog's feed from the blog HTML page (top); the syndicated content, displayed as raw XML (middle) and in a feed reader (bottom). Source: http://cowgill.blogs.com.

```
<?xml version="1.0" encoding="utf-8"?>
<weblogUpdates version="1" time="20070225T10:39:33Z">
<weblog name="Revival.com NewsFeed"
        url="http://www.revival.com/revivalrss/revivalrss.aspx"
        service="yahoo.com"
        ts="20070225T10:39:40Z" />
<weblog name="Save me from a villainous imagination"
        url="http://annaonthemoon.livejournal.com/data/rss"
        service="yahoo.com"
        ts="20070225T10:40:01Z" />
<weblog name="Whoot!"
        url="http://feeds.feedburner.com/typepad/jamesaduncan/whoot"
        service="yahoo.com"
        ts="20070225T10:40:04Z" />
. . .
```

Figure 2.5: Sample pings from a ping server.

Tags. Tags are simply textual labels assigned to posts to facilitate organization and navigation of the blog's content by topic. Most blogging tools allow tagging in various forms, and in most cases there are no restrictions on the number or content of tags used. Tags appear in Figure 2.6, both at the end of a blog post and as a navigational tool on the sidebar; as of mid-2006, tags were used in about half of all blog posts [67].



ABOUT

Email Me

CATEGORIES

Blogging
Craftiness
Faith
Family
Fast, cheap and easy
Fun
Quotes
Works-For-Me Wednesday

« Could Have | Main | What Matters »

Looking For a Web Site

My nine-year-old son is eager to listen to what Hubs and I teach him about our faith, but he also wants to learn on his own. He is specifically asking for a "website where I can learn about God." Does anyone have one they'd recommend for a child this age?

Thursday, July 13, 2006 in Faith, Family | Permalink

TrackBack

TrackBack URL for this entry:
http://www.typepad.com/t/trackback/5410202

Listed below are links to weblogs that reference Looking For a Web Site:

Figure 2.6: Tags, appearing both at the end of a blog post and as a navigational tool to the left. Source: http://rocksinmydryer.typepad.com/shannon/.

Blogger Profile. Many blogging tools allow the blogger to create a special "profile" page which contains information about the blogger, allowing visitors to get a glimpse into the person behind the blog. The profile of a blogger may include information such as age, gender, geographic location, profession, interests, and a personal statement. In addition, some blogging platforms provide meta-information about the blog such as statistics about posting, comments, links to other blogs by the same author, and so on; these are displayed as part of the blogger profile. According to Herring et al. [113], the usage of profiles in blogs is increasing; in particular, there is an increasing tendency to reveal at least the given name of a blogger, so that she is not anonymous to readers. The level of detail provided in the profile varies widely—and is related to the blogging platform itself (some platforms tend to attract bloggers which supply substantially more detailed profiles than others [123]). Sample profile pages of bloggers are shown in Figure 2.7.

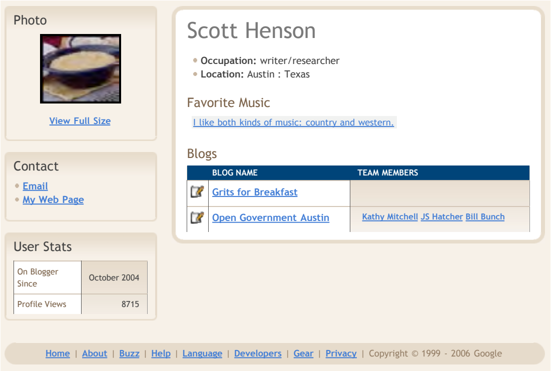
Blogrolls. A blogroll is a list of "favorite blogs" of a blog author: blogs she regularly follows, or finds interesting or important. Typically, the blogroll appears on a sidebar along the blogger's posts, and serves two purposes. First, it is a navigational aid for visitors to the blog, helping them find other related or interesting blogs. Second, the blogroll works as a "measure of confidence" from one blogger to another, acknowledging a (sometimes one-sided) relation between the blogs. Because of the importance assigned to blogroll links, bloggers tend to be careful in adding and removing blogs from their blogroll, making them largely static. Some community-oriented blogging platforms (e.g., LiveJournal or Xanga) provide a similar "friend list" where bloggers reference other blogs from the same platform they follow. As of 2006, 41% of bloggers maintained blogrolls or lists of friends [165]; Figure 2.8 shows an example of a blogroll (on the left side).

2.1.3 A History of Blogging

Some credit Tim Berners-Lee, the creator of the World Wide Web, as creating also the first blog: in 1992, as the web was making its first steps, he maintained a "What's new on the WWW" page, listing new websites and occasionally commenting on them. The National Center for Supercomputing Applications (NCSA), which developed the first graphical web browser, started a similar page a year later. Other web sites mentioned as early blogs and having a format closer to that of current blogs are David Winer's Scripting News,[7] Cameron Barrett's Camworld,[8] and Jorn Barger's Robot Wisdom[9], all created in 1996–1997. In the following two years additional blogs appeared, but at a very slow rate; blogging platforms did not exist and bloggers had to be reasonably proficient in web page authoring and hosting technologies.

The first turning point for blogs came in 1999. An exhaustive list of known blogs compiled at the beginning of the year contained only 23 blogs [37]. But

[7] http://scripting.com
[8] http://camworld.com
[9] http://robotwisdom.com/



Photo

View Full Size

Contact

- Email
- My Web Page

User Stats

On Blogger Since	October 2004
Profile Views	8715

Scott Henson

- **Occupation:** writer/researcher
- **Location:** Austin : Texas

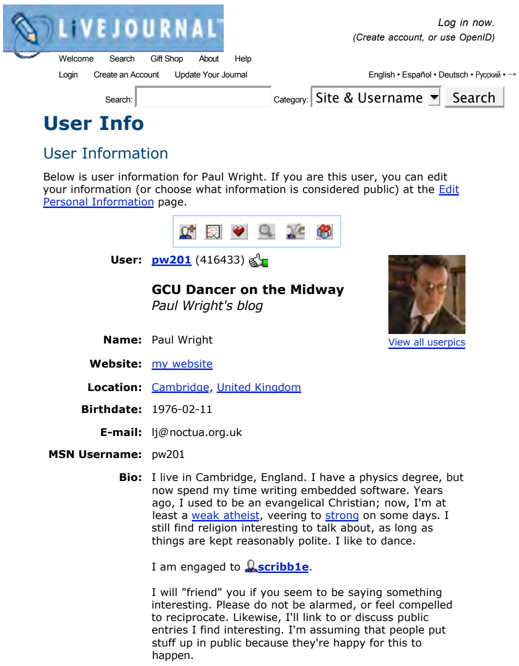
Favorite Music

I like both kinds of music: country and western.

Blogs

BLOG NAME	TEAM MEMBERS
Grits for Breakfast	
Open Government Austin	Kathy Mitchell JS Hatcher Bill Bunch

Home | About | Buzz | Help | Language | Developers | Gear | Privacy | Copyright © 1999 - 2006 Google



LiVEJOURNAL

Log in now.
(Create account, or use OpenID)

Welcome Search Gift Shop About Help
Login Create an Account Update Your Journal
English • Español • Deutsch • Русский • →
Search:
Category: Site & Username
Search

User Info

User Information

Below is user information for Paul Wright. If you are this user, you can edit your information (or choose what information is considered public) at the Edit Personal Information page.

User: pw201 (416433)

GCU Dancer on the Midway
Paul Wright's blog

View all userpics

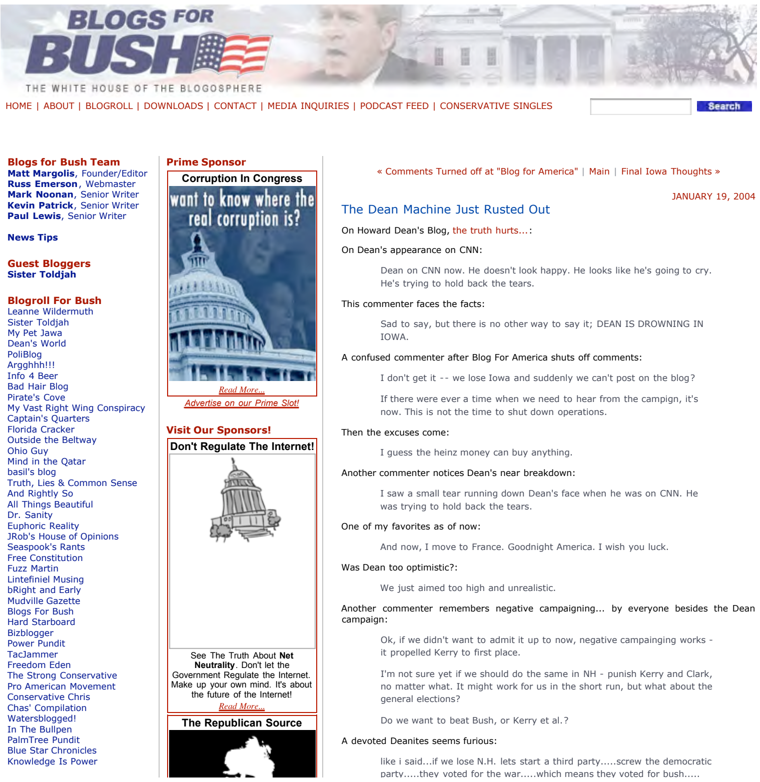
Name: Paul Wright
Website: my website
Location: Cambridge, United Kingdom
Birthdate: 1976-02-11
E-mail: lj@noctua.org.uk
MSN Username: pw201
Bio: I live in Cambridge, England. I have a physics degree, but now spend my time writing embedded software. Years ago, I used to be an evangelical Christian; now, I'm at least a weak atheist, veering to strong on some days. I still find religion interesting to talk about, as long as things are kept reasonably polite. I like to dance.

I am engaged to **scribb1e**.

I will "friend" you if you seem to be saying something interesting. Please do not be alarmed, or feel compelled to reciprocate. Likewise, I'll link to or discuss public entries I find interesting. I'm assuming that people put stuff up in public because they're happy for this to happen.

Figure 2.7: Sample blog profiles. Top: http://gritsforbreakfast.blogspot.com; Bottom: http://pw201.livejournal.com.

during this year, a number of blogging tools were released, the major ones being the community-oriented opendiary.com (formally launched at the end of 1998), livejournal.com, pitas.com, (later diaryland.com), and blogger.com. Many of these platforms were developed by early blogging entrepreneurs, which originally created them to ease the blogging process for themselves and their communities. All platforms offered free, simple tools for setting up and maintaining a blog, supporting features such as comments, permalinks, and blogrolls; a blog could now be created in minutes by anyone, with no technical background required. Indeed, the number of blogs surged, reaching the thousands and tens-of-thousands within a year—but still having relatively little impact on the vast


BLOGS FOR BUSH
THE WHITE HOUSE OF THE BLOGOSPHERE
Search
want to know where the real corruption is?

HOME | ABOUT | BLOGROLL | DOWNLOADS | CONTACT | MEDIA INQUIRIES | PODCAST FEED | CONSERVATIVE SINGLES

Blogs for Bush Team
Matt Margolis, Founder/Editor
Russ Emerson, Webmaster
Mark Noonan, Senior Writer
Kevin Patrick, Senior Writer
Paul Lewis, Senior Writer

News Tips

Guest Bloggers
Sister Toldjah

Blogroll For Bush
Leanne Wildermuth
Sister Toldjah
My Pet Jawa
Dean's World
PoliBlog
Argghhh!!!
Info 4 Beer
Bad Hair Blog
Pirate's Cove
My Vast Right Wing Conspiracy
Captain's Quarters
Florida Cracker
Outside the Beltway
Ohio Guy
Mind in the Qatar
basil's blog
Truth, Lies & Common Sense
And Rightly So
All Things Beautiful
Dr. Sanity
Euphoric Reality
JRob's House of Opinions
Seaspook's Rants
Free Constitution
Fuzz Martin
Lintefiniel Musing
bRight and Early
Mudville Gazette
Blogs For Bush
Hard Starboard
Bizblogger
Power Pundit
TacJammer
Freedom Eden
The Strong Conservative
Pro American Movement
Conservative Chris
Chas' Compilation
Watersblogged!
In The Bullpen
PalmTree Pundit
Blue Star Chronicles
Knowledge Is Power

Prime Sponsor
Corruption In Congress
Read More...
Advertise on our Prime Slot!

Visit Our Sponsors!
Don't Regulate The Internet!
See The Truth About **Net Neutrality**. Don't let the Government Regulate the Internet. Make up your own mind. It's about the future of the Internet!
Read More...
The Republican Source

« Comments Turned off at "Blog for America" | Main | Final Iowa Thoughts »

JANUARY 19, 2004

The Dean Machine Just Rusted Out

On Howard Dean's Blog, the truth hurts...:

On Dean's appearance on CNN:

> Dean on CNN now. He doesn't look happy. He looks like he's going to cry. He's trying to hold back the tears.

This commenter faces the facts:

> Sad to say, but there is no other way to say it; DEAN IS DROWNING IN IOWA.

A confused commenter after Blog For America shuts off comments:

> I don't get it -- we lose Iowa and suddenly we can't post on the blog?
>
> If there were ever a time when we need to hear from the campign, it's now. This is not the time to shut down operations.

Then the excuses come:

> I guess the heinz money can buy anything.

Another commenter notices Dean's near breakdown:

> I saw a small tear running down Dean's face when he was on CNN. He was trying to hold back the tears.

One of my favorites as of now:

> And now, I move to France. Goodnight America. I wish you luck.

Was Dean too optimistic?:

> We just aimed too high and unrealistic.

Another commenter remembers negative campaigning... by everyone besides the Dean campaign:

> Ok, if we didn't want to admit it up to now, negative campainging works - it propelled Kerry to first place.
>
> I'm not sure yet if we should do the same in NH - punish Kerry and Clark, no matter what. It might work for us in the short run, but what about the general elections?
>
> Do we want to beat Bush, or Kerry et al.?

A devoted Deanites seems furious:

> like i said...if we lose N.H. lets start a third party.....screw the democratic party.....they voted for the war.....which means they voted for bush.....

Figure 2.8: A sample blogroll from http://blogsforbush.com.

majority of web users.

The next big change in blogging practices came in 2001, following the events of September 11th of that year. With the global political changes, the dominant topics discussed in blogs shifted from web and technology to politics [85]. Some politically-oriented blogs such as dailykos.com became extremely popular, with millions of visitors; in parallel, blogs received coverage in mainstream media, blogging how-to articles and books appeared, and bloggers were being regularly hired by newspapers [63]. The blogging medium was now better known to the general public, and the number of blogs climbed exponentially. Other large-scale events with similar outcomes include the invasion of Iraq in 2003 and the U.S. election campaign in 2004 (where, for the first time, both presidential candidates included blogs on their websites). The influence of blogs during this campaign extended beyond their readership through interaction with national mainstream media [2], by featuring breaking news and prominent commentary. According to a large-scale survey from late 2004, 27% of internet users in the U.S. were reading blogs—a substantial increase of 58% from the number reported in similar surveys earlier that year [166, 248]. According to the same survey, 12% of internet users commented on a blog post at least once. In a way, between 2001 and 2004 the blogging phenomena transformed from the fringe to the mainstream—in a similar manner to the process which the web itself underwent in the early 1990s.

In the years that followed, blogs became more and more common, with corporate blogs appearing on the sites of many companies, complementing press releases and announcements; some companies (e.g., IBM, Microsoft, or Sun) even encouraged employees to maintain their own blogs, perceiving them as a direct channel of communication with customers. Prominent bloggers were often cited and interviewed by mainstream media, and seen as an important source of news and commentary. During this time period, additional services allowing easy generation of web content by individuals became extremely popular: some were focused on a specific content type (e.g., Flickr for photos, YouTube for videos, del.icio.us for bookmarks), and some providing a complete social-networking platform with blogs, groups, profiles, photos and more (e.g., MySpace). The popularity of these services can be attributed to a large extent to the blogging phenomenon, which transformed internet users from passive consumers of information to content generators. Bloggers are, indeed, trend-setters also for other forms of content generation: surveys show that they are much more likely to share content such as photos and videos than other internet users [165].

As of mid-2006, one in twelve internet users in the U.S. maintains a blog, and four out of ten users read blogs (percentages among teenagers and young adults are higher); the vast majority believe they will continue to blog in the future [165]. The influence of blogging is substantial: in a somewhat controversial decision, Time Magazine named online content creators such as bloggers as the 2006 Person of the Year [95], referring to them as the group that has most influenced events during that year.

Is blogging a passing trend or a long-term change in the behavior of users on the web? It's hard to tell at this stage, but even if blogs are a fad, they are undoubtedly a particularly large-scale and enduring one. At the time of writing, a decade after the first blogs appeared, the blogspace maintains substantial size and influence, and seems to appeal to many—both authors and readers. User-generated content, it seems, is here to stay—at least for the near future.

2.1.4 Blog Genres

Blogs come in a variety of shapes and forms, and various models have been proposed to categorize them. In an early classification of blogs, Blood [38] distinguishes between three types: filters, journals, and notebooks. Filter blogs are those containing mostly links to web sites, "pre-surfed" and commented by the blogger for her readers (hence the name—these blogs "filter" the web for their audiences). Journal blogs are diary-like personal blogs, where posts relate primarily to the blogger's life rather than external events: simply put, these are online, public diaries. Notebooks are blogs consisting of longer essays and commentary, in which the issues addressed are either from the blogger's personal life or external to it.

Similarly, Herring et al. [113, 110] manually divide a random sample of blogs into four types: personal journals, topical blogs, filter blogs, and mixed ones. The proposed types are contrasted by placing them on a "computer-mediated communication (CMC) continuum," indicating the different form of communication taking place in the blogs: at one end of the spectrum are static web pages for which communication is a unilateral one-to-many, and at the other end are various forms of asynchronous CMC, such as email and instant messaging, for which content is updated regularly and discussion is two-sided. The different forms of blogs, then, are derived from their position on this continuum (see Figure 2.9).

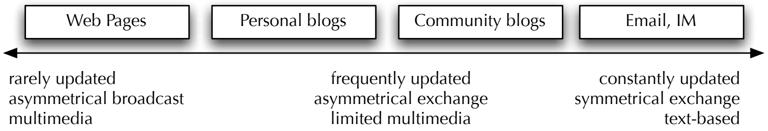


Figure 2.9: Different blog genres occupy different positions on the computer-mediated communication continuum, according to Herring et al. [112].

While many early blogs were of a filter type [37, 25], personal journals currently constitute the majority of the blogspace [165]; for example, although Herring's sample intentionally skipped journal-oriented blogging platforms such as LiveJournal, about 70% of the blogs in it were found to be personal journals. More recent categorizations of blogs such as Nowson's [227] omit filter blogs altogether, focusing instead on the differences between topical blogs and personal ones.

A more general approach to blog type classification is proposed by Krishnamurthy [154], placing blogs on a two-dimensional continuous space: a "personal vs. topical" dimension, and an "individual vs. community" one. Representing these dimensions as orthogonal axes, blogs can be seen as belonging to one of four broad categories: online diaries (personal-oriented, individually-written blogs), support groups (personal-oriented blogs with multiple contributers), commentary (topical blogs maintained by individuals), and "group content" blogs (topical, community-authored blogs). A visualization of this categorization scheme is shown in Figure 2.10.

Other blog types and sub-types have also been discussed, including niche genres such as travel blogs (maintained while the blogger is on a journey), corporate blogs (officially or semi-officially maintained by a company), and multimedia-oriented blogs (e.g., photoblogs—blogs in which the posts are mostly images). The different categorizations of blog types do not collide; in fact, defining blog genres, and categorizing blogs as belonging to one of them, depends to a large extent on the task those types are used for. One can think of various additional ways to categorize blogs, each viable and coexisting with others. One possible such categorization would distinguish between blogs according to their intended audience: the readers which the blogger is trying to reach. Some bloggers do

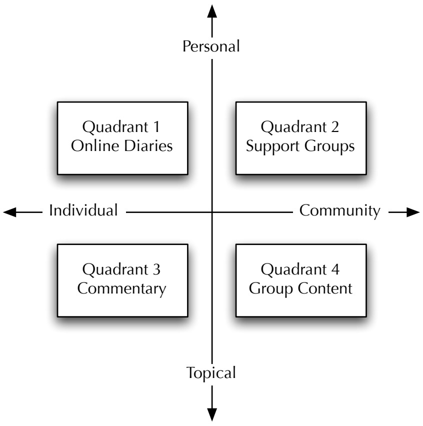


Figure 2.10: The orthogonal dimensionality of blogs according to Krishnamurthy [154].

not make their posts public (or, they are not aware that the posts are public); typically, these are personal journals, intended for the blogger herself, or for a small group of friends and relatives. Bloggers focusing on a specific topic typically intend to reach an audience which has a prior interest in the blog's topic. Finally, some bloggers are reaching out for as large an audience as possible; these tend to be either less personal blogs (i.e., political commentary), or artistic/witty blogs functioning as a platform with which the blogger hopes to win the attention of many. Such a (continuous) classification scheme is shown in Figure 2.11; surveys show that half of the bloggers blog mostly for themselves [165], and those that address an audience are usually aiming at "nano-audiences" of family and friends [239].

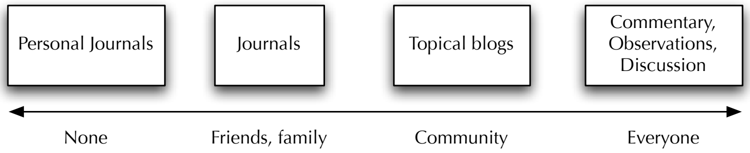


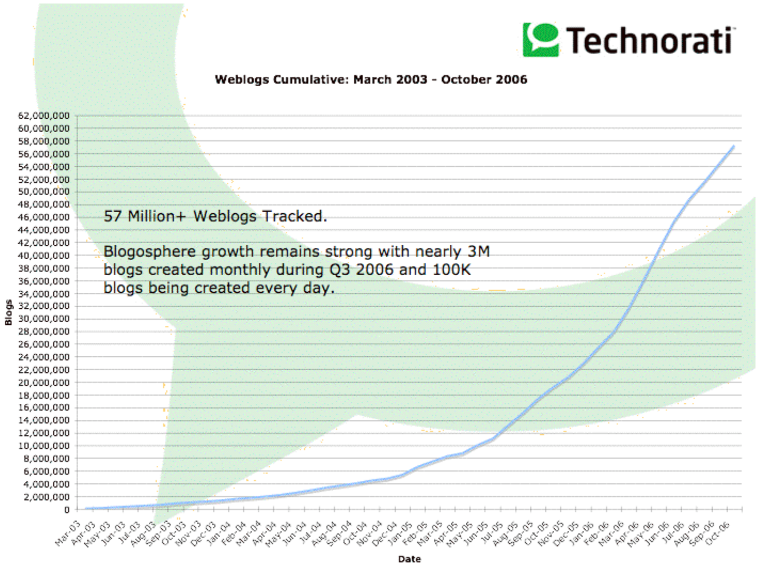
Figure 2.11: Blogs categorized according to their intended audience.

2.2 The Blogspace

The *blogspace*, or the *blogosphere*, is a term used to refer to the collection of all blogs and the links between them. This section surveys this collection, providing descriptive statistics and a structural analysis of it; in particular, we focus on characteristics of the blogspace distinguishing it from other web documents, and making it unique as a corpus for text analytics.

2.2.1 Size and Growth

Total amount of blogs. According to blog search engine Technorati, the total number of publicly-accessible blogs has doubled every 5–7 months in recent years, reaching close to 60 million in late-2006 [66]; this exponential growth is shown in Figure 2.12. Similar trends emerge from surveys conducted by the Perseus Development Corporation in 2003 and 2005 [239, 240]; this total number of blogs is even considered moderate when taking into account the incomplete coverage of non-English (particularly Asian) language blogs.[10] However, many of these blogs are inactive: according to the Perseus surveys, two-thirds of blogs have not been updated in two months, and about a quarter contain but a single post; Technorati also reports a substantial amount of abandoned blogs, as do blogging platforms which publish blogging activity statistics (e.g., LiveJournal). So, while 60 million blogs may have been *created* over the past years, the size of the *active* blogspace—those blogs which are continuously updated—is most likely significantly smaller.



57 Million+ Weblogs Tracked.

Blogosphere growth remains strong with nearly 3M blogs created monthly during Q3 2006 and 100K blogs being created every day.

Figure 2.12: Total number of blogs created, 2003–2006. Source: Technorati.

The active blogspace. A good indication for the size of the active blogspace is the daily number of new blog posts, as tracked by search engines. The growth of this number is more moderate, and is strongly affected by large-scale events; major ones, such as U.S. elections or large-scale catastrophes, result in a surge of new posts. A graph of the number of daily posts during 2004–2006 as reported by Technorati appears in Figure 2.13, showing an irregular, mostly linear growth

[10] For example, reports from 2005 estimate the Chinese blogspace alone at 37 million blogs [96], and the European blogspace at 10 million [281].

pattern. All in all, the number of posts per day in recent years rose from the low hundreds of thousands in early 2003 to 1.2–1.5 million in late 2006—that is, multiplying four-to-five times over the entire period.[11]

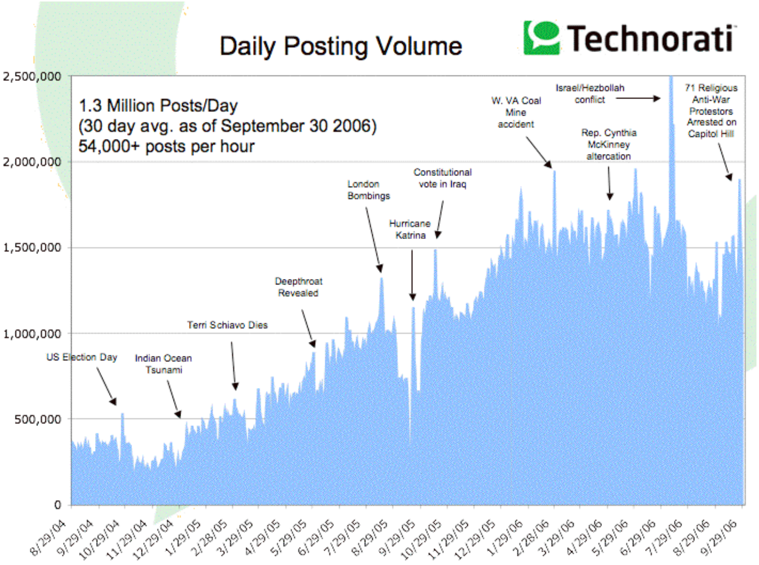


Figure 2.13: Daily number of new blogs posts, 2003–2006. Source: Technorati.

An active blog can be defined in multiple ways: by measuring the posting frequency, the time passed since the last post, and so on. Different definitions result in different numbers of active blogs—anywhere between 1 and 10 million. But under any of these definitions, the *growth* in the number of active blogs—like the growth in the daily posting volume—is linear.

Blogspace growth and the web. How does the growth of the blogspace compare with that of the web itself? According to forums which track the size of the publicly-accessible web over time such as the Internet Systems Consortium[12] or Netcraft,[13] the size of the public web, in terms of number of hosts, has multiplied two to three-fold during the same 3-year period in which the number of blogs doubled twice a year; the growth of blogs in recent years, then, resembles the explosive growth of the web itself in the mid- and late-1990s. The growth of the active blogspace, though, is similar to that of the rest of the web.

Future growth. Forecasts regarding the size of the blogspace in the future vary; while some domains show early signs of saturation (particularly the U.S.

[11] Numbers vary across search engines, mostly due to different definitions of what constitutes a blog and different spam interception mechanisms.

[12] http://www.isc.org/

[13] http://news.netcraft.com/

market [258, 250]), others—such as Asian markets—are projected to show strong continuous growth [96]. At any rate, existing growth rates will not be sustained in the far future, as they are limited by the population of internet users.

In summary, the overall amount of existing blogs has been growing exponentially in recent years, while the active blogspace maintains a more moderate, linear growth pace. Conservative estimates place the number of active bloggers in the few millions worldwide, while more lenient ones refer to tens of millions. Additional growth, particularly of non-English blogs, is expected also in the near future. But whatever the number of active blogs really is, there is no dispute that they provide a glimpse into the lives of a substantial portion of internet users worldwide, on an unprecedented scale.

2.2.2 Structure of the Blogspace

Most blogs—as most other web pages—do not operate in a vacuum: they are connected to other blogs and other web pages. When referring to the Web, we refer not only to web pages but also to the net of links between them; similarly, the blogspace refers not only to blogs but also to the links between them and the structure they induce. In this section we examine the main characteristics of this structure, and compare it with the structure of the web.

The Blogspace as a Scale-Free Network

The term "scale-free network" was coined by Barabási and Albert in 1999 [26] to describe a class of self-similar networks whose topology is invariant to scaling (i.e., the network connectivity properties are maintained as the network increases in size).[14] In particular, descriptions of scale-free networks focus on the observed power-law distributions of node degrees in such networks—which include various biological, social, and industrial networks, as well as the World Wide Web itself (as observed by Barabási and, independently, by others [78, 159]).

Power-law distributions. A random variable X is said to have a power-law (or Zipfian) distribution if, for some constants α and c, $P(X \geq x) = c \cdot x^{-\alpha}$. In such a distribution, the tail asymptotically falls according to the power α. Power-law distributions have been observed in many domains, e.g., frequencies of words in text, distributions of income, and, as noted earlier, the node degrees in the Web graph. When plotted on a log-log scale, a power-law distribution appears as a straight line; for example, Figure 2.14 shows the power-law distribution of link degrees on a subset of the web graph. In simple terms, networks with a power-law

[14] A more formal definition of scale-free networks is given in [168], and involves the degree to which highly-connected nodes in the network are connected to other highly-connected nodes.

distribution of node degrees have a few nodes with very high degrees, and many nodes with low ones.[15]

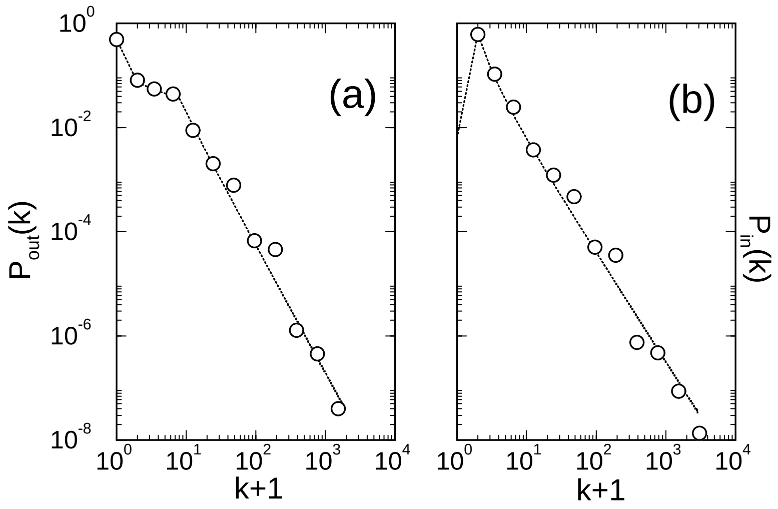


Figure 2.14: The distributions of out- and in-degrees in pages in the nd.edu domain ((a) and (b), respectively). The y-axis shows the probability that a document's degree is k. Source: [6].

Barabási et al. explain the power-law behavior of the Web (as well as some biological, social, and industrial networks) using *preferential attachment*—a model of network formation in which new nodes are more likely to link to existing nodes with high connectivity than to less connected ones. A different generative model for this type of graph is the copying model, in which new nodes copy subsets of the links of other nodes [158].

Power-laws in the blogspace. As the blogspace expanded, the power-law distributions of its inbound and outbound link degrees were observed. In 2003, Kottke [153] investigated the top-linked blogs according to Technorati, as well as the top referrers to his own blog, noting good correlation with a power-law. At the same time, Shirky [276] similarly observed power-law distributions of inbound links as well as sizes of communities in livejournal.com, arguing that the combination of diversity in blogs and the freedom of bloggers to choose to whom they link necessarily leads to inequality in the distribution of links: a small number of highly popular blogs and many almost-unlinked-to blogs. This observation was verified by numerous later studies of link structure in the blogspace (e.g.,

[15] There is an ongoing debate whether different distributions, such as the log-normal one, offer better models for these networks [209]; this is out of the scope of this work, which focuses on the simpler models offered by power-law distributions.

Marlow, analyzing data from 2005 [185]). Figure 2.15 demonstrates this, showing the distribution of inbound and outbound links to blog posts in a collection of over 100,000 blogs, the TREC Blogs06 collection [182], on a log-log scale.[16] Figure 2.16 shows, similarly, the power-law distribution of the inbound links at the blog level in the same collection.

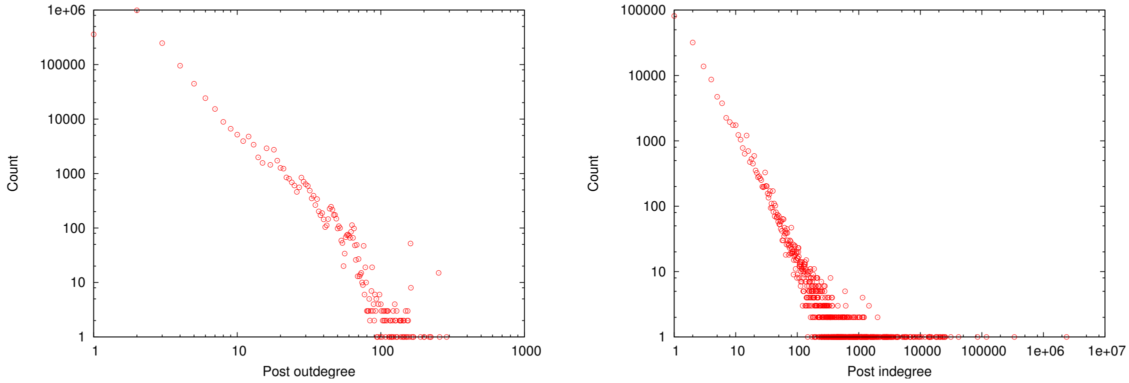


Figure 2.15: The distributions of out- and in-degrees in blog posts in the TREC Blogs06 corpus (left and right, respectively).

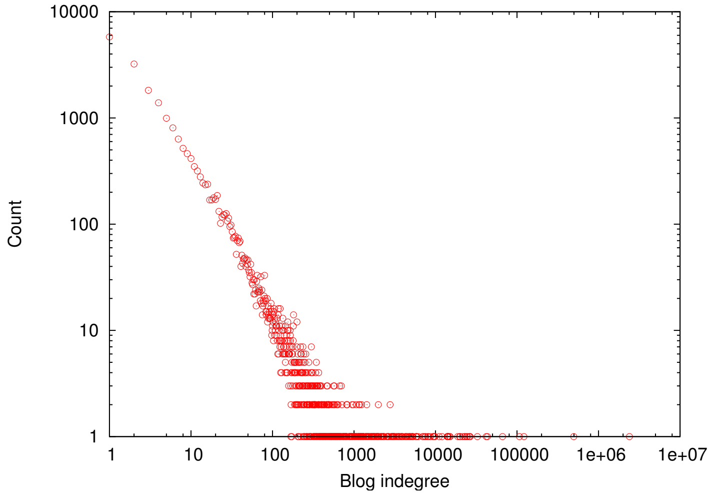


Figure 2.16: The distribution of inbound-degrees in blogs in the TREC Blogs06 corpus.

The power-law distribution of blogs became a source of frustration for many bloggers, who felt that this is a "rich-gets-richer" medium in which it is exceed-

[16] Links within the same blog are ignored. Note the irregularity in the out-degree around degree 11—this is most likely an artificial peak, caused by the fact that the corpus spans 11 weeks, and some links are automated weekly archives.

ingly difficult for newcomers to obtain a large audience. A more careful examination by Marlow [184], comparing an analysis of inbound links originating from blogrolls with an analysis of permalinks coming from the posts themselves, showed interesting results: while both analyses showed power-law distributions, the top-linked blogs differed between the two link sources. In other words, highly-linked-to content is not written exclusively by highly-linked-to bloggers; if inbound links are taken as a measure of influence, this means that influential material is not necessarily written by influential people. Further work by Marlow shows that the amount of inbound links is directly related to the number of posts appearing in a blog during a given period—the extent to which it is updated [185]; this, too, shows that preferential attachment is not the only factor shaping the unequal distribution of links in the blogspace. Drezner and Farrell [63] argue that the distribution of links in the political blogspace is better modeled by a log-normal distribution, implying that, in some domains, it is easier for link-poor blogs to climb up the rank levels. Hard evidence for this can be seen in the exponent of the distribution: while the exponent of the power-law distribution of link in-degrees on the web was reported between 2.1 to 2.5, the exponent of the blogspace is lower: in the TREC Blogs06 corpus, it is about 1.75 at the blog post level, and 1.4 at the blog level. A lower exponent means less differences between the "rich" and the "poor"—the link-rich blogs may have substantially more incoming links than the low-ranked ones, but, on average, there is a greater difference in link in-degree distribution on the web itself.

The "Long Tail" and the A, B and C Lists. Power-law distributions have a "heavy" tail—a longer and slower-decaying one than the common exponential decay. This is most apparent when plotted on a linear scale, ordered by the probability of events, and is referred to as a "power-law tail" or "Pareto tail." Figure 2.17 shows the same link in-degree distribution appearing in Figure 2.16, this time plotted on a linear scale; the asymptotic tail to the right is clear.

In the blogspace, this power-law tail refers to the vast majority of blogs—those having small audiences and few links and citations.[17] The head of the distribution—those blogs which are widely-read, cited, and linked to, is often referred to as the "A-list," and is prominent in many studies of the blogspace (e.g., [38, 184, 4]). The remaining blogs are sometimes grouped into two additional sets: the "B-list," which are blogs with moderate readership and linkage (but not quite as high as A-list blogs), and the "C-list" (or "Z-list") which are the rest of the blogs. Figure 2.18 shows a schematic partition of the blogspace into these lists, according to their link in-degree and listing an example from each of the lists.

Many blogs try to reach as large an audience as possible. According to Du

[17] A New York Times article from 2004 mocked this by saying "never have so many people written so much to be read by so few" [103].

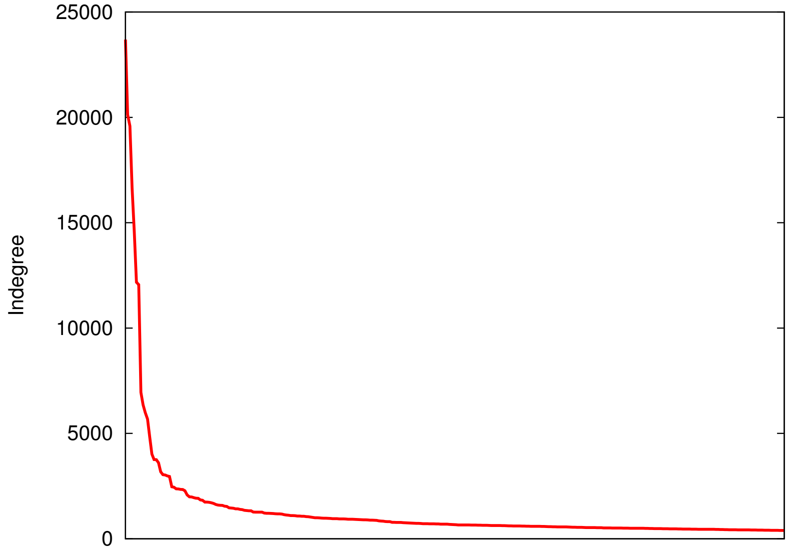


Figure 2.17: The distribution of link in-degrees in the TREC Blogs06 corpus, on a linear scale.

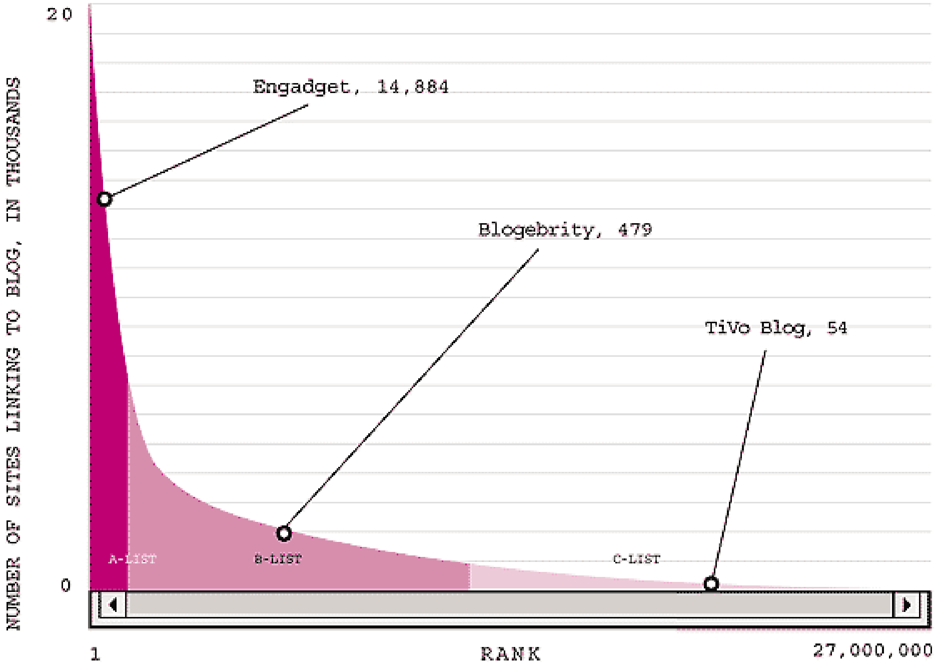


Figure 2.18: The “A”, “B”, and “C” lists. Source: [290].

and Wagner [72], this is not an unreachable goal: over a period of 3 months, almost half of the top-100 blogs (as measured by their link in-degrees, according to Technorati) were replaced by others. However, the rate of change in the top-10 or top-20 linked blogs is lower, and many blogs have been highly ranked for long periods: “making it” to the real top is not trivial.

In 2004, Anderson used the term *The Long Tail* [11] to refer to the effect of the power-law tail on economic models. According to the Long Tail theory, although the tail consists of low-probability events, its collective weight is higher than that of the head because of its length. Applied to the blogspace, this means that while most blogs have few readers, in aggregate terms they reach a much larger audience than the “A” and “B” lists combined. These blogs are the core of the uniqueness of the blogspace: not a few influential journalists, but the unedited voice of the many. The long tail of the blogspace, then, is particularly interesting to those applications that address accessing large amounts of blogs, such as those developed in this thesis.

The Blogspace as a Small-World Network

Small-world networks are a class of networks introduced by Watts and Strogatz in 1998 [307]. These networks are characterized by a short average path length between any two nodes, and a high clustering coefficient—the degree to which the neighbors of a node form a clique, averaged over all nodes in the graph. Formally, in a small-world network the diameter grows with the logarithm of the number of nodes; there is no restriction on link degrees as long as this constraint is met. In practical terms, small-world networks have many communities of neighboring nodes, and the distance between the communities themselves is fairly low; typically, a substantial part of the network is included in the same connected component. Like scale-free networks, small-world networks are found in many domains; Watts and Strogatz observe small-world properties in social networks of movie actors, biological networks, and industrial networks. The web itself, or at least the largest strongly connected component of it, is also a small-world network [6, 1].[18]

Small-world properties have also been observed in the blogspace. An analysis of the links between LiveJournal users shows a remarkably high clustering coefficient of 0.2 [157], meaning that 20% of the time, two friends of the same blogger are also friends. This is substantially higher than the clustering coefficient of random networks of similar size, and also higher than other small-world networks (the web, for example, had a clustering coefficient of 0.11 in 1999 [1]). In 2003, Kumar et al. [156] observed that there is a large connected component in the blogspace, which is growing exponentially. Analyzing data from 2005, Marlow [185] shows that this component includes 90% of the blogspace, and blogs which are

[18] Note that while many networks have been observed to have both small-world and scale-free properties, these are distinct from one another: there are scale-free graphs which do not have the small-world property, and vice versa. For example, the California Power Grid, a network of generators and transformers, is a small-world but not scale-free network; the German highway system is a scale-free network, but does not have the small-world property. See a thorough discussion of the differences between the properties in [221].

not part of it are either spam or pockets of specific, non-English bloggers. Herring et al. [109] observe that most top blogs are reachable from a given random blog with a relatively low number of “hops” through other blogs.[19]

The small-world nature of the blogspace is reflected in two important aspects: formation of communities, and propagation of information throughout the blogspace.

Communities in the blogspace. The dense neighborhoods typical of small-world networks are, in the case of the blogspace, neighborhoods of individuals—sometimes referred to as *virtual communities*. Jones [134] bases his definition of such a community on a combination of factors, including some level of interactivity, sustained membership of contributors and a virtual public space in which communications takes place. This is extended by Blanchard’s notion of “sense of community”—feelings of membership, influence, and integration which tie the members to the community [36]. Most researchers agree that communities of bloggers meet these defining criteria for virtual communities: members of blog communities indeed have a sense of community and are actively participating in group interactions. In fact, some blogs (e.g., metafilter.com) were created as a community effort in the first place, with multiple authors and contributors. Other, singly-authored blogs, have developed a community of followers around them [35]. White [309] distinguishes between different categories of blog communities: communities centered on a specific blogger, with other members of the community participating through comments; communities centered on a given topic; and platform-bound communities, which are closely tied to the blogging environment they operate in (the latter are common in social-networking sites such as MySpace). Most discussions of communities in the blogspace are centered on formation of multiple-blog communities, encompassing several different blogs, such as the study of a knitting blog community presented in [308].

A number of approaches have been proposed for automated identification of communities in the blogspace. Kumar et al. [156] use the link structure alone as evidence of community existence; small cliques are considered as community seeds, then expanded to include dense surrounding subgraphs. The behavior of the identified communities is studied over time by exploring bursts of links in blogs from the same community. Supporting the notion of the blogspace as a small-world network, this study shows that there are substantially more communities in the blogspace graph than in random graphs which have, otherwise, similar properties to that of the blogspace graph. A step beyond the basic link structure is taken by Lin et al. which introduce the concept of *mutual awareness* between blogs [173]. Here, bloggers are assumed to become mutually aware of one another

[19] The same study shows that, contrary to small-world expectations, 42% of blogs are not linked at all; however, the blog sample used is fairly small, meaning that many links may possibly not have been considered.

as a result of bi-directional communication between them, e.g., commenting or trackbacking. These actions are visible by both bloggers—the linking one and the linked one—making them different from simple links between blogs or blog posts. On top of this concept, Lin et al. develop a framework for identifying mutual awareness, representing it in a graph, and mining communities from this graph; evaluation shows that communities discovered using these bi-directional actions are more accurate than those found using the blog link graph only. Similarly, Ali-Hasan and Adamic [7] use different link types such as comments and trackbacks to analyze a small number of blog communities.

Other works focus on discovering not the global community structure of the blogspace, but rather the community of a given blog or topic. Tseng et al. analyze communities on-demand, given a user query [295]. Their framework includes two phases: in the first, blogs matching a query are retrieved and ranked using a PageRank-like method. In the second phase, a graph containing only the top-ranking blogs is analyzed to find connected sub-graphs which are then labeled as communities. By using tomographic clustering and varying the minimal ranking required for a blog to be included in the graph analyzed, this system offers the user a method to control the number and quality of communities found. Chin and Chignell [53] use the structure of the network induced by comments left on posts of a given blog to identify its community. Their work models sense of community characteristics such as influence and reinforcement of needs with structural properties of graphs such as betweenness and closeness. The community identification process is then tuned by questionnaires sent to the bloggers. Finally, Efimova et al. point out a number of non-structural indicators that can be used to discover a blogging community, including blog readership and mutual mentions of community-related events such as meetings. However, computational extraction of most of these is complex, and the experiments performed are limited to analyzing the link structure [75]; later work employs a text analysis framework not to discover a community, but to follow its interests and their evolution over time [13].

Somewhat different from other work in community extraction in the blogspace is the work on inferring related blogs, sometimes referred to as *latent* communities: blogs which may not be directly connected, but which discuss similar topics and could be of mutual interest. Such relations between blogs can be identified using co-citation analysis [127], keyword extraction [269, 32], or topic models [274]. However, a latent community is more a set of topically related blogs than a community as defined in other studies: its members may well not even be aware of each other, much less have a sense of belonging to a community.

Community formation and structure in the blogspace differs across cultures, as shown by Lei et al. [99]: comparing social behavior in the Chinese-speaking and non-Chinese speaking subsets of an invitation-only blogging platform, they observe increased activity in the Chinese-speaking blogs. The hypothesis is that Chinese bloggers are seeking to expand their social network, whereas non-Chinese ones are more focused on maintaining it. Different community structures have also been observed by Gorny [93] for a subset of Russian-speaking blogs, and a difference between linking patterns of left and right-wing political blogs was discussed by Adamic and Glance [2].

It is unclear whether blog communities extend beyond the virtual world into the real one. Liben-Nowell et al. [170], study the profiles of some 500,000 LiveJournal users: in the vast majority of the cases, bloggers who identified themselves as "LiveJournal friends" resided in the same city. This indicates that, in many cases, communities of bloggers are not only virtual, and relations between the bloggers exist in the real world too. However, Ali-Hasan and Adamic [7] conduct surveys between bloggers in communities they identify, finding that the relations between bloggers remained online only. Whether bloggers interact out of the blogspace seems to be tied to the blogging platform, where communities of social-networking oriented sites such as LiveJournal or MySpace tend to reflect real-life relations more than other blogging communities.

Information propagation. The epidemic-like spread of information throughout the blogspace has been studied by Adar et al. [4, 3] and Gruhl et al. [98]. Tracking the propagation of well-cited URLs (Adar) and topics (Gruhl) during specific periods in 2003, both observe distinct types of "interest profiles" that govern the spread of information. Adar describes four groups: one group enjoys sustained interest throughout the period, and usually refers to popular web pages such as large company home pages. The other three groups display different peak behavior: sharp spikes on the first day, decaying rapidly (the "Slashdot effect"), and two slower decaying spikes, one peaking on the first day and usually containing less serious news content, and the other peaking on day two, and containing editorial and other serious news content. Similarly, Gruhl describes three patterns: single-spike topics, multiple-spike ones, and "chatter" topics which have a steady discussion level. The models developed by Gruhl et al. for information spread in the blogspace are shown to be similar to models of disease spread; in these models, individuals go through three phases. First, a person becomes suspectible to being infected; then, the person is infected; finally, recovery takes place (this is known as the SIR model: Susceptibility, Infection, Recovery). In the blogspace, susceptibility occurs when two blogs are related, exposing one to the content of the other; infection is the actual propagation of information from one blog to the other; and recovery occurs when the infected blogger is no longer posting about the infectious topic. Adar et al. follow a simpler model, where inference of a propagation of information between two blogs is derived from a combination of link and content similarity measures between the blog posts, as well as by taking into account the time the meme appeared in the blogs. Both Adar et al. and Java et al. [131], use these information propagation models to identify blogs which are likely to be sources of information—blogs which, if infected with a meme, will maximize the propagation of the meme throughout the blogspace.

In summary, the blogspace is both a scale-free and small-world network, and both its scale-free and its small-world properties are distinct from those of the web itself. This leads to several characteristic features: the scale-free nature results in the "A-list" of blogs and its counterpart, the "Long Tail." The small-world property is reflected in the community-oriented structure of the blogspace. Combined, the scale-free and small-world properties create the patterns of rapid information propagation through blogs.

2.2.3 The Language of Blogs

As the "unedited voice" of people, blogs are assumed to have a different writing style than other genres of written text. Qualitative reports on the language used in blogs support this: Nilsson studied research-oriented blogs, noting informal, self-referential language which combines elements of spoken and written English [224]; Efimova and de Moor observe a combination of dialog and monolog language [74]; Herring et al., as noted earlier, place blogs and the style in which they are written on the CMC continuum between personal web pages and newsgroups [112]. In this section we examine the language used in blogs from a computational linguistics point of view, and report on related observations by others.

To compare the language used in blogs to other genres of text, we used the British National Corpus (BNC) [46], a large (98 million words, 577MB) collection of categorized texts from various sources. As a collection of blogs, we used two samples of English blog texts from the TREC Blogs06 collection: a random set of 500 LiveJournal blogs (3.2 million words, 31MB) and a random set of 500 Typepad blogs (8.7 million words, 54MB). The LiveJournal sample is typical of personal journals written by teens and young adults; the Typepad collection contains a higher concentration of topical blogs, and a somewhat older average blogger age (Typepad is a paid service).

Vocabulary and word usage. A basic method to analyze a corpus is to compare the vocabulary used in it with the vocabulary of a different corpus—and more importantly, compare not only the words used but also their frequencies [142]. Comparisons typically use measures such as χ^2 and log likelihood to identify words which are indicative of the corpus, when compared to a more general collection of texts [183]. To identify such indicative words in blogs, we compared the frequencies of words appearing in the blog sample described above both with word frequencies in the BNC, in Usenet newsgroups and in the TREC Terabyte collection, a large-scale corpus of web documents [54]; the top-indicative words compared to each of these domains appear in Table 2.2.[20]

[20] Frequencies for Usenet were obtained from [272].

Web	Usenet	BNC
I	blog	I
my	trackback	blog
me	I	my
blog	comment	email
you	my	posted
so	your	comment
like	link	trackback
he	posted	web
just	you	me
trackback	please	post
was	will	link
she	us	news
really	school	American
her	night	love
got	remember	your

Table 2.2: Top distinctive words in the blogspace according to the log likelihood measure, compared to the Web (left), Usenet (center), and the BNC (right).

Clearly, aside from blogging-specific words, blogs have a distinctive personal feel, combining the monolog and dialog ("I", "my", "you", "your"), as well as words relating to personal surroundings ("love", "school", "she/he") and references to current events. This is even more evident when restricting the blog word frequencies to the personal-oriented LiveJournal blogs only; in that case, the top distinctive words are almost all of personal-experience nature. In terms of usage of infrequent words, a comparison of blogs and various genres in the BNC shows similar levels to those found in school essays and fiction, and substantially lower than those of newspaper text [227].

Further insight is obtained by comparing the language model of blogs with the models of various genres of the BNC, as well as the web itself.[21] We compare these models using the Kullback-Leibler (KL) divergence, a measure of similarity of distributions which quantifies the amount of information wasted by encoding events from one distribution using the other; KL-divergence has been successfully employed in text classification and retrieval for measuring distances between texts [231], and is discussed in more details in Chapter 3. The KL-divergence values between the language of blogs and other languages are shown in Table 2.3. Lower values mean a higher similarity to the language used in blogs.

Again, the word usage in blogs most resembles that of personal correspondance; the most similar models are a combination of spoken and informal written language, and the least similar ones—formal written texts.

[21] Language models are discussed in more detail in Chapter 3; essentially, they are probability distributions over words and word n-grams, assigning to each n-gram a probability of observing it in a given language.

Genre	KL-divergence
Personal Letters	0.25
TV/radio discussions	0.28
Interviews (spoken)	0.33
Conversations	0.42
School Essays	0.43
Usenet	0.44
Tabloids	0.46
Newspapers	0.48
Fiction	0.51
University Essays	0.74
Web	0.75
Scientific Articles	1.06

Table 2.3: KL-divergence values between the word frequencies of blogs, BNC genres, Usenet, web corpora.

Complexity and style. Next, we examine the complexity of the language used in blogs, again comparing it to other domains. A widely-used measure for the complexity of text is *perplexity*, an information-theoretic measure of the predictability of text, given a language model [42]. The perplexity score can be interpreted as the average number of words that may follow a given word in a language, given the context of the word (i.e., the words preceding it); it is therefore higher for general-domain texts than for focused ones (e.g., the perplexity of the general-English Brown corpus is 247, and that of a collection of medical texts 60). While perplexity is used mainly in the context of speech recognition systems, it has also been applied to text analysis and information retrieval [17]. Table 2.4 shows the perplexity values for various BNC genres and our blog samples; the language models evaluated were the commonly used tri-gram models with Good-Turing discounting (again, more details on language models and perplexity calculation are given in Chapter 3). In addition, the table shows the out-of-vocabulary (OOV) rate—the percentage of words in a test sample that did not appear in the training one. OOV words are normally not used when calculating perplexities, and provide a separate indication of the complexity of the language: texts with high OOV rates are more topically diverse than those with low OOV rates, resulting in larger vocabularies.

The relatively high perplexity of blog language, compared with other genres to which it is similar in the type of vocabulary (e.g., personal correspondence), indicates less regularity in use of language: sentences are more free-form and informal (especially in the personal journal blogs), and adhere less to strict rules. Additionally, the high out-of-vocabulary percentage indicates a less topically-focused corpus, as well as higher prevalence of typographical errors and neologisms and, possibly, an increased level of references to named entities from the blogger's environment (e.g., names of people and locations in the surroundings of the blogger).

Genre	**Perplexity**	**OOV**
Personal Letters	55	3.4%
Interviews (spoken)	114	4.2%
Conversations	118	5.0%
TV/radio discussions	195	7.5%
University Essays	234	7.8%
Fiction	245	11.2%
Blogs (Typepad)	*261*	*15.2%*
School Essays	295	8.5%
Blogs (Both)	*301*	*15.2%*
Scientific Articles	323	11.1%
Blogs (LiveJournal)	*332*	*14.1%*
Newspapers	355	10.9%
Web	371	4.5%
Tabloids	437	11.1%

Table 2.4: Perplexity and Out-of-Vocabulary percentages: blogs compared with various BNC genres and with the web.

This leads to higher complexity in the analysis of blog text: for example, Doran et al. [71] find that annotation of events in the blogspace is substantially more involved than similar annotation in mainstream media, due to the difference in writing style.

An additional measure often used in computational stylistics is *readability*: the level of readers who can comprehend a text and their fraction of the population. There are several formulas for approximating readability, including the Gunning-Fog Index [101], the Flesch-Kincaid Formula [145], and SMOG Grading [189]. All measures combine the number of syllables or words in the text with the number of sentences—the first being a crude approximation of the syntactic complexity and the second of the semantic complexity. Although simplistic and controversial, these methods are widely-used and provide a rough estimation of the difficulty of text. Scores are calculated on a scale which corresponds to the minimal school grade required for reading or understanding the text; very easy passages are graded 5 to 6, newspapers and magazines are typically graded 10 to 12, and technical articles can reach scores of 16 or more.[22] Table 2.5 shows the median readability measures of the same samples of blogs used earlier, compared to the median scores of various genres in the BNC and the web.

Clearly, blogs—especially personal journal ones—tend to use shorter sentence length (and shorter word length); readability scores are close to fictional prose, tabloids, and school essays. To some extent, this can be related to the bloggers' average age, which—again, particularly for journal blogs—is in the teens. A comparison of blogs and genres in the BNC using the F-measure [114], a measurement of the degree of the formality of text performed by Nowson shows similar results:

[22] Flesch-Kincaid scores are somewhat lower than other measures.

Genre	Gunning-Fog	Flesch-Kincaid	SMOG
Conversations	4.8	1.6	5.6
Interviews (spoken)	7.2	3.8	7.4
Fiction	8.4	5.5	7.8
Blogs (LiveJournal)	*8.6*	*5.6*	*8.1*
Tabloids	9.4	6.6	8.7
Blogs (Both)	*9.9*	*7.0*	*8.9*
TV/radio discussions	10.1	7.0	8.9
Blogs (Typepad)	*10.7*	*7.5*	*9.8*
School Essays	11.3	8.0	9.9
Personal Letters	11.7	8.6	9.8
Newspapers	12.1	9.7	10.9
University Essays	13.9	10.8	12.2
Web	15.9	13.2	13.9
Scientific Articles	16.5	13.1	14.0

Table 2.5: Readability measures: blogs compared with various BNC genres.

the formality and contextuality of blogs resembles that of school essays [227].

There are numerous other methods to compare the language style of corpora [139, 286, 14]; we choose one of the most commonly-used features, frequencies of part-of-speech tags—nouns, adjectives, and so on. Table 2.6 shows the different percentages of key part-of-speech tags in blogs and some BNC genres (tagging was performed with TnT [39]). Comparing the various parts-of-speech, the language of blogs again shows a combination of different domains: the low percentage of prepositions typical of informal spoken text, combined with high levels of common nouns—often found in formal writing such as newspapers and high levels of proper nouns. Pronouns are not as prevalent as in many other genres, mostly since first- and second-person pronouns ("I," "my," "you"), which are common in blogs, are grouped by part-of-speech taggers with third-person pronouns ("he," "they") which are common in many domains.

Language style variation. Gender and age of bloggers are strongly reflected in the language they use [121, 227, 266]. For example, Schler et al. [266] compare words used by different age groups and genders of blogger, finding that males tend to use more technology and political-oriented words, while females use more family and personal-life-oriented words; also, an expected difference in vocabulary appears when examining different age groups—teen bloggers discuss school and live-at-home issues, bloggers in their twenties are occupied with college and going out, and bloggers in their thirties discuss family, work and politics. To a lesser extent, personality traits can also be computationally predicted from a blogger's writings [230, 227]. In contrast to these findings, Herring and Paolillo find that gender variations are minor, and are more related to the blog genre [111]—however, the sample used is relatively small.

Genre	Pronouns	Nouns (Proper)	Nouns (Common)	Prepositions
Conversations	11.5%	1.1%	17.4%	8.75%
Fiction	10.1%	0.6%	23.3%	12.2%
Interviews (spoken)	9.2%	0.7%	18.7%	12.3%
TV/radio discussions	7.6&	0.5%	21.5%	13.8%
School Essays	7.1%	0.4%	22.5%	14.4%
Personal Letters	6.8%	0.6%	22.9%	14.5%
Tabloids	5.7%	0.5%	28.7%	12.8%
Newspapers	3.6%	0.4%	28.1%	14.9%
University Essays	2.6%	0.1%	27.4%	15.8%
Scientific Articles	1.5%	0.2%	25.9%	15.5%
Blogs (Typepad)	4.2%	0.6%	31.7%	11.9%
Blogs (LiveJournal)	6.5%	1.1%	23.7%	11.3%
Blogs (Both)	4.8%	0.8%	29.6%	11.8%

Table 2.6: The percentages of some part-of-speech tags in blogs and other genres.

In summary, from a computational linguistics point of view, the language used in blogs is substantially different from that of other mediums. Important characteristics of blog language include a high level of first- and second-person references, informal (yet accessible) text, and vocabulary resembling that of personal correspondence, but less repetitive.

2.2.4 Demographics

As the collective writings of many individuals, blogs are a natural domain for demographic analysis, of two types: both the study of the bloggers' self-reported profiles, and automated identification of demographics from the content and features of a blog. Work on demographic classification of blogs is similar to that performed on non-blog text, using standard text classification approaches [266, 45]; we survey here the main work on analyzing known profiles, separately for gender/age and for geographic location.

Gender and Age

Surveys of bloggers yield somewhat conflicting figures for the female/male ratio in the blogspace: a survey conducted by the Perseus Development Corporation in early 2003 reports that 56% of bloggers are females. Another survey conducted a year later by the Pew Internet & American Life Project concluded the opposite, with the majority (57%) of bloggers being male [248]; the following Pew survey in 2006 found an even distribution between males and females [165]. Studies of the data itself (rather than surveys) show similarly mixed results, including reports on majority of males [113] and females [227]. A closer examination shows different female/male ratios in different age groups: a majority (63%) of females among young bloggers, and a small majority (53%) of males in adult bloggers [266]. Overall, it seems the blogspace is not dominated by either of the sexes, although different blogging patterns exist: females tend to write longer blog posts (double in size, on average [227]), and males dominate the topical and filter blog types [113].

The distribution of age groups in the blogspace, on the other hand, is substantially skewed. All studies note that the majority of bloggers—in some surveys, up to 90%—are under 30, with a particularly large percentage of them being teens [113, 165, 227, 240]. Given the number of bloggers—up to a few tens of millions, as of late 2006—and the skewed distribution of age and location among them, it can be inferred that, in some countries and for some age groups (e.g., teens in Japan or the U.S.), blogs are extremely widespread—much more than the numbers typically given for blog popularity among the entire population of internet users. As expected, the time of day in which bloggers post depends on their age: one study shows that younger bloggers post late, while bloggers in working-age tend to post during the afternoon [45]; other work suggests correlation between activity in Facebook,[23] a college-age social-networking site, and schoolwork—with substantial decline in activities during weekends.

Distributions of age and gender differ across the globe. An analysis of 52,000 blog posts for which the blogger's age is known shows a different distribution in different countries, with as much as 8 years difference between the average blogger age [45]. This suggests a cultural bias in blogging practices between countries (similar observations, focused on deviations of Russian-speaking blogs from other blogs, are made by Gorny [93]).

The gender and age distributions in the blogspace differ substantially from those found in other computer-mediated communication domains such as newsgroups and discussion boards, which are dominated by adult, white, tech-savvy males [110]. The representation of minority groups differs as well: a 2006 U.S.-based survey shows the representation of African Americans and Hispanics among bloggers is substantially higher than in the general internet population [165].

Language and Geographic Location

In the early days of blogging, the geographic distribution of bloggers was centered in the U.S., and consistent with concentrations of high socio-economic status [171, 157]. However, as blogging became more common, the geographic profiles of bloggers shifted; a 2006 survey shows increased popularity of blogging in suburban and rural areas [165]. Global distribution changed too: in mid-2006, more Asian blog posts were tracked by Technorati than English ones [66], with Chinese showing particularly fast growth. In fact, Chinese and Japanese are much more dominant in the blogspace than on the web: Chinese web pages consist of

[23] http://facebook.com

14% of all web pages and Japanese consist of 9% of them, compared with English which covers more than a third of the web (source: Wikipedia). However, a higher prevalence of certain languages such as Portuguese and Farsi in the blogspace compared to their popularity on the web may be related to technicalities (e.g., preference of local ping servers over global ones) [185].

Bloggers from different locations show different blogging patterns; an analysis of blog profiles [122, 123] and ping notifications [123] shows a correlation between blogger location and blogging practices, such as the blogging platform used or the average hour of posting. As a geo-tagged collection of informal texts, the blogspace can also be used to demonstrate the geographic differences in jargon and language use: certain words are used only in some areas [179]; additional applications include mining blog posts associated with a location for experiences and activities in that location, for tourism purposes [160]. Preliminary experiments in identifying the location of bloggers from the blog text using a machine learning approach are reported in [322], with modest success rates.

To summarize, the blogspace is still dominated by young, tech-aware people—but this is changing, as blogging is adopted by additional audiences and as technology becomes more and more pervasive. Blogs are substantially more mainstream than a few years ago, spreading worldwide and throughout many sectors of society.

2.3 Computational Access to Blogs

After introducing blogs, the blogspace, and the characteristics of both, we now turn to view the blogspace as a domain for knowledge extraction, discuss a number of areas in computational analysis of blogs relating directly to this thesis, and survey related work.

2.3.1 Search and Exploration

Searching and Ranking Blogs

As the amount of information on the web exploded in the late 1990s, the search paradigm became the main channel through which web information is accessed. By the time blogs emerged, search technology was ubiquitous, and as the blogspace gained momentum, search engines dedicated to it quickly surfaced. Early discovery platforms such as the non-profit Blogdex and Daypop (2001) were soon followed by commercial services supporting search, major ones being Technorati (2002), BlogPulse and Feedster (2003), PubSub (2004), and Sphere (2005). In later stages—particularly after 2005, when blogs were already an established, mainstream medium—large-scale web search engines developed or acquired blog search services and offered them to users as separate search mediums (Google, Ask.com), or integrated search results from blogs into web search results in marked, separate listings (Yahoo). Typically, blog search engines offer two separate ranking approaches: recency-based (latest post displayed first), or a more traditional combination of keyword relevance and authority, estimated by link indegree. All engines originally focused on retrieval of blog posts (rather than entire blogs), assuming this is the information unit which is of interest to the searcher—although Technorati later added blog-retrieval (which it calls "exploration"), and Google has a similar feature, "Related Blogs."

While most web search engines present relatively simple user interfaces and focus on keyword-based search, many of the dedicated blog search engines featured, from their early stages, advanced navigational tools. These tools were designed to take advantage of the properties of the blogspace—structure, timeliness, and language type—indicating that users are interested in more than the limited web search scenarios. Structure-related tools include following conversations between bloggers (BlogPulse), exploring the link environment of a blog (Technorati), listing top-linked posts, blogs and news stories, and directly linking to the blog of each post in the search results (all engines). Timeline-related tools include displaying daily occurrences of search terms over time in the blogspace—enabling quick identification of bursts in topics (BlogPulse, followed by others), and temporal analysis of posting patterns in the retrieved blogs (BlogPulse, Sphere). Language-related tools include mining phrases and named entities (BlogPulse), identifying related news stories and books (Sphere), and searching blog tags (Technorati). Another feature setting blog search engines apart from web search engines was syndication of the search results: by offering these as feeds, blog search engines provided a simple way for searchers to be updated about changes in the results of their searches. This was done both because the dynamic nature of the blogspace, coupled with recency-based ranking, results in frequent changes to top-ranked posts, and because blog readers were more likely to use syndication technologies and appreciate the feature.

The challenges facing blog search services differ from those web search engines face. While the amount of data in the blogspace is small when compared to the web, refresh rate is of higher importance: many blog searches relate to ongoing events [205], and returning the latest results is of crucial importance. Additionally, web searches often focus on early precision, since users only examine the top results [284]; but in the blogspace, recall is equally important, as users are often tracking references to names of products or people (including themselves, through vanity searches) and expect complete coverage (this is discussed in more details in Chapter 8).

Ranking search results. Usage of link-analysis methods to estimate web page authority has had a dramatic effect on the quality and reliability of web search. As blogs, too, exist in a hyperlinked environment, a number of link-based methods for authority ranking in the blogspace have been explored. Fujimura et al. [80] propose *EigenRumor*, a HITS-based method [146] in which the analyzed link graph contains two types of nodes: those representing bloggers, and those representing posts. Links are created between a blogger and her posts, and between a post and any blogger it references (the reference is typically to a specific post, and is converted to a link to the blogger). The usual hub and authority scores are calculated from this graph; ranking search results according to these scores shows improved performance over content-based ranking only. Wu and Tseng [316] employ another variation on the HITS algorithm, in which the link graph is constructed between blog posts rather than blog pages. In addition, the post-to-post links are weighted according to the similarity between the posts (estimated with tf·idf -weighted cosine similarity) and their temporal distance. The hub and authority scores which are calculated per post are then propagated to the blog level using a range of different methods, obtaining an average authority of posts in the blog and an overall authority.

While link-analysis methods are aimed at estimating authority, the dynamic nature of the blogspace sometimes requires a different approach to ranking. As discussed earlier, blogs induce epidemic-like information propagation patterns; in such settings, it is often useful to identify the source of the information—the blog which infected the rest. Adar et al. [4] propose *iRank*, a ranking algorithm aimed exactly at this; it operates by constructing the information-flow graph of blogs (by following explicit and implicit citations in the posts), and calculating PageRank values over it.

Blog retrieval at TREC. An important step in the process of understanding and developing retrieval technologies for blogs was taken with the introduction of a Blog Track at the annual Text REtrieval Conference (TREC) in 2006 [235]. The task investigated in this track was *Opinion Retrieval*: identifying blog posts which express an opinion (positive, negative, or mixed) about a given topic or entity. Typical systems developed to address this task employed a two-stage process: first, standard retrieval approaches are used to retrieve blog posts which are topically relevant for the topic; then, various techniques are applied to the top-retrieved posts, to identify the presence of opinions in them. Among the techniques used for the latter stage were dictionary-based shallow sentiment analysis methods, text classification approaches, and linguistic analyses. Overall, the results of the track indicate that a strong base ranking is substantially more important than the various approaches used to detect opinionated content. Having said that, it should be noted that as this track was held for the first time, participants did not have access to training data for opinion retrieval; given such training data (e.g., the results of the first run of this track), future performance may indicate a stronger effect of opinion-extraction approaches on retrieval scores.

More details about the opinion retrieval task at TREC and the approaches of participants to it are given in Chapter 9, which focuses on this task.

Topics and Trends

A recurring theme in exploratory analysis of blogs is the identification and tracking of topics in the blogspace. Topic Detection and Tracking (TDT) has long been studied in a similar evaluation forum as TREC, for timelined news corpora [8], with tasks such as identifying topics in a stream of news articles, locating the first reference to an emerging story, and classifying articles by topic. The blogspace proves an especially interesting domain for TDT applications, as it is both a timelined and hyperlinked domain. In particular, much work focuses on trends in the blogspace—topics and concepts which have received sustained attention over some time, sometimes visible blogspace-wide.

The effectiveness of simple methods for trend detection in the blogspace are shown by Hurst in [124]: a number of possible trend models are proposed (bursty, linearly increasing, etc.); by calculating the similarity between these models and the actual occurences of terms over time in blogs, different trends are identified. Similarly, Oka et al. [233] extract topics by following the frequencies of terms over time, measuring their deviation from the average frequencies. Terms which are significantly overused during a period are further analyzed by matching them with co-occurring terms which have the same temporal frequency profile. Comparisons of popular terms in the blogpsace with popular terms in mass media have also been performed using a similar approach, identifying overused terms in a given time-slot [81, 179]. Qi and Candan [245] use a segmentation approach to analyze topic development in filter blogs over time, identifying three major development patterns: topics which maintain a sustained level of discussion in a blog ("dominant" pattern), topics which slowly change to other ones ("drifting" pattern), and topics which appear and disappear suddenly, temporary shifting the discussions in the blog ("interrupted" pattern). The identified patterns can then be used to construct a navigational, topical timeline of the filter blog, as well as visualize it. In [52], Chi et al. improve on trend-analysis methods which are based on term counts only by combining the temporal profile of each blog with its link structure; the results are more robust trends, and, sometimes, identification of shifts in keyword usage patterns which are not clearly observed from counts alone. Similarly, Zhou et al. [325] incorporate a social network analysis framework with a trend-detection approach; this enables not only identification of trends using the interactions between participants in the network, but also locating, for each trend, the influential actors of the network which lead it.

Other than these content analysis based methods, link analysis has also been used for topic tracking in blogs: a HITS-based method for identifying "hot stories" in the blogspace and blogs which summarize them was described by Wu and Tseng [316].

As with other forms of user-generated content, tags are often used both by the bloggers themselves, to facilitate navigation and organization of their blog, and by blog readers—to categorize and store posts from various sources [186]. Most blogging platforms utilize tags as a category-based navigational tool in the blog; some platforms also offer "tag clouds"—a visualization of the tags used in a post or in the entire blog, which indicates their popularity. As noted earlier, some blog search engines enable keyword search in blog tags; other services (e.g., Shadows[24]) focus exclusively on tag search, although their tag space contains not only blog tags but also tags assigned to other web content. Automated assignment of tags to blog posts by extracting terms and phrases has from the post has been proposed by Berendt and Navigli [32]; Brooks and Montanez [41] explore a similar approach, as well as use hierarchical clustering of tags to construct a topical hierarchy of blogs.

2.3.2 Sentiment Analysis in Blogs

Sentiment analysis is a field in computational linguistics involving identification, extraction, and classification of opinions, sentiments, and emotions expressed in natural language. Much of the work in this field has been focused on classifying the polarity of text—identifying whether opinions expressed in it are positive or negative. Work on sentiment analysis outside the blogspace is plentiful, primarily targeting product reviews (e.g., [65, 238, 175]) but also news corpora [312], message boards [64], and other sources of information. Immediate applications of sentiment analysis methods include tools for tracking attitudes in online texts towards commercial products and political issues, among others.

As the blogspace provides a unique window into people's personal experiences and thoughts, research of sentiment analysis in blogs constitutes a substantial amount of the computational studies of blogs in general (see, e.g., the percentage of work dealing with sentiment analysis in blog research venues such as the workshops held at the annual World Wide Web conference during 2003–2005 [317] or the AAAI Symposium on Analyzing Blogs held in 2005 [47]), and is also targeted by the commercial market (e.g., [192]). Most work on sentiment analysis in the blogspace consists of applying existing methods for sentiment classification to blogs; reports on the success of these are mixed. Chesley et al. [51] used a combination of textual features with verb and adjective polarity to classify sentiment in English blogs, achieving accuracy levels which approach those reported in other domains. Ku et al. [155], on the other hand, observe substantially lower accuracy for Chinese blogs than for a Chinese news corpus—both at the sentence level and the document level, and Mullen and Malouf [213] report low accuracy in classifying the political orientation in a blog-like online discussion site.

Blogs, however, offer more than simple positive or negative polarity classification, and more fine-grained analysis has been performed for classifying emotions

[24] http://shadows.com

and moods [84, 167, 236]; as this task is also addressed in this thesis, related work will be discussed in Chapters 4 and 6.

2.3.3 Blog Spam

Ranking high in the results of search engines is crucial to web sites, particularly commercial-oriented ones; this led to the multi-million industry of *Search Engine Optimization* (SEO), which deals with improving the rank of web pages in search engine results. While some SEO methods are considered “legitimate” by search engines (e.g., improving the content and layout of web pages, maintaining a high level of updates to the content, and so on), some are viewed as search engine spam, aimed at pushing web pages “higher than they belong” in search results by misleading search engines about the true nature of a page. For example, some search engine spam methods are aimed at increasing the number and quality of inbound links to a page, as web search engines usually employ link-analysis approaches to estimate the importance of a web page using its inbound links.

Link-based search engine spam methods are found in many domains on the web, and the blogspace is no exception. In particular, two types of spam plague the blogspace: *comment spam* and *spam blogs*. Comment spam exploits one of the features bloggers appreciate the most: the ability of visitors to a blog to comment on a post, creating content which is displayed alongside the post itself. Since comments can include links, spammers use them to create incoming links to their web pages, increasing their link-based prestige score. This problem has become so substantial for search engines as to trigger cooperation between rival search engines, introducing a special hyperlink tag (`rel="nofollow"`) that attempts to address this [225]. A variant of comment spam is trackback spam; in fact, the automated nature of the trackback mechanism makes it an even easier target than comment spam, with estimates of the percentage of spam trackbacks being as high as 98% [214].

Spam blogs (splogs), on the other hand, are full-blown blogs whose content is automatically generated or hijacked from other sources; these appear as legitimate blogs to search engines which index them and follow the links found in them. Splogs may also be used to host advertisements and benefit from generated revenue. Some splogs, as well as other spam web pages, also generate spam pings (spings)—ping update notifications aimed at improved indexing and recency estimation of their pages.

A few factors have contributed to the increase of both types of spam in the blogspace. First, the relative ease of setting up a blog (or generating comments to existing blogs) by automated agents have made life easy for spammers: with relatively little effort they are guaranteed links hosted by a respected blogging platform, alongside legitimate links. On top of this, the increasing importance of blogs on the web prompts search engines to assign high importance to links originating from blogs, and to increase the significance of recency in their ranking algorithms—making links from blogs even more worthwhile. Finally, blogs enjoy a larger and larger audience, improving the effectiveness of advertisements placed on blogs (both legitimate and spam ones). All of these lead to a growth in the amount of spam blogs from an estimated 2% of the blogspace in March 2005 to 10–20% in early 2006, causing major skews in analyses of blogs [125]. Java et al. [131] show that the effect of spam in the blogspace on various blog-related tasks is substantial, particularly when using spam-vulnerable methods such as HITS.

As is the case with other forms of web spam, various approaches have been investigated for preventing or reducing blog spam. Kolari et al. [150, 149] use a machine learning approach to detect splogs with accuracy levels reaching 90%; this work operates on the blog level rather than on single posts, building on the fact that splogs differ from legitimate blogs in terms of language and link structure. Inspecting the entire blogspace, they discover an overwhelming majority of spings (75% out of all blog pings). A similar approach for exploiting the divergence between typical blog language and spam blog language is used by Narisawa et al. [219]. Lin et al. [172] base their spam detection approach not only on content analysis, but also on automatically detecting regularities in the structural and temporal properties of the blog.

But spam blogs may be blocked even prior to fetching them, based on their ping notifications alone: Hurst points out some differences between pinging patterns of spam-generated ping (spings) and legitimate ones [123], suggesting that these differences can be used to distinguish spings from legitimate ping notifications. Salvetti and Nicolov [261] successfully adopt a URL classification scheme used for general web pages to sping identification.

Work on comment spam is less common than that on blog spam: a collaborative filtering approach to comment spam filtering is proposed by Han et al. [105]; here, bloggers manually filter spam in their blog, propagating their decisions to trusted fellow bloggers. While this is an effective approach, it requires substantial human effort on the bloggers' part.

The issue of detecting and filtering spam in web documents is usually treated as orthogonal to the analysis of the documents themselves. As the focus of this thesis is text analytics, we will follow this approach, assuming to a large extent that the data we analyze is clean of spam. As an exception, we treat comment spam—which we view as more unique to the blogspace than splogs (which resemble other forms of web spam pages)—in Chapter 5. In addition, we discuss spam in the context of retrieval in Chapter 9.

2.3.4 Blogs as Consumer Generated Media

Consumer Generated Media (CGM, [34]) is a term referring to opinions, experiences, and commentary regarding products, brands, companies and services which originate in consumers of these products. Sources for CGM—a particular type of user-generated content—are many, and include emails, discussion boards, product review sites, Usenet articles, and, of course, the blogspace. Blogs are of special interest as a source of CGM for a few reasons: first, the informal, unedited content of blogs is considered by some marketers and consumers as a more direct reflection of people's attitudes than other sources; second, the popularity of blogs exceeds that of other, more technical forums such as Usenet; finally, as bloggers are early adopters of technology and heavy users of it [165], they form an important focus group for analysts. Additionally, commentary posted by bloggers may be picked up by mainstream media and reciprocated; this is more common in the case of high-influence blogs.[25]

As the volume of blogs and other public online forums such as message boards increased in the early 2000's, commercial enterprises which base their business model on mining business intelligence from these sources emerged; examples are Nielsen BuzzMetrics,[26] Umbria Communications,[27] and Cymfony.[28] The technologies these companies employ are aimed at answering marketing questions such as "What are people saying about my product" by examining CGM. The methodology used by Nielsen BuzzMetrics is discussed in [88, 5]: it consists of a platform combining crawling, information extraction, sentiment analysis and social network analysis. Case studies presented using this platforms show how marketing information such as the type of user complaints about a product can be effectively and efficiently mined from blogs, in a process which would have been much more complicated with other means. A similar platform is discussed by Tong and Snuffin in [293].

A demonstration of the potential of blog analysis for marketing is shown by Gruhl et al. [97], which observe that spikes in book sales are, in many cases, preceded by spikes in references to these books in the blogspace. They compare bursts of blog posts referring to a given book with spikes in sales according to Amazon, and find that "a sudden increase in blog mentions is a potential predictor of a spike in sales rank." Further, they discover that tracking references to books in blogs substantially improves the ability to predict future sales spikes. Smaller-scale experiments on relating the number of occurrences of movie titles in a small number of movie-review blogs are reported in [291], with good success rates. In Chapter 6, we will extend this type of work, and combine it with sentiment analysis methods.

Rather than passively mining the blogspace for business intelligence, Java et al. [131] propose application of formal influence models to information propagation patterns in the blogspace, to *generate* CGM. This work attempts to locate a

[25] For example, negative commentary about a computer manufacturer that appeared in BuzzMachine, a high-readership blog, has been widely cited in other blogs as well as in mainstream media; see http://buzzmachine.com/archives/cat_dell.html.

[26] http://nielsenbuzzmetrics.com

[27] http://umbrialistens.com

[28] http://cymfony.com

set of influential blogs which, if discussing a certain topic, are likely to maximize the “buzz” around this topic in the rest of the blogspace. From a marketer’s point of view, this set of blogs constitute an important marketing channel.

2.4 Scope of this Work

In the years since they were first introduced, blogs evolved from logs of visited pages to channels in which individuals express themselves, sharing their opinions and views with an audience. The volume and reach of blogs increased greatly, and the blogger profile changed too: moving from a tech-aware crowd to the masses, blogs are currently playing a role in the lives of many people from different backgrounds and locations. It is personal blogs that compose the majority of the blogspace today: a “long tail” of blogs with small audiences, maintained as a hobby by millions of individuals worldwide; these are also the blogs on which this thesis focuses.

Part I

Analytics for Single Blogs

The view of blogs as representing individuals motivates many of the methods developed for analyzing blogs in this thesis. We begin our exploration of text analytics for the blogspace at the level of these individuals, examining one blog at a time, and, sometimes, individual blog posts. There is a person behind each blog, and there is a story behind each post: what can we learn from the text about this person and about the events or thoughts being described? How can we use this knowledge to benefit the blogger and the blog's readers? What services can be provided based on analyzing the contents of a blog?

There are many text analysis directions to follow; the previous chapter listed just some of the main ones in the context of the blogspace. Some of these directions are not particular to blogs: instead, methods which have been successfully applied to various domains are applied also to blog text, e.g., to perform topical categorization or sentiment analysis. In the work presented in this Part we choose to address tasks which we view as specific to the blogspace, either because they involve the distinct language of blogs, utilize blogging features such as tags or comments—or both.

While most computational analysis of blogs is carried out on large collections of blogs—possibly, the entire blogspace—the main aim of this part of the thesis is to demonstrate the type of information that can be mined from individual blogs or posts, and provide useful applications for this information. In particular, this part of the thesis concentrates on three broad text analysis themes, in the context of blogs. Chapter 3 addresses the task of mining knowledge about the blogger: her interests, preferences, and surroundings; extracted information is evaluated in the framework of commercial applications. Chapter 4 applies methods for text classification—itself a mature text analytics area—to tasks which are unique to blogs, including affect analysis and automated tagging. Finally, Chapter 5 uses a content analysis approach to address a form of spam mostly prevalent in the blogspace: comment spam. We view these three high-level tasks as complementary: the first is user-centric, focusing on the individual behind the blog. The second set of tasks is centered on the text rather than its author. The final task combines both a blogger-oriented and text-oriented view to form a practical solution to a problematic area of blog analytics—dealing with spam.

Chapter 3

Language-based Blog Profiles

We begin our exploration of text analysis at the blog level by investigating the "person behind the blog." Our hypothesis is that knowledge about this individual can be mined from the language used in her blog, and that this blogger profile is useful for a range of blog access tasks. The tool we use for profiling a blogger is statistical language modeling: much of the work presented in this chapter is based on applications of language models to text mining, and some of these approaches appear also in subsequent chapters. In particular, we report on two successful applications of language models in the blogspace, both related to marketing products and services to bloggers and their audience; but first, we introduce language models and their usage in text analytics.

3.1 Statistical Language Modeling

As their name suggests, statistical language models attempt to capture regularities of natural language phenomena using statistical methods. Shannon used them to predict the character-level complexity of the English language, measuring the degree to which it can be compressed [271]; in the 1970s and 1980s they have been applied to speech recognition, to predict the most likely next word given a sequence of encountered words. In the 1990s and 2000s they have successfully been adopted by researchers in other areas such as information retrieval [243] and machine translation [44], both for "next word prediction" and for the related problem of estimating the likelihood of a sequence of words in a language.

At its core, a language model is a probability distribution over sets of strings: the probability assigned to a string is the likelihood of generating it by a given language, or observing it in text generated by the language. A reasonable language model of everyday English will assign a higher probability to the string "the" than to the string "book," as the former is more frequently-occurring in the language. Similarly, the sentence "I have a book" will be assigned a higher probability than the less likely "I have a cook," which will in turn have a higher probability than

"Cook a have I"—a string which is very unlikely to appear in everyday English text. Typically, the real model of a language is unknown; instead, it is estimated from a representative spoken or written sample of that language. There are various ways of estimating models (cf. [256]); we focus on the most commonly-used approach: n-gram based language models. In these models, probabilities are assigned to every word n-gram in a language for a given n: *unigram* models assign probabilities to single words, *bigram* models to pairs of words, *trigram* models to word triplets, and so on.

The core idea of n-gram language models is that the likelihood of a word depends only on its immediate context, the $n-1$ words before it (the Markov independence assumption [135]). This greatly simplifies the resulting models compared to "complete" models: instead of estimating a probability for any string in the language, direct estimation is only performed for strings which are exactly word n-grams, and probabilities of longer strings are estimated using the chain rule, using the probabilities of the n-grams appearing in them. For example, the probability of the bigram $(w_1\ w_2)$ would be estimated, using a unigram model, as $p(w_1\ w_2) = p(w_1) \cdot p(w_2)$, and the probability of the trigram $(w_1\ w_2\ w_3)$ would be estimated, using a bigram model, as $p(w_1\ w_2\ w_3) = p(w_1) \cdot p(w_1\ w_2) \cdot p(w_2\ w_3)$.

The order of the language model—that is, the size of n—controls the amount of context encoded within the model. The complexity of a model increases exponentially with n, as the possible number of n-grams in a text with vocabulary V—the n-grams which will be assigned probabilities directly—is $|V|^n$. While some degree of context is crucial to some applications—most notably, speech recognition which typically employs trigrams or even higher-order models, in other domains—such as information retrieval—the improvement gained by using these more complex models over the performance of unigram models is minor if any [282].

Building n-gram models. A straightforward approach to estimating an n-gram language model is a maximum likelihood estimate (MLE, [135]). In this approach, the probability assigned to an n-gram $(w_1\ \ldots\ w_n)$ is simply its frequency in the text, normalized by the number of possible $(n-1)$-grams which share the same prefix, $(w_1\ \ldots\ w_{n-1})$:

$$p_{\mathrm{MLE}}(w_n|w_1, \ldots, w_{n-1}) = \frac{\mathrm{count}(w_1\ \ldots\ w_n)}{\mathrm{count}(w_1\ \ldots\ w_{n-1})}$$

Models estimated this way quickly encounter the *zero frequency* problem: any n-gram not appearing in the representative sample of text with which the model is estimated will be assigned a probability of zero. In unbounded domains such as natural language this is problematic: the representative sample is unlikely to contain all n-grams that will appear in other texts generated by the same language. For example, Zhu and Rosenfeld report that even after using 103 million words to construct a model, 40% of the trigrams in unseen text from the same domain were not included in it [327]. Worse yet, the multiplicative chain rule implies that a single n-gram with a zero probability will cause the probability of any string containing it to be zero as well.

To address the zero frequency problem, a process of flattening the probability distribution of the language model called *smoothing* is used. Smoothed language models assign non-zero probabilities even to n-grams that do not appear in the sample text, while n-grams that do appear in the text are assigned a lower probability than their MLE probability. Various smoothing techniques have been proposed [50]. *Discounting* approaches target the n-gram counts directly: by decreasing the count of observed n-grams and increasing the count of unobserved ones, probability mass is transferred from the n-grams appearing in the sample text to those not appearing in it. A widely-used discounting method is based on the Good-Turing Estimate [82], which sets the total probability mass to be redistributed among unseen events to the same mass assigned to all events which occur with frequency 1; these, in turn, receive the total probability mass assigned to events with frequency 2; and so on. Additional approaches often used to smooth language models include *backoff* and *interpolation* models, in which evidence from different sources is combined to estimate the probability of an n-gram. Such sources may include lower-order n-grams (i.e., an unseen bigram is assigned a probability derived from the probabilities of the unigrams contained in it), or a larger, less domain-specific text sample (i.e., the entire English language). The main difference between backoff and interpolation is whether the additional evidence is used only in case of unseen n-grams (backoff), or also for n-grams with non-zero probabilities (interpolation). Commonly used are the interpolation method proposed by Jelinek-Mercer, and Katz smoothing—a backoff method; these methods and others are surveyed in [50].

Evaluating language models. A task-oriented approach to evaluation of language models is to compare the performance of different models on the task they were designed for (e.g., speech recognition or document classification). However, as this is a time-consuming task which is also dependent on performance of other components, a task-independent metric called *perplexity* is often used. As mentioned in Chapter 2, perplexity is, broadly speaking, the average number of words that may follow a given word in a language, given the word's context.[1] The perplexity of a model built from a representative sample of text (a training set) measures the extent to which it predicts an unseen sample from the same text (a test set), giving an indication of the usefulness of that language model [43]. The closer the probability distribution of the model is to that of the test set distribution, the lower perplexity the model has, and a language model with a low perplexity value is a better approximation of the real model of the language than

[1] Perplexity is itself based on the information-theoretic notion of entropy, and a deeper analysis of it is out of the scope of thesis; a good overview is given in [135].

a higher-perplexity one.

A different use for perplexity is evaluating corpus diversity or complexity. By splitting a text to a training used to build a model and a test set used to measure perplexity (possibly, repeating this process with different splits and averaging the results), an indication of how complex it is to model a given text is obtained. Low perplexity values measured this way indicate that the text is "easier to model," i.e., more regular. When constructing language models of different corpora, their perplexities can be compared to measure their relative complexity. General-domain texts have higher perplexities than domain-specific ones; returning to the example given in the discussion of the language of blogs in Section 2.2.3, the perplexity of the general-English Brown corpus is 247, and that of a collection of medical texts is 60.

3.2 Blogger Profiles

After introducing language models, we turn to a specific application of them for text mining: extraction of keywords from text to form a term-based profile of it.

3.2.1 Language Model Comparison and Keyword Extraction

Language models are probability distributions, and as such, standard measures of similarity of distributions can be applied to calculate the distance between two models. A standard measure for similarity between distributions is the Kullback-Leibler (KL) divergence which has been mentioned in Section 2.2.3; it is also called relative entropy. Formally, the KL-divergence between two probability distributions Θ_1, Θ_2 is

$$\mathrm{KL}(\Theta_1 \| \Theta_2) = \sum_x p(x|\Theta_1) \log \frac{p(x|\Theta_1)}{p(x|\Theta_2)}$$

where $p(x|\Theta_i)$ is the probability of the event x in the model Θ_i. Strictly speaking, KL-divergence is not a distance, since it is not symmetric; it is, however, widely used as a measure of similarity between distributions, particularly between a "true" distribution and an arbitrary one which approximates it [56]. From an information-theoretic point of view, KL-divergence measures the amount of information wasted by encoding events from one distribution using another one. When comparing language models, the events are occurrences of n-grams, and KL-divergence measures how easy it is to describe text generated by one language using the model of another.

KL-divergence answers the question "how different are two language models." This is useful for estimating the similarity of different languages (as has been done in the previous Chapter, comparing the language of blogs with various English genres), or the similarity of two texts (by building language models for each, and comparing the models—in the same approach used for application of language modeling in information retrieval systems, e.g., [115]). However, it does not provide details as to what this difference consists of: what text is likely to be observed in one language and not in the other.

One possible answer to this last question can be found by comparing the individual frequencies of n-grams in two languages, rather than aggregating the differences to a single measure. Several goodness-of-fit tests can be used for this; a good survey is given in [142]. In practice, one of the most stable approaches is the *log likelihood* test, which is useful when the probability distributions compared contain many rare events [73]—as is often the case in language models, particularly those estimated from relatively small amounts of text. Given two language models, the log likelihood test assigns every n-gram in them a divergence value indicating how different its likelihood is between the two languages: words with high log likelihood values are more typical of one language than the other, and words with low values tend to be observed in both languages with similar rates.

Formally, the log likelihood test for comparing the frequencies of an n-gram x in two corpora C_1, C_2 is calculated as follows:

$$\mathrm{LL} = 2 \cdot \left(\mathrm{c}(x, C_1) \log \frac{\mathrm{c}(x, C_1)}{E_1} + \mathrm{c}(x, C_2) \log \frac{\mathrm{c}(x, C_2)}{E_2} \right),$$

where $\mathrm{c}(x, C_i)$ is the raw frequency of x in C_i, and $E_i = \frac{|C_i| \cdot (\mathrm{c}(x,C_1)+\mathrm{c}(x,C_2))}{|C_1|+|C_2|}$.

The divergence values of individual n-grams between language models provide an elegant way of extracting keywords from a given document (or set of documents) taken from a larger collection. First, language models are constructed both for the given document (or set of documents) and for the entire collection. Then, these two models are compared. Ordering the terms of the models according to the divergence values assigned to them functions as a profile of the document. Prominent terms in the profile—terms with higher divergence values—are more "indicative" of the content of the document, as their usage in it is higher than their usage in the rest of the documents. For example, according to this method, the most indicative terms for this chapter, compared with other chapters of the thesis, are (in order of importance) "model," "ad," "language," and "profile"—terms which indeed appear to capture the main topics discussed (the word "ad" is prominent in subsequent sections of this chapter).

3.2.2 Profiling Blogs

Applying a language-model based keyword extraction mechanism to profile blogs is a straightforward process: create a language model for the blog by aggregating all its posts, and compare it with a language model of a large collection of blogs using a method such as log-likelihood. Figure 3.1 shows an example of the top indicative n-grams found this way in a specific blog, "Challies,"[2] demonstrating the possible insight given by the profile about a blogger: without reading the blog, it can be said with high likelihood that Christianity and religion are central topics of interest of the person behind this blog; reading the blog posts confirms this.

god
christ
bible
passion christ
scripture
review
jesus
mary
sabbath
church
jesus christ
purpose

Figure 3.1: Top n-grams from a profile of a blog, based on log-likelihood.

In the remainder of this chapter, we utilize these blog-induced profiles for real-life applications in the blogspace. First, we conduct a small-scale experiment to test the usefulness of profiles, by evaluating whether they can be used to determine the blogger's commercial taste. We then extend this framework by developing more complex models that can be applied to single blog posts, and by evaluating their usage in the more involved task of contextual advertising in personal blogs.

3.3 Matching Bloggers and Products

An immediate application of blogger profiling is identifying, using these profiles, the cultural preferences of a blogger: products or services she is likely to appreciate, political views which correlate with her opinions, and so on. In this section we address this task, choosing book preferences as the cultural bias we try to predict from the blogger's profile. Rather than actually identifying books that match the blogger's profile, we seek to discover the *categories* of books a blogger is most likely interested in. These categories enable advertisers and vendors to custom-tailor ads and offers to the blogger and her readers, e.g., by offering discounts on best-sellers or latest releases from these categories.

The task, then, is to generate a list of suggestions of categories of books the blogger is likely to purchase, given the text of her blog. Our approach to this task is as follows. First, we construct blogger profiles with simple, unigram language models. The next stage involves distilling book categories from these profiles; for this, we consult a large repository of categorized books, aggregating the categories of books matching the blogger's interests as reflected in her profile. The result is a "book category profile"—the product profile of a blogger, a ranked list of

[2] http://www.challies.com

product categories she is likely to purchase. To evaluate how well this list reflects the true wishes of a blogger, we apply it to blogs containing explicit lists of "desired books," comparing the predicted categories with the blogger-given ones. As a baseline, we construct similar book category lists, not from the blogger language model-base profile, but from explicit references to books in her blog; we show that the language model approach substantially outperforms this. We follow with additional details about the process.

Interest Indicators

In general, to effectively identify the interests of a blogger, we are searching for indications of these interests in the blog. Indications may be explicit: some blogging platforms (e.g., Yahoo 360, Blogger.com) allow bloggers to list their interests in their blogger profile. For the many bloggers who do not utilize this option, or who do not use a platform supporting this, we can search for implicit indications of interest. We use two methods to do this: one attempting to identify explicit references to books in the blog, and the other based on the language modeling approach we have just described. As mentioned earlier, the first approach serves as a baseline for evaluating the language model-based one.

Identification of names of books (and other products) in text is known to be difficult, even on a closed-domain corpus predominantly dealing with products [241]—and more so on an open-domain text such as that found in blogs. Our approach to this is a simple one: we tag the blog text with a named entity tagger, and employ heuristics on the results to identify possible book titles. Heuristics include locating entities that appear in close proximity to a small set of book related keywords ("read," "book"); matching patterns such as "⟨ENTITY⟩ by ⟨PERSON⟩;" identifying links to product pages on retailers such as Amazon.com, and so on. Extracted entities are scored based on a combination of their recurrence in the blog, their NE-tagger confidence score, and a score derived from the heuristic used to select them; the top-scoring entities are used to populate the list of books referenced by the blog, and serve as the interest indicators of the blogger. Our approach is precision-oriented, rather than recall-oriented; an examination of the top-scoring entities extracted shows that the majority of them are indeed book titles.

The other approach to finding interest indicators in blogs—the one we are evaluating—consists of identifying the indicative words of the blog using profiling as described earlier, and results in a set of keywords such as those appearing in Figure 3.1.

From Indicators to Book Categories

Once a set of indicators is found, we proceed by deriving book categories matching the indicators; this is done by first retrieving categorized books that match the indicators, then aggregating their categories into a complete profile. To retrieve a list of categorized books we utilize a large collection of books manually categorized by a retailer—in our case, the Amazon.com catalog, which contains millions of categorized titles. Given an indicator—a possible book name or an extracted keyword—we search Amazon for the top books related to this indicator. The categories of the returned results are aggregated and sorted by their frequency in the results.

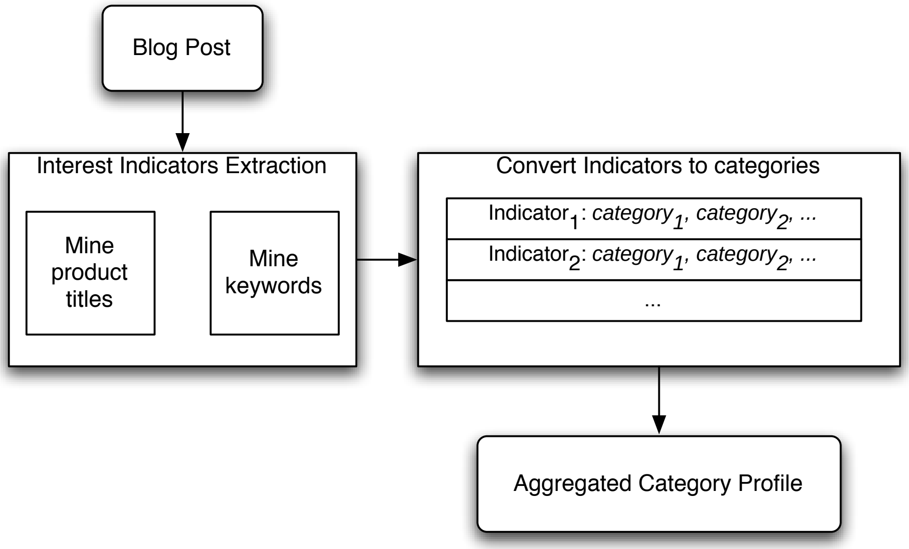

Blog Post
Interest Indicators Extraction
Mine product titles
Mine keywords
Convert Indicators to categories
Indicator$_1$: *category$_1$*, *category$_2$*, ...
Indicator$_2$: *category$_1$*, *category$_2$*, ...
...
Aggregated Category Profile

Figure 3.2: Book category profile construction.

The entire process is summarized in Figure 3.2. First, indicators (product titles or keywords) are mined from the text of a blog. Each indicator is used to obtain a list of book categories by matching it with categorized books from the Amazon catalog. Finally, the lists generated by the different indicators are aggregated to a single, weighted profile.

A Worked Example

We demonstrate the process of constructing the models on a particular blog: "Guided By Echoes."[3] Table 3.1 shows the top-ranking book references extracted from the blog text, and the top categories associated with the generated queries to Amazon. Numbers in parenthesis are the total number of products with the listed category over all results. Note that the extraction contains noise, for example, the (misspelled) band "Roger Clyne and the Peacemakers" which was extracted as a book name. The effect of this on the final result is diminished by the fact that we only query a list of categorized books (in practice, we query the Amazon.com book collection only): in this particular example, no book results are found.

[3] http://guidedbyechoes.livejournal.com.

Possible book titles in the blog text	All My Children Supergirl and the Legion of Super Heroes Golden Age Superman and Earth Roger Cline and the Peacemakers The Bible . . .
Relevant Amazon books	*The Official All My Children Trivia Book* (Television, Performing Arts) *Supergirl and the Legion of Super Heroes* (Comics, Graphic Novels, Superheroes, Juvenile Fiction) . . .
Derived product profile	Superheroes (14) Economics (13) Business (13) Juvenile Fiction (11) . . .

Table 3.1: Sample profile based on product extraction.

Extracted keywords	wolverine replica discretion hulk pencils . . .
Relevant Amazon books	*Wolverine: Origin* (Comics, Graphic Novels, Superheroes, Marvel, Fantasy) *The Chaos Engine : Book 1 (X-Men: Doctor Doom)* (X-Men, Parenting & Families, Fantasy, Science Fiction) . . .
Derived product profile	Superheroes (46) Graphic Novels (39) Fantasy (38) Economics (37) . . .

Table 3.2: Sample profile based on keyword extraction.

Similarly, Table 3.2 shows the keywords extracted from the blog, the top books returned by Amazon for queries containing these words, and the generated model.

3.3.1 Evaluation

The main question we face at this stage is whether the keyword extraction approach, combined with the usage of an external catalog to map between keywords and products, provides a better understanding of the type of products a blogger is interested in. In particular, we want to find out whether this approach yields better results than identifying explicit references to products in the blogger's writings—our baseline. To evaluate our proposal, we require a set of bloggers for which the (book) purchase profile is known. This information is typically difficult to obtain for various reasons, most notably privacy issues. However, the tendency of bloggers to expose some aspects of their lives aids us in this case. Many bloggers maintain *wishlists*, lists of desired products; Amazon, for example, provides a special mechanism for creating such wishlists which can easily be linked from a blog. We use these wishlists to obtain ground truth regarding the blogger's book preferences: the categories of books listed in a blogger's wishlist are taken to be the real categories we attempt to derive.

Data

We obtained a set of 400 random blogs containing a link to Amazon wishlists. Non-English blogs and blogs with a small amount of text (less than 200KB, after stripping HTML and template-like text), or with fewer than 30 books in the wishlist were discarded, leaving 91 blogs with, on average, 1.1MB of text each. Wishlists were parsed in the same manner used to parse Amazon's results in the model construction phase: categories of books were aggregated to build a weighted list of the blogger's declared commercial interests, functioning as a golden standard. Table 3.3 shows this golden standard, as built for the blog used as a working example in the previous section.

Experiments and Results

Next, the methods for building advice models were employed, resulting in two models per blog: based on products and based on keywords. Both these models and the the "ground truth" model are built using the same Amazon categorization scheme, so the resulting categories are comparable. To measure their similarity, we use the overlap in the top-3 categories of both models. If two of the categories appearing in the top-3 model built by a method appear also in the golden model, the overlap is 2/3, and so on: in the example in Table 3.3 the overlap is 1/3 with both of the constructed models. The average overlap over all blogs was 0.142 for the product-based baseline method, and 0.311 for the keyword-based

Wishlist	amazon.com/gp/registry/17G9XYDK5GEGG
Books in wishlist	*The Big Book of Conspiracies* (Comic books, Conspiracies, Controversial Knowledge, . . .) *Buffy the Vampire Slayer: Origin* (Young Adult, Comics, Humor, Juvenile Fiction) . . .
Blogger product profile	Games (61) Role Playing (45) Superheroes (42) Comics (42) . . .

Table 3.3: A sample wishlist profile.

method; experimenting with combinations of the methods did not yield additional improvements.

Failure Analysis

An examination of failures—blogs for which little or no overlap exists between the models—shows that the majority of them are diary-like, highly personal blogs, with little topical substance. Often, this is computationally discernible: e.g., the keyword extraction phase for these blogs results in short lists, since only a small number of nouns exceed the minimal log-likelihood value to be considered "distinctive." A possible extension to our method would be to identify these cases and assign confidence values to the generated models.

Additionally, in the experiments reported here we did not take into account the hierarchical structure of Amazon's categorization scheme; doing so would have resulted in higher scores—e.g., in the example, the category "Graphic Novels" is a parent category of the ground truth category "Superheroes."

3.3.2 Conclusions

These initial results are encouraging: given the simplicity of our keyword method, it performs fairly well, correctly identifying about a third of the categories the blogger is most interested in, out of a large hierarchy of hundreds of different categories. Certainly, part of the improved performance, compared with a book name extraction approach, is the difficulty in correctly identifying books in blogs and the relative infrequent explicit references to them—keywords can be easily identified in any blog with enough content.

The experiments reported in this section are of a limited scale, and serve mostly to demonstrate that the approach of relating bloggers to products through analysis of their text is worth pursuing. In the next section we develop this approach more thoroughly and demonstrate its success on a different task in the area of matching bloggers and products or services.

3.4 Extended Blogger Models

The task of relating an individual to commercial preferences through a blog introduced in the previous section is a novel one; but the language modeling approach used to build this commercial profile is well-established, and not particular to blogs. Similar profiles can be generated from any text which, like blogs, contains references to personal experiences and preferences; email and postings to newsgroups are examples.

In this section we extend the process of generating (commercially-oriented) profiles, this time utilizing properties which are unique to blogs—their semi-structured format, the meta-data associated with blog posts, and knowledge of connectivity between posts. These profiles are particularly useful for tasks which require profiling at the single post level rather than on the level of a complete blog. Often, the text of a single post does not contain enough text to estimate a useful language model for it, or (especially for personal journals) is topically unfocused; using the various properties mentioned earlier, we attempt to bypass this problem and enrich the model with additional sources. This section describes the construction of these more complex profiles; the next one evaluates them using a task which drives much of the internet's growth in recent years—online advertising.

3.4.1 Language Models of Blog Posts

Our approach to creating profiles at the post level is based again on divergence measures for language models, which enable identification of indicative words in a text. The key difference is that we analyze different sources of information for a given post. Clearly, the content of the post itself is an important source of information. But other sources of information about a specific blog post are contents of other posts from the same blog, contents of comments posted to it, contents of posts linking to it, and so on; the community-oriented structure and the temporal nature of blogs supply yet more sources of knowledge. Our approach, then, attempts to compose a profile of a post by combining the information present in each of these representations.

Concretely, given a blog post p, we construct the following separate models, each built as described in Section 3.2, by comparing the language used in text associated with the model with the text of a larger, more general model (a large subset of the blogspace, in our case):

Post Model. For this model we use the most straightforward content: the contents of the blog post p itself.

Blog Model. This model is built from all posts from the same blog as p which are dated earlier than p. The intuition behind this is that interests and characteristics of a blogger are likely to recur over multiple posts, so even if p itself is sparse, they can be picked up from other writings of the blogger. Only posts earlier than p are used to conform to real-life settings: in an actual deployment of an online profiling system such as the one we develop later in this section, a post is analyzed shortly after it was written and future posts from the same blogger are not known yet.

Comment Model. This is the first model exploiting properties specific to blogs. The comments posted in response to p are likely to be directly related to the post; this model is based on their aggregate contents.

Category Model. As mentioned in Chapter 2, tags are short textual labels that many bloggers use to categorize their posts; they range from high-level topics ("sport," "politics") to specific ones ("Larry's birthday," "Lord of the Rings"). For this model, we used all blog posts filed under the same category as p, as the bloggers themselves decided that they share the same topics.

Community Model. The identification of communities of blogs is beyond the scope of this thesis; much work has been dedicated to this, as surveyed in the previous chapter. We take a simple approach to community identification, marking a blog as belonging to the community of p's blog if it links at least twice to that blog. The text of all blogs in the community of p determined this way is aggregated to create the community model.

Similar Post Model. For this model, we use the contents of blog posts which are most similar to p, out of a large collection of blog posts. This is a standard method of enriching text and is used in various applications dealing with extraction of terms from text, e.g., query expansion [21]. The way in which similar posts were selected is detailed later in this section, when we describe an instantiation of this model for a real-life task.

Time Model. Many blog posts contain references to ongoing events at the time of their writing: mentions of news items, holidays, new products, and so on. To accommodate this, we construct a model based on all blog posts published during a window of time around the publication time of p, capturing events that influence a large number of bloggers.

Each model, then, provides terms that reflect a particular aspect of a blog post: its content, author, the events at the time of its writing, and so on. These terms function as indicators of the interests related to the blog post—much as the indicators in the previous section, which were used to identify book interests. The next stage is to combine these different aspects to a single representation.

3.4.2 Model Mixtures

Forming combinations of different language models is a common practice when applying these models to real-life tasks. While finding an optimal mixture is a complex task [163], there are methods of estimating good mixtures [163, 136]. In our case, we are not combining pure language models, but rather lists of terms derived from language models. As with most model mixtures, we take a linear combination approach: the combined weight of a term t is

$$w(t) = \sum_i \lambda_i \cdot w_i(t),$$

where λ_i is the weight assigned to model i and $w_i(t)$ is the weight assigned to the term i by model i. Two general approaches are usually used to estimate λ_i, the model weights: static and on-line methods [136]. Static weights are fixed weights for each model: weights which are determined a-priori by an estimation of the model's typical importance. On-line methods derive posterior weights for each model, based on its expected contribution to the combination, a contribution that may vary according to the model's properties and the other models in the combination. We use both static and on-line weights for our combination.

Static weights. Clearly, the contribution of each of the different models is not equal; for example, the model representing the blogger herself is arguably more important than the one representing her community. Optimal prior weights can be estimated for each model in the presence of training material. We mark the static weight assigned to model i as λ_i^s.

On-line weights. These weights are aimed at capturing the relative importance each model should have, compared to other models induced by a particular blog post. In our setup, we associate this importance with the quality of the model—better formed models are given a higher weight. Our models consist of lists of terms; one way to evaluate the quality of such a list is to check its coherency: the degree to which the terms in the list are related, an approach often used in evaluation of text clustering methods. To measure coherency, we calculate the pointwise mutual information (PMI, [183]) between any two terms in the list, and take the mean of these values as the coherence of the list.

PMI is a measure of the degree of statistical dependence between two events (terms, in our case) and is defined as

$$\text{PMI}(t_1, t_2) = \log \frac{p(t_1 \& t_2)}{p(t_1)p(t_2)}$$

where $p(x)$, the probability assigned to event x, corresponds in our case to the probability of observing the term x. PMI-IR [296] uses information retrieval to estimate these probabilities using document frequencies in a large corpus such as the web itself. The measure thus becomes

$$\text{PMI-IR}(t_1, t_2) = \log \frac{\text{df}(t_1 \& t_2)}{\text{df}(t_1) \cdot \text{df}(t_2)}$$

where $\text{df}(t_i)$ is the number of documents containing the term t_i (in the case of the web—obtained using search engine hit-counts) and $\text{df}(t_1 \& t_2)$ is the number of documents containing both terms within the same window of text, or, in the absence of term proximity information, in the entire document.

Returning to our model, we define the coherency of the set of terms $T_i = \{t_1, t_2, \ldots, t_n\}$ included in model i as

$$\text{coherency}(T_i) = \frac{1}{|T_i|} \cdot \sum_{\{j,k\} \in T_i, j \neq k} \text{PMI-IR}(t_j, t_k).$$

The on-line weights obtained this way are denoted as λ_i^o.

The combined weight assigned to model i is $\lambda_i = \lambda_i^s \cdot \lambda_i^o$; the final term weight for term t, used to construct the post profile, becomes

$$w(t) = \sum_i \lambda_i^s \cdot \lambda_i^o \cdot w_i(t)$$

Note that terms may appear in multiple models, boosting their final weight in the combined model. Once again, in the presence of training material this combination can be replaced with a more informed one, e.g., a linear combination with optimized weights.

3.4.3 Example

The profile constructed for a blog post, like the models used to create it, consists of a weighted list of terms, where the weight of each term identifies its importance in the profile. Table 3.4 shows an example of the different models constructed for a given blog post, their weights, and the resulting combined model.[4] The post itself deals with birds visiting the blogger's garden, and this is reflected in the post model. Additional models, in particular (in this case) the community and category ones, expand the profile, showing that the blogger's interests (and, hence, the interests of visitors to the blog) can be generalized to nature and related areas.

[4] The blog post is taken from a corpus which is described in the next section. All the experimental data in this section and the next one, including this example, is in Dutch and was translated into English for convenience.

Permalink	http://alchemilla.web-log.nl/log/4549331
Date	January 4th, 2006
Post	*Life in the Garden* Birds are flying around the tree and the garden behind our house. . . Hopping blackbirds, a few red-breasts, some fierce starlings and, surprisingly, a few Flemish jays. I thought Flemish jays live in the forest. I haven't heard the trouble-making magpies from the neighbors for a couple of days, they must have joined the neighbors for their winter vacation :) I see now . . .
Post terms	garden, spot, starlings, blackbirds, (feeding)-balls
Blog terms	nature, bird, moon, black, hats, singing, fly, area
Comment terms	jays, hydra
Category terms	bird, moon, arise, daily
Community terms	nursery, ant, music, help, load, care
Similar-post terms	birds, garden, jays, blackbirds, Flemish, red-breasts
Time terms	(none)
Model weights	Post:0.63, Blog:0.21, Comment:0.02, Category:0.05, Similar-posts:0.09 Time:0
Combined model	birds, spot, garden, jays, blackbirds, nature . . .

Table 3.4: Example of terms generated by different blog models.

3.5 Profile-based Contextual Advertising

We now put these extended blog post profiles to the test, by using them to improve contextual advertising in journal-type blogs.

Contextual advertising (also called content-based advertising) is a form of online advertising in which advertisements are shown on a web page based on its content, to increase the likelihood of their relevance to users accessing the page [252]. Current contextual advertising platforms are designed for "topical" web pages—those that are mostly centered on a certain topic, and are often designed and maintained by professionals. Applying the ad-matching methods used by these platforms to user-generated content such as blogs is problematic: as we have seen, blog posts—in particular, those belonging to personal-journal blogs—typically contain non-topical, unfocused content, for which it is difficult to directly match an ad. We view content-based advertising in personal blogs as an ideal test case for the models we developed in the previous section, models that aim to capture more than the contents of the blog post alone.

There are two issues addressed in this section. First, we evaluate the effectiveness of contextual ad placement methods developed for general (non-blog) web pages, when applied to personal blogs; we show that the performance in this domain is below that achieved for non-blog pages. Second, we develop an ad placement method based on the extended profiles we have just described, evaluate it, and show that it substantially improves over state-of-the-art, while maintaining similar computational requirements.

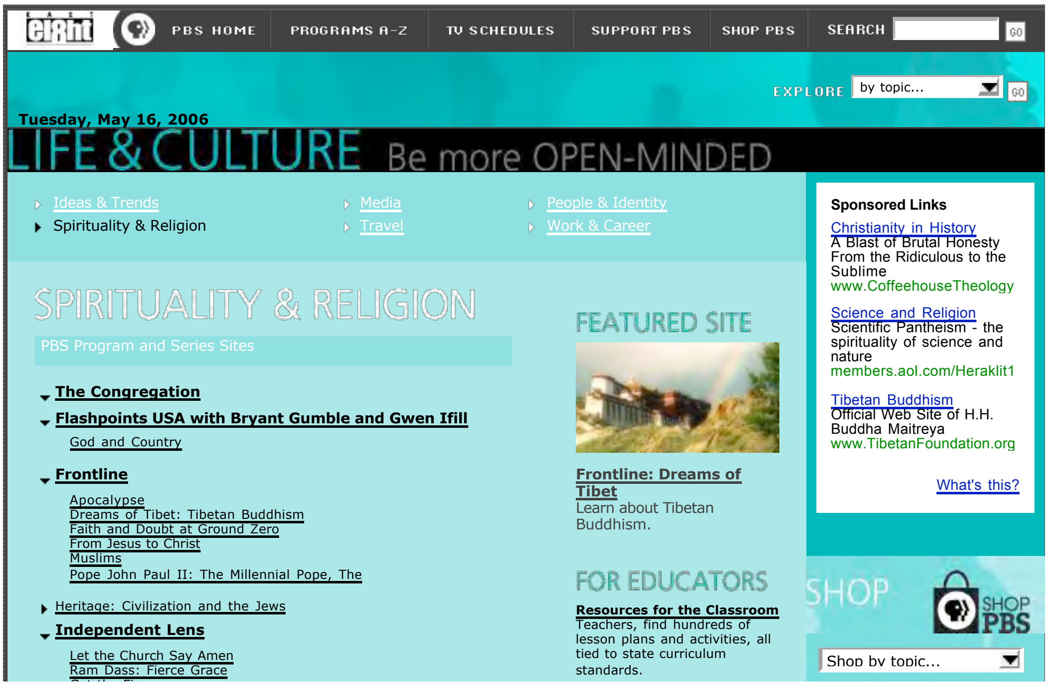
3.5.1 Contextual Advertising

First deployed in 2003, contextual ad placement services allow websites to pay to have their advertisements displayed alongside the contents of related web pages. Programs such as Google's AdSense, Yahoo's Publisher Network, and Microsoft's adCenter ContentAds have become very effective in generating revenue both for the advertiser and the ad-matching mediator by associating the content of a web page with the content of the displayed ads, increasing the likelihood of their usefulness. Often, the ads are non-intrusive and are clearly marked as such; on top of that, they enjoy the reputation of the ad selection platform (which is typically a well-known web search engine). Figure 3.3 shows an example of contextual advertising: in this example, Google's AdSense program is used to display ads related to spirituality and religion, alongside the listing of TV programs offered by the American Public Broadcasting Service in this field.[5]

As contextual ad placement has become a substantial source of revenue supporting the web today, investments in this task, and more specifically, in the quality of ad placement, are increasingly important. Most of the advertisements are currently placed by search engines; advertisements that are not relevant may negatively impact the search engine's credibility and, ultimately, market share [305, 33].

Since the area of content-based ad placement is relatively new, and as it involves many "trade secrets," the amount of existing published work is limited. The work most closely related to that we describe here is that of Ribeiro-Neto et al. [252], involving a lexicon expansion technique for contextual ad placement. This approach uses a variety of information sources, including the text of the advertisements, the destination web page of the ad, and the triggering words tied to a particular ad; the main task is to bridge the possible vocabulary gap between an ad and the web page on which it should be placed. Later work by the same authors [161] applies a genetic algorithm framework to ad matching, improving performance in the domain of news web pages. Work on ad placement prior to [252] was of a more restricted nature. E.g., Langheinrich et al. [162] propose a

[5] Example taken from `http://www.pbs.org/life/life_spirituality.html`



Tuesday, May 16, 2006

LIFE & CULTURE Be more OPEN-MINDED

Ideas & Trends
Spirituality & Religion
Media
Travel
People & Identity
Work & Career

SPIRITUALITY & RELIGION

PBS Program and Series Sites

The Congregation

Flashpoints USA with Bryant Gumble and Gwen Ifill

God and Country

Frontline

Apocalypse
Dreams of Tibet: Tibetan Buddhism
Faith and Doubt at Ground Zero
From Jesus to Christ
Muslims
Pope John Paul II: The Millennial Pope, The

Heritage: Civilization and the Jews

Independent Lens

Let the Church Say Amen
Ram Dass: Fierce Grace

FEATURED SITE

Frontline: Dreams of Tibet
Learn about Tibetan Buddhism.

FOR EDUCATORS

Resources for the Classroom
Teachers, find hundreds of lesson plans and activities, all tied to state curriculum standards.

Sponsored Links

Christianity in History
A Blast of Brutal Honesty From the Ridiculous to the Sublime
www.CoffeehouseTheology

Science and Religion
Scientific Pantheism - the spirituality of science and nature
members.aol.com/Heraklit1

Tibetan Buddhism
Official Web Site of H.H. Buddha Maitreya
www.TibetanFoundation.org

What's this?

Figure 3.3: Contextual advertisements on a web page (in a sidebar to the right).

system that is able to adapt online advertisements to a user's short-term interests; it does not directly use the content of the page viewed by the user, but relies on search keywords supplied by the user to search engines and on the URL of the page requested by the user. Tau-Wih et al. [288] report not on matching advertisements to web pages, but on the related task of extracting keywords from web pages for advertisement targeting. The authors use various features, ranging from *tf* and *idf* scores of potential keywords to frequency information from search engine log files. An important observation made in this work is that an improvement of $X\%$ in ad-matching can lead to an improvement of $X\%$ in the end result (in this case, sales from advertisements), unlike many other text analytics tasks where the effect of performance enhancements on the end result is not linear. This makes work on quality of ad-matching particularly beneficial.

The work presented here was, at the time of making it publicly available, one of the first studies of contextual advertising, and the first tailored to a specific domain in which advertising is more complicated—the blogspace. We will show that our approach compares favorably with state-of-the-art, while maintaining similar efficiency as known methods.

3.5.2 Matching Advertisements to Blog Posts

Contextual placement of text advertisements boils down to matching the text of the ad to the information supplied in a web page. Typically, a textual ad is composed of a few components: the self-explanatory *title*, designed to capture the attention of the viewer, a short *description* providing additional details, a *URL*, the target a user will be taken to if the ad is clicked, and a set of *triggering terms*. The triggering terms, which are not displayed to the web page viewer, are provided by the advertisers and function as terms associated with the ads, assisting the process of matching ads with context. We follow the approach taken in [252], which concatenates the text of all these different components to a single textual representation of the advertisement. The challenge we are facing is to select ads (out of a collection of ads represented in this concatenated manner) that are most likely to be of interest to readers of a given blog post.

The post itself is represented in our approach using the mixture-model profiling mechanism we have described in the previous section. To match this profile with advertisements, we again follow [252], which take an information retrieval approach to the task: advertisements are considered a document collection, and a query is used to rank them; in our case, the query is the top-indicative terms of the constructed profile. The ranking scheme we used for the retrieval is a language modeling-based one, which has shown to achieve same-or-better scores as top-performing retrieval algorithms [115].

End-to-end, the ad selection process for a given blog post p proceeds as follows:

1. First, we build the different language models relating to various aspects of p.
2. Next, a profile is constructed from each model by ranking the terms in it according to their divergence values from a model of a large collection of blog posts.
3. The top-diverging terms of each model are combined to form a single weighted list using a linear combination of the models; weights for the combination are determined both by prior knowledge of the relative importance of each model type, and by examining the coherency of each set of top-indicative words being combined.
4. Finally, a query consisting of the top terms in the combined model is used to rank all advertisements; the top-ranking ads are shown to the user.

Table 3.5 shows the selected ads chosen from a large collection of advertisements using this approach, for the post used as an example in Table 3.4 (the details of the collection are given in the next section). Earlier, we observed that the interests of the blogger (and, presumably, her readers) were identified to be related to nature; matched ads are indeed relevant to this area.

3.5.3 Evaluation

We now describe the experiments conducted to evaluate our ad placement method and the results obtained. With these experiments we address two research ques-

Combined model	birds, spot, garden, jays, blackbirds, nature . . .

Selected ads	Interested in working in nature protection and environment? Click on StepStone.
	`www.stepstone.nl`
	Directplant delivers direct from the nursery. This means good quality for a low price.
	`directplant.nl`
	eBay - the global marketplace for buying and selling furniture and decorations for your pets and your garden.
	`www.ebay.nl`

Table 3.5: Example of ad-matching.

tions: first, we are interested in evaluating the effectiveness of state-of-the-art ad placement methods when applied to blogs, a less focused and more demanding domain. Our second aim is to test whether ad placement for personal blogs can be improved using the approach outlined earlier, which takes into account the content of multiple aspects of a blog post.

Blog Corpus. The blog corpus we use consists of 367,000 blog posts from 36,000 different blogs, hosted by `web-log.nl`, the largest Dutch blogging platform, which specializes in personal journal blogs. For comparison, the content of the blogs in our collection is similar to the content of typical LiveJournal or MySpace blogs: highly personal content, where the average blogger is a teenager or young adult expressing thoughts and opinions about daily events. The collection consists of all entries posted to these blogs during the first 6 weeks of 2006, and contains 64 million words and 440MB of text. In addition to the blog posts, the corpus includes comments posted in response to the posts—a total of 1.5 million comments, 35 million words, and 320MB of text. The vast majority of text is in Dutch, with a small amount of blogs written in English, Turkish, Indonesian, and other languages.

Ad Corpus. Our main motivation for choosing the particular blog collection just described is that its hosting platform is also an online advertising service, which provided us with the contextual advertisements it delivers. This advertisement collection includes 18,500 ads which are currently displayed with the blogs in our collection. In total, 1,650 different web sites are advertised in the collection, and 10,400 different "triggering words" are used. Figure 3.4 shows examples of advertisements from this collection.

Title:	*ArtOlive - More than 2,250 Dutch Artists*
Description:	The platform for promoting, lending and selling contemporary art. Click to view the current collection of more than 2,250 artists, or read about buying and renting art.
URL:	`www.galerie.nl`
Trigger Words:	painting, sculpture, galleries, artist, artwork, studio, artists, studios, gallery

Title:	*Start dating on Lexa.nl*
Description:	It's time for a new start. About 30,000 profiles every month. Register now for free.
URL:	`www.lexa.nl`
Trigger Words:	dating, meeting, dreamgirl, contacts

Figure 3.4: Sample advertisements from our collection.

Parameters and Model Implementation. The specifics of our implementation include the following parameters.

- For the retrieval components of our method—the construction of the "similar post" model and the final matching of the ads themselves—we used the language modeling approach to information retrieval described in [115]. This method is reported to achieve same-or-better scores as top-performing retrieval algorithms. For the similar post model, the 50 top-ranking posts from the collection were used.
- As noted earlier, our data—both blogs and ads—is in Dutch; since Dutch is a compound-rich language, we used a compound-splitting technique that has led to substantial improvements in retrieval effectiveness compared to unmodified text [118].
- In the absence of training material, a naive prior weighting scheme was used to estimate the relative importance of the different models. According to this scheme, all models have the same weight w, except the post model which is weighted $2w$ and the time model which is assigned the weight $\frac{w}{2}$.
- PMI-IR values were based on web hit counts as provided by Yahoo through its search API [320].

3.5.4 Experiments

To determine the effectiveness of our ad placement method, we compare it to two other methods for selecting ads based on content.

As a baseline, we indexed all ads and used the blog post as a query, ranking the ads by their retrieval score; in addition, the appearance of a triggering word in the post was required. This is similar to the AAK ("match Ads And Keywords") method described in [252] and used there as a baseline, except the usage of a language modeling approach to retrieval rather than a vector space one. This most likely improves the scores of the baseline: as we mentioned earlier, the language modeling retrieval method we use has been very successful in many tasks, and certainly outperforms the simpler vector space model. In the reports of the results of our experiments, we refer to this baseline as AAK.

To address the first of our main research questions (How effective are state-of-the-art ad placement methods on blogs?), we implemented the impedance coupling method AAK_EXP described in [252] (the acronym stands for "match Ad And Keywords to the EXPanded page"); this represents current state-of-the-art of content-based ad matching.[6] Again, we used the language modeling framework for the retrieval component in this method, which most likely improves its performance.

Finally, to address the second research question (can ad placement for personal blogs be improved?), we used the language modeling mixture method for ad placement described earlier in this section. We refer to this method as LANG_MODEL.

Assessment. To evaluate the different approaches, we randomly selected a set of 103 blog posts as a test set. The top three advertisements selected by all three methods for each of these posts were assessed for relevance by two independent assessors (the number of posts as well as the number of ads judged per post is identical to [252] for comparison reasons). The assessors viewed the blog posts in their original HTML form (i.e., complete with images, links, stylesheets and other components); at the bottom of the page a number of advertisements were displayed in random order, where the method used to select an ad was not shown to the assessor. The assessors were asked to mark an advertisement as "relevant" if it is likely to be of interest to readers of this blog post, be they incidental visitors to the page or regular readers.

The level of agreement between the assessors was $\kappa = 0.54$. Due to this relatively low value, we decided to mark an advertisement "relevant" for a blog post only if both assessors marked it as relevant.[7]

[6] The authors of [252] implement a number of methods for ad matching; AAK_EXP and AAK_EXP_H are the top-performing methods, where AAK_EXP_H shows a minor advantage over AAK_EXP but requires an additional crawling step which we did not implement.

[7] The requirement that two independent assessors agree on the relevance of an ad leads to more robust evaluation, but also reduces the scores, as fewer advertisements are marked as relevant. It is more strict than the assessment methodology used in [252], where each ad was judged by a single assessor only. A different policy, marking an advertisement as relevant if *any* of the assessors decided it is relevant, boosts all scores by about 40% (preserving their relative differences), but makes them less reliable.

3.5.5 Results

The metrics used to evaluate the ad selection methods were precision@1 and precision@3: the fraction of ads, out of the top-1 or top-3 ranked ones by a given method, which were marked as relevant by both assessors. As in [252], we limit the measures to the top three ads as placing many ads on a blog page is likely to disturb visitors to the blog, and evaluation should focus on early precision.

Table 3.6 shows the average precision scores for all methods, as well as a partial breakdown of the contributions made by the component models in our method (LANG_MODEL).

Method	Precision@1	Precision@3
AAK [252] (baseline)	0.18	0.18
AAK_EXP [252]	0.25 (+39%)	0.24 (+33%)
LANG_MODEL:		
Post only	0.19 (+5%)*	0.18 (0%)*
Post + Similar Posts	0.24 (+33%)	0.24 (+33%)
All models	*0.28 (+55%)*	*0.29 (+61%)*

Table 3.6: Ad-matching evaluation. Relative differences are with respect to the baseline; an asterisk marks lack of statistical significance.

All differences are strongly statistically significant using the sign test, with p values well below 0.001, except for the scores for the *Post only* model, where no statistical significance is established.

Confirming the findings in [252], the use of the sophisticated query expansion mechanism of AAK_EXP yields a substantial improvement over the baseline. However, the improvement is somewhat lower than that gained for generic web pages: while the average improvement reported in [252] is 44%, in the case of personal blogs the average improvement is 36%. Returning to the first question we raised, regarding performance of state-of-the-art ad placement algorithms on personal blogs, it appears that their success in this domain is not quite as high as for other web pages.

Jumping ahead to the combination of all models in the LANG_MODEL approach shown in the last line in Table 3.6, we observe a substantial improvement over the state-of-the-art AAK_EXP—suggesting that this is indeed a beneficial approach for capturing a profile of the blog post for commercial purposes.

An examination of the contribution of different component models of the LANG_MODEL mixture reveals a large variance: some models are highly beneficial for some of the posts, while completely redundant for others. Table 3.6 shows the performance of the mixture model for two specific combinations of models, which exhibited stable performance across different posts. The first "combination" uses only the model of the original post; this yields similar performance to the baseline, as the information used by both is identical. The second stable combination includes the post model and the *Similar Posts* model; this shows similar results to AAK_EXP, which is, again, expected—as both make use of the same information: the post itself, expanded by posts with similar content.

3.5.6 Deployment

We discussed the effectiveness of the language model mixture method; we now turn to briefly analyze its efficiency, and report on its performance in a real-life setting.

The processing steps required for each placement of ads on a blog page include collecting the text required for the different models, generating the models themselves as well as their mixture, and using the model combination to select the advertisements to be placed.

Collecting Model Text. In all models but the *Similar Post* model, aggregating the text needed for the model involves simple lookups (since a blog post points directly to its blog, its comments, and so on), and is low on computation costs. The communities, as used in our approach, are inferred offline—making generation of the *Community* model a matter of lookups too. Collecting the text for the *Similar Post* model requires retrieving from a corpus of blog posts, a step comparable to that required by AAK_EXP; given current retrieval technology, the computation required is moderate.

As the background language model is static, it is calculated offline and does not require computation per post.

Model Generation. Given the text of the model, comparing it with a background text to generate a weighted list of terms is a relatively inexpensive task; computational costs involved are substantially lower than the I/O involved in collecting the text itself.

Model Combination and Ad Selection. A time-consuming phase of combining the model is calculating their coherency values, since it involves obtaining many pairwise PMI-IR values. However, the top terms in the models follow a power law, and caching the PMI-IR values of a relatively small amount of pairs covers the vast majority of the required values for a model.

The ad selection process itself includes retrieval from a relatively small corpus of short documents—again, a computationally inexpensive task.

To summarize, the performance of our method is similar to AAK_EXP: the most demanding phase is the retrieval of additional posts for constructing the *Similar Post* model, and this is done once per blog post, not once per page view.

Model Updates. As we mentioned, once the ad selection process is complete for a given post, the selected advertisements can be stored—obviating the need for additional computations when the post is viewed again. However, there are a number of reasons to limit this type of caching. First, from a marketing perspective, displaying different ads in consecutive page loads increases exposure and enables the ad-matching platform to take into account additional parameters of the selection (such as the revenue generated by each ad). Second, showing a range of ads is useful for determining the effectiveness of the placement algorithm, by comparing its choices to the user clicks. Finally, the nature of blogs adds another reason for limiting the ad caching, a reason which is less prominent in other domains. Blogs are highly dynamic, quickly evolving over time; some models, such as the *Comment* model or the *Time* model, change frequently and require updates. A practical solution to this is to update these models sporadically, once a threshold is exceeded (for example, update the *Time* model hourly, or the *Comment* model every 5–10 new comments).

Performance. We tested the computation time of our approach using a single machine with a P4 3GHz CPU and 2GB of RAM, which repeatedly placed ads on 10,000 random blog posts. All components of the end-to-end system, including ad selection, databases, and web page serving, were deployed on the same machine. The average computation time to place advertisements on a single post was 89 milliseconds (wall-clock), well under the time required for the rest of the tasks in the pipeline—fetching the post text from the database, formatting and serving it. Moreover, the system tested did not implement any model caching or precalculation, other than caching hitcount values for reducing the time required for PMI calculations; assuming more caching is used, end performance can be improved.

To conclude, the proposed approach to contextual ad placement is not only effective but also practical; the computational load required is comparable with other ad placement mechanisms, and can be further reduced.

3.5.7 Conclusions

Our aim in this section was two-fold: to determine the effectiveness of state-of-the-art ad placement methods on blogs (as opposed to general non-blog web pages), and to propose a blog-specific ad placement algorithm that builds on the intuition that a blog represents a person, not a single topic. We used manual assessments of a relatively large test set to compare our blog-specific method to a top performing state-of-the-art one, AAK_EXP. While AAK_EXP performs well, the richness of information in blogs enables us to significantly improve over it, with little penalty in terms of complexity. We also demonstrate that deployment of our method in real-life settings is feasible.

The success of our method is based on the use of properties which are relatively unique to blogs—the presence of a community, comments, the fact that the post itself is part of a blog, and so on. We believe that further improvements may be achieved by using non-blog specific features; among these are linguistic cues such as sentiment analysis (shown, in Section 6.1, to improve other commercial-oriented tasks dealing with blogs), as well as non-linguistic ones such as ad expansion, e.g., from the page pointed to by the ad [252]. Another interesting line of further work concerns the integration of additional knowledge about the blog reader, as mined from her clickstream or her own blog or profile.

To return to the beginning of this Chapter, we set out to discover whether we could learn about the person behind the blog—her preferences and surroundings—from her writings. We proposed to do this by utilizing language model-based profiling, and demonstrated the usefulness of this approach for two tasks: matching products to bloggers, and matching advertisements to blog readers. But this type of profiling can be applied to other domains too: for example, we experimented with applying information profiles to question answering tasks, finding this a beneficial approach (our findings are out of the scope of this work, and are available in [206]). This suggests that blog profiles too can be applied to additional tasks, such as information retrieval and extraction.

Chapter 4

Text Classification in Blogs

In the previous Chapter we focused on modeling the interests and preferences of the person behind the blog, and demonstrated the usefulness of this for commercial tasks. In this Chapter we turn to a more general task involving blog content: classification and categorization, both at the blog post and at the blog level.

Text classification has many possible applications to blogs; for most of these, traditional approaches which are used with other sources of text perform well. For example, a Bayesian classifier for categorizing text into high-level domains such as “sport” or “news” and trained on a news corpus produced similar success rates on blog and newspaper test sets [179]. We therefore focus on those tasks which are more blog-specific, addressing features which are prominent in the blogspace, such as affective text or tags.

4.1 Mood Classification in Blog Posts

We have shown, in Chapter 2, that much of the uniqueness of blog language is its personal, subjective nature—including expressions of thoughts, emotions, and moods. In the work that follows, we address the task of classifying blog posts by mood. That is, given a blog post, we want to predict the most likely state of mind with which the post was written: whether the author was depressed, cheerful, bored, and so on. We approach this as a supervised machine learning task, and focus on the features we identify and their contribution to the success.

Mood classification is useful for various applications; in the particular case of blog posts (and other large amounts of subjective data) it can also enable new textual access approaches, e.g., filtering search results by mood, identifying communities, clustering, and so on. It is a particularly interesting task because it offers a number of scientific challenges: first, the large variety in blog authors creates a myriad of different styles and definitions of moods; locating features that are consistent across authors is a complex task. Additionally, the short length of typical blog entries poses a challenge to classification methods relying on statistics from a large body of text (which are often used for text classification). Finally, the large amounts of data require a scalable, robust method, with features that are inexpensive to compute.

After surveying related work, we proceed by describing the corpus used for the mood classification experiments reported in this section, and follow with an account of the classification features and the results of the classification experiments themselves.

4.1.1 Related Work

Most work on text classification is focused on identifying the topic of the text, rather than detecting stylistic features [268]. However, stylometric research—in particular, research regarding emotion and mood analysis in text—is becoming more common recently, in part due to the availability of new sources of subjective information on the web such as blogs. Read [249] describes a system for identifying affect in short fiction stories, using the statistical association level between words in the text and a set of keywords; experimental results show limited success rates, but indicate that the method is better than a naive baseline. We incorporate similar statistical association measures in our experiments as one of the feature sets used (see Section 4.1.3). Rubin et al. [257] investigated discriminating terms for emotion detection in short texts; the corpus used in this case is a small-scale collection of online reviews. Holzman and Pottenger report high accuracy in classifying emotions in online chat conversations by using the phonemes extracted from a voice-reconstruction of the conversations [119]; however, the corpus they use is small and may by biased. Liu et al. [176] present an effective system for affect classification based on large-scale "common-sense" knowledge bases.

Two important points differentiating our work from other work on affect analysis are the domain type and its volume. At the time the work presented here was made public (mid-2005), no prior work on computational analysis of affect in blogs had been published. With the increasing interest in analysis of blogs, other studies of emotions and moods in blogs have followed the work we discuss here. Among these are other mood classification experiments similar to the ones reported here [167], attempts to verify formal emotion models using blog language [84], and a characterization of a particular mood, "happiness," according to a corpus of blogs [193], which uses the corpus made available as part of the work presented here. In terms of volume, the accessibility of the blogspace offers amounts of contemporary, affect-rich, publicly-available data that far exceed the sizes of previous corpora used for these studies, such as the customer reviews or face-to-face conversations mentioned earlier. Blogs are an important domain for affect recognition—not only because of their rising importance and accessibility in recent years, but also because of their characteristics as a text corpus. The highly personal, subjective writing style and the use of non-content features such as emoticons (see Section 4.1.3) introduce new challenges.

Closely related areas to mood classification are the fields of authorship attribution [188, 152] and gender classification [151], both of which are well-studied. Since these tasks are focused on identifying attributes that do not change over time and across different contexts, useful features typically employed are non-content features (such as the usage of stopwords or pronouns). In contrast, moods are dynamic and can change—for the same author—during a relatively short time span. This causes both the features used for mood classification to be more content-based features, and the documents used for classification to be different: while authorship attribution and gender detection work well on long documents such as journal articles and even books, mood classification should be focused on short, time-limited documents.

Finally, a large body of work exists in the field of sentiment analysis, which was already mentioned in Chapter 2. This field addresses the problem of identifying the semantic polarity (positive vs. negative orientation) of words and longer texts, and has been addressed both using corpus statistics [310, 107], linguistic tools such as WordNet [137], and "common-sense" knowledge bases [176]. Typically, methods for sentiment analysis produce lists of words with polarity values assigned to each of them. These values can later be aggregated for determining the orientation of longer texts, and have been successfully employed for applications such as product review analysis and opinion mining [65, 298, 238, 220, 64, 94].

4.1.2 A Blog Corpus

The corpus used for the experiments reported here consists of 815,494 LiveJournal blog posts. LiveJournal, a popular blogging platform already mentioned earlier, was at the time of collecting the data one of the largest blogging sites, with several million users, and more than a million active ones.[1] A pioneer of explicit emotion tagging in blog posts, the interface used by LiveJournal allows users to include an optional field indicating their "current mood" at the time of posting the entry. Bloggers can either select a mood from a predefined list of 132 common moods such as "amused," "angry" and so on, or enter free-text. If a mood is chosen when publishing a blog entry, the phrase "current mood: X" appears at the bottom of the entry, where X is the mood chosen by the user. A collection of a large number of posts with an assigned mood creates a corpus of "mood-annotated" text.

One obvious drawback of the mood tagging in this corpus is that it is not provided in a consistent manner; the blog writers differ greatly from each other, and their definitions of moods differ accordingly. What may seem to one person as a frustrated state of mind might appear to another as a different emotional state—anger, depression, or maybe indifference. This is, in a way, also an advantage: unlike other corpora, in this case we have direct access to the writer's opinion

[1] Although its market share is decreasing in favor of platforms such as MySpace, in late 2006 LiveJournal still maintains close to 2 million active bloggers, according to its self-published statistics [178].

about her state of mind at the time of writing (rather than an observation of an external annotator). Another caveat is that the corpus does not constitute a representative sample of the adult population, or even of all bloggers: most LiveJournal users are under the age of 20, there are more females than males, and so on; see detailed statistics in [178]. Finally, the available "moods" for LiveJournal bloggers do not correspond to any well-known model of moods or emotions such as Plutchik's wheel model or Shaver's taxonomy [263]; while most of these moods would generally be accepted as genuine moods (e.g., "depressed," "excited"), others are arguably not real moods ("hungry," "cold"). These are all significant shortcomings for a corpus, but given the difficulty of obtaining realistic large-scale texts with mood indications, the corpus still constitutes a unique data source.

The blog corpus was obtained as follows. First, for each one of the 132 common moods given by Livejournal as predefined moods, we used the Yahoo API [320] to get a list of 1,000 web pages containing a LiveJournal blog post with that mood. Since the Livejournal web pages contain multiple blog posts (up to 20), with different moods, some of the web pages overlapped; in total, our list contained 122,624 distinct pages, from 37,009 different blogs. We proceeded to download the posts in these pages, getting in total the 815,494 posts mentioned earlier—22 posts per blog, on average. Of these posts, 624,905 (77%) included an indication of the mood; we disregarded all other posts.

As expected, the distribution of different moods within the posts follows a power law. The number of unique moods in the corpus is 54,487, but 46,558 of them appear only once, and an additional 4,279 appear only twice; such moods are inserted by users in the free-text field rather than chosen from the predefined list. Table 4.1.2 shows the distribution of the most popular moods in our corpus (percentages are calculated from the total number of posts with moods, rather than from the total number of posts altogether).

Mood	**Frequency**		**Mood**	**Frequency**	
amused	24,683	(3.95%)	calm	10,025	(1.60%)
tired	20,210	(3.23%)	bouncy	10,000	(1.60%)
happy	16,407	(2.63%)	chipper	9,509	(1.52%)
cheerful	12,939	(2.07%)	annoyed	8,236	(1.32%)
bored	12,717	(2.04%)	confused	8,129	(1.30%)
accomplished	12,137	(1.94%)	busy	7,933	(1.27%)
sleepy	11,505	(1.84%)	sick	7,817	(1.25%)
content	11,149	(1.78%)	anxious	7,036	(1.13%)
excited	11,045	(1.77%)	exhausted	6,914	(1.11%)
contemplative	10,705	(1.71%)	crazy	6,411	(1.03%)
blah	10,676	(1.71%)	depressed	6,361	(1.02%)
awake	10,072	(1.61%)	curious	6,305	(1.01%)

Table 4.1: Frequently occurring moods in our corpus.

To ensure a minimal amount of training data for each mood we attempt to classify, we only use posts for which the mood is one of the top 40 occurring moods in the entire corpus. This leaves us with 345,014 posts, the total size of which is 366MB (after cleanup and markup removal). The number of words in the corpus is slightly more than 69 million (an average of 200 words per post), while the unique number of words is 596,638.

4.1.3 Feature Set

The choice of a feature set for a classification experiment is a key part of the experiment: state-of-the-art learning algorithms will produce poor results when trained using irrelevant features. In the case of text classification, several feature sets such as word counts are commonly used; in the blog domain, additional sets of features seem beneficial. In this section we list the features we used in our experiments, grouped by "feature family."

First, we employ "classic" text analysis features—features which are used in other text classification tasks, both style-related and topic-related.

Frequency counts. Perhaps the most common set of features used for text classification tasks is information regarding the occurrence of words, or word n-grams, in the text. The majority of text classification systems treat documents as simple "bags-of-words" and use the word counts as features [268]. Other measures commonly used as features in text classifiers are frequencies of part-of-speech (POS) tags in the text. In our experiments, we use counts of both word and POS tag n-grams; POS tags were acquired with TreeTagger, a state-of-the-art tagger [267]. Comparing the n-gram models of a given mood to the n-gram model of the entire corpus in the way detailed in the previous Chapter, the usefulness of these features is apparent; Table 4.2 shows some examples of top-indicative word n-grams for various moods.

We have experimented both with unigram word and POS features, as well as with higher-order n-grams; as the improvements obtained using higher-order models over unigram models were not significant, the experiments we describe here are based on unigram models only.

Length-related features. Four features are used to represent the length of a blog post: the total length in bytes, the number of words in the post, the average length of a sentence in bytes, and the average number of words in a sentence. A naive method was used for sentence splitting, taking standard punctuation marks as sentence delimiters.

Next, we list features that are related to the subjective, opinionated nature of blog language.

Mood	**Top words**	**Top bigrams**	**Top trigrams**
hungry	hungry eat bread sauce	am hungry hungry and some food to eat	I am hungry is finally happened I am starving ask my mother
frustrated	n't frustrated frustrating do	am done can not problem is to fix	I am done am tired of stab stab stab I do not
loved	love me valentine her	I love love you love is valentines day	I love you my god oh i love him love you so

Table 4.2: Most discriminating word n-grams for some moods.

Semantic orientation features. The semantic orientation of a word is "its evaluative character" [298]—whether it expresses an overall negative or positive concept. Semantic orientation is measured both in terms of polarity (positive to negative) and intensity (weak to strong); examples of words with a positive orientation are "modest" (weak) and "tremendous" (strong); similarly, negatively-oriented words include "doubtful" (weak) and "tyranny" (strong).[2] In text analytics, semantic orientation at the word, sentence, or document level is often used to estimate the overall orientation of a piece of text, such as product reviews (e.g., [65, 298, 238]).

The semantic orientation of a blog post seems like a particularly useful feature for mood prediction: some moods are clearly "negative" (annoyed, frustrated) and some are clearly "positive" (cheerful, loved). The contents of blog posts with positive moods will most likely have a more positive orientation than negative-mood ones.

In our experiments, we use both the total orientation of all words in a post, and the average word orientation in the post as features. Since the estimation of word semantic orientation is highly dependent on the method used for calculating it, we use two different sources for the word-level orientation estimation.

The first source is a list of 21,885 verbs and nouns, each assigned with either a positive, negative, or neutral orientation. The method used for creating this list is described by Kim and Hovy in [143]; it is based on the WordNet distances of a word from a small set of keywords with well-known orientation. To calculate the total and average orientation of a post, we assign a value of +1 to every positive word and −1 to every negative one, summing (or averaging) the words.

The second source we use is a similar list of 1,718 adjectives with their corresponding real-numbered polarity values, either positive or negative. This list

[2] While many words with positive or negative orientation are adjectives, nouns sometimes have a non-neutral orientation too, as in the example of "tyranny."

was constructed using Turney and Littman's method described in [298]; their method is based on measuring the co-occurrence of a word with a small set of manually-classified keywords on the web.

The two main differences between the lists are coverage (the former list being substantially larger), and the inclusion of information regarding the intensity of the semantic orientation in the latter list: higher absolute values mean a stronger orientation. Additionally, the difference in the methods used to create the lists results in occasional disagreement between their value; examples of words and their values in both lists are given in Table 4.3, illustrating this. In fact, out of 1,327 words found on both lists, 378 (28%) had conflicting values.

Word	**Kim&Hovy**	**Turney&Littman**
pricey	Positive	−4.99
repetitive	Positive	−1.63
teenage	Negative	−1.45
momentary	Negative	+0.01
fair	Positive	+0.02
earnest	Positive	+1.86
unparalleled	Negative	+3.67
fortunate	Positive	+5.72

Table 4.3: The semantic orientation values of words according to different sources.

Mood PMI-IR. The next set of features we use is based on PMI-IR, introduced earlier in Section 3.4. Recall that PMI-IR is used to calculate dependency between terms based on their co-occurrence in a large collection of text:

$$\text{PMI-IR}(t_1, t_2) = \log \frac{\text{df}(t_1 \& t_2)}{\text{df}(t_1) \cdot \text{df}(t_2)}$$

Individual PMI values between words in a document and a given concept can be aggregated to estimate an overall measure of relation between the concept and a document, e.g., by summing them [298]. In our setting, classification of text by mood, the "concepts" are moods: we are interested in measuring the association between the words used in a blog post and a set of known moods. For this, we calculated the PMI-IR of each of the 3,000 most frequently occurring words in the corpus with each of the top 40 occurring moods (a total of over 100,000 individual PMI-IR values). The values were then normalized, for each word, between −1 (assigned to the least-associated-mood) and +1 (highest-associated-mood); low values indicate a good negative correlation between the words and a mood, high values a good positive correlation, and words which are not correlated with the mood at all have PMI-IR values around zero. Search engine hit-counts were obtained from Yahoo using their API [320]; some example PMI-IR values

Mood	Word	PMI-IR
happy	apart	−0.98
	somebody	−0.03
	appreciated	+0.66
	joke	+0.82
bored	funny	−0.78
	early	−0.02
	obvious	+0.33
	homework	+1.00

Word	Mood	PMI-IR
food	blank	−0.56
	depressed	+0.63
	cranky	+0.85
	hungry	+0.96
tomorrow	great	−0.32
	thoughtful	−0.08
	busy	+0.23
	sleepy	+0.52

Table 4.4: Example normalized PMI-IR values of ⟨word,mood⟩ pairs, grouped by moods (left) and words (right).

between a frequently occurring word and a mood (after normalization) are shown in Table 4.4.

Once the individual PMI-IR values between the frequent words and the frequent moods are calculated, incorporating them as features in the learning process is straightforward. We used 80 PMI-related features for each blog post, in a similar manner to that described in [249]. These 80 features consist of the total PMI-IR of all words in the post, and the average PMI-IR of the words—with each of the top-40 moods for which the PMI-IR values were calculated. Words included in the post but not found on the list of 3,000 most frequent words were ignored.

Finally, we turn to features that are unique to online text such as that appearing in blogs, email, and certain types of web pages.

Emphasized words. Historically, written online text such as email was unformatted (that is, raw ASCII was used, without layout modifiers such as different font sizes, italic text and so on). This led to alternative methods of text emphasis, including using all-capitalized words ("I think that's a GREAT idea"), and usage of asterisks or underscores to mark bold or italic text ("This is *not* what I had in mind," "Did you bother _checking_ it before sending??").

While today most online text has extensive formatting options, these emphasis methods are still used sometimes, especially in cases where text is added through a standard text-box on a web page, with no formatting options. Our corpus contains 131,829 such emphasized words, in 71,766 different posts (9% of all posts).

Since emphasized words are explicitly indicated as such by the blogger, they are, at least intuitively, important indicators for the content type of the post. On top of this, many of the popular emphasized words are used to convey emotion: the five most popular such words in our corpus are "*sigh*," "*shrugs*," "*grins*," "*cough*," and "*laughs*"—words which are clearly related to the author's feelings at the time of writing, or to her attitude towards the topic discussed. We use as a feature the frequency of each emphasized word in a post, as well as the total number of stressed words per post.

Special symbols. This last set of features captures usage of two types of special characters in the blog posts. The first type is punctuation characters such as ellipsis, exclamation marks, and so forth. The intuition behind modeling the frequencies of these symbols is that, in some cases, increased usage of them is characteristic of certain kinds of text [264], and was found beneficial in some text classification tasks, such as detecting email spam [260]. Punctuation appears as a relevant feature for mood detection too; for example, usage of exclamation marks is more likely to coincide with moods such as "excited" or "frustrated" than with "sleepy" or "curious." We use as features the frequencies of common special symbols in each blog post; these include punctuation marks and some additional non-alphanumeric symbols such as asterisks and currency signs (in total, 15 such symbols were used).

The second type of special symbols we use as feature are *emoticons* (emotional icons). Emoticons are sequences of printable characters which are intended to represent human emotions or attitudes; often, these are sideways textual representations of facial expressions. Examples of such emoticons are `:)` (representing a smile) and `;)` (representing a wink)—both viewed sideways. Usage of emoticons originated in email messages and quickly spread to other forms of online content. It is often used in blogs: our corpus contains 92,991 emoticons, appearing in 64,947 posts, 8% of all posts. Similarly to the first set of special symbols, we use the frequencies of popular emoticons in the blog posts as features: we used 9 such symbols, each appearing in 100 posts or more.

4.1.4 Experimental Setting

In this section we describe the experiments performed for classifying the mood of a blog post and their results. Our main question is, naturally, to what degree mood can be classified from blog text, when compared to similar text classification tasks such as demographic profiling or authorship attribution. Additional questions we intend to address are whether increasing the amount of training data results in significant improvements to the classification performance; whether classification accuracy improves for posts with a larger amount of text; and whether classifying abstract mood categories (such as "active" and "passive") is easier than classifying the exact mood.

Setup. The family of learning algorithms we use for the experiments is support vector machines (SVMs).[3] SVMs are popular in text classification tasks since they scale to the large amount of features often incurred in this domain [133];

[3] We used the SVMlight toolkit, `http://svmlight.joachims.org/`.

they have been shown to significantly outperform other classifiers for this type of task [321].

Experiments. We performed two sets of experiments. The first set is intended to evaluate the effectiveness of identifying specific, individual moods in a blog post, and to examine the effect of changes in the training set size on classification accuracy. For each mood we created a training set with randomly drawn instances from the set of posts associated with that mood as positive examples, and an equal amount of negative examples, randomly drawn form all other moods. The test set we used contained, similarly, an equal amount of random positive and negative instances, distinct from those used for training.

For the second set of experiments, we manually partitioned the moods into two "mood sets" according to some abstraction, such as "positive moods" vs. "negative moods." We then repeated the training and testing phase as done for the individual mood classification, treating all moods in the same set as equivalent. The purpose of these experiments was to test whether combining related moods improves performance, since many of the moods in our corpus are near-synonyms (e.g., "tired" and "sleepy").

For classifying individual moods, the training set size was limited to a maximum of a few thousand positive and a few thousand negative examples, since many moods did not have large amounts of associated blog posts (see Table 4.1.2). For classifying the mood sets, we used a larger amount of training material.

Since both our training and test sets contain the same number of positive and negative examples, the baseline to all our experiments is 50% accuracy (achieved by classifying all examples as positive or all examples as negative).

4.1.5 Results

Table 4.5 lists the results of the classification of individual moods. The test sets contained 400 instances each. For the training sets we used varying amounts, up to 6,400 instances; the table lists the results when training with 1,600 instances and with 6,400 instances.

The classification performance on most moods is modest, with an average of 8% improvement over the 50% baseline (with 6,400 training examples); a few moods exhibit substantially higher improvements, up to 15% improvement over the baseline, while a small number of moods perform equivalently or worse than the baseline. Examining the better and worse performing moods, it seems that the better ones are slightly more concrete and focused than the worse ones, e.g., "depressed," "happy" and "sick" compared to "okay" and "calm." However, this is not consistent as some concrete moods show low accuracy ("hungry") whereas some of the non-focused moods perform averagely ("blah").

An error analysis reveals that many of the misclassified posts are fairly short,

	Correct			**Correct**	
Mood	1,600	6,400	**Mood**	1,600	6,400
confused	56.00%	65.75%	tired	52.00%	55.25%
curious	60.25%	63.25%	bored	51.50%	55.25%
depressed	58.25%	62.50%	sleepy	44.25%	55.00%
happy	54.50%	60.75%	crazy	54.00%	55.00%
amused	57.75%	60.75%	blank	56.00%	54.50%
sick	54.75%	60.25%	cheerful	52.50%	54.25%
sad	53.00%	60.25%	anxious	51.75%	54.25%
frustrated	57.00%	60.25%	aggravated	52.75%	54.25%
excited	55.50%	59.75%	content	50.75%	54.00%
ecstatic	54.00%	59.75%	awake	51.50%	53.75%
bouncy	51.00%	59.50%	busy	50.75%	53.50%
thoughtful	52.75%	59.00%	cold	50.25%	53.25%
annoyed	57.00%	59.00%	exhausted	52.50%	52.50%
loved	57.00%	57.75%	drained	47.50%	52.25%
blah	53.75%	57.75%	hungry	51.50%	50.75%
hopeful	51.50%	57.50%	good	48.50%	50.50%
cranky	55.00%	57.25%	creative	47.75%	50.50%
contemplative	53.25%	57.00%	okay	46.75%	49.00%
accomplished	54.75%	55.75%	calm	44.75%	49.00%

Table 4.5: Classification performance: individual moods.

meaning that most of the features used in the classification process mean little. When focusing on longer posts, performance increases visibly. For example, Table 4.6 shows results of similar classification experiments, performed only on the longer posts: posts with less than 250 words were excluded from the training and test sets altogether. The resulting training sets were smaller, ranging up to 1,000 instances (the table shows results for 500 and 1,000 instances). In this setting, the average improvement over the baseline is 16%, and the success rates on top-performing moods approach that of similar complex classification tasks such as authorship attribution.

The most discriminating features according to the formed SVM models included the semantic orientation of the post; its PMI-IR values with various moods; frequencies of certain POS tags—nouns, prepositions, adjectives, and determiners; and length features. Words whose frequencies were beneficial as features included first- and second-person pronouns ("I," "my," "you") as well as words with obvious relations to some of the moods ("night," "miss," "happy," "funny").

Features which had little or no contribution to the learning process included mostly word frequencies (both stressed and unstressed)—unsurprising given their sparseness. Most special characters—excluding ellipsis and some forms of emoticons—were also ranked low in the list of contributing features.

We now turn to the second set of experiments, that in which sets of moods—rather than single moods—were classified. The results of the classification of two

	Correct			**Correct**	
Mood	500	1,000	**Mood**	500	1,000
accomplished	73.50%	77.50%	aggravated	54.50%	67.00%
contemplative	70.75%	75.00%	hopeful	51.00%	66.25%
sleepy	62.25%	72.75%	drained	58.25%	66.25%
busy	64.00%	72.25%	creative	65.00%	66.00%
content	61.00%	72.00%	bouncy	55.75%	66.00%
bored	69.00%	72.00%	anxious	57.75%	65.50%
calm	65.50%	71.75%	excited	55.00%	65.25%
tired	62.50%	71.50%	good	55.25%	65.25%
cheerful	60.00%	71.50%	sick	50.00%	64.75%
blah	62.75%	71.25%	cold	55.25%	64.50%
exhausted	58.25%	71.25%	blank	52.25%	64.50%
happy	58.75%	70.50%	confused	52.75%	64.25%
thoughtful	65.75%	70.00%	hungry	56.25%	63.50%
amused	65.75%	69.75%	depressed	52.75%	62.75%
curious	59.75%	69.50%	frustrated	50.50%	62.75%
awake	60.00%	69.00%	crazy	49.25%	62.50%
loved	65.00%	67.75%	sad	48.25%	62.25%
okay	56.00%	67.50%	cranky	47.75%	60.50%
annoyed	54.75%	67.25%	ecstatic	44.75%	59.50%

Table 4.6: Classification performance: individual moods, discarding short posts.

mood partitions—active/passive and positive/negative—are shown in Table 4.7, for increasingly large training sets. Somewhat surprising, the classification of the aggregated sets does not seem to be an easier task than classifying a single mood, despite the substantial increase in the amount of training examples, and even when discarding the short posts (Table 4.8). A possible explanation is that this is an over-abstraction: "angry" and "tired" are both negative, but otherwise share little. In work that follows the work presented here, Gènèreux and Evans show that some partitions of moods can reach higher accuracy rates [84].

Size of training set	**Active/Passive**	**Positive/Negative**
800	50.51%	48.03%
1,600	50.93%	53.00%
3,200	51.50%	51.72%
6,400	51.77%	54.92%
20,000	53.53%	54.65%
40,000	55.26%	57.33%
80,000	57.08%	59.67%

Table 4.7: Classification performance: active vs. passive moods and positive vs. negative moods.

Size of training set	**Active/Passive**	**Positive/Negative**
800	55.26%	55.88%
1,600	56.62%	56.76%
3,200	57.50%	58.24%
6,400	57.62%	59.00%
10,000	58.38%	60.88%
20,000	59.02%	61.18%

Table 4.8: Classification performance: active vs. passive moods and positive vs. negative moods, discarding short posts.

4.1.6 Discussion

Disappointed by the results, particularly for the case where post length was not constrained, we decided to let humans perform the individual mood classification task, and see if this yields substantially higher performance. For each one of the 40 most frequent moods, we randomly selected 10 posts annotated with that mood, and 10 posts annotated with a random other mood. We then presented these 20 posts to a human assessor without their accompanying moods; the assessor was told that exactly 10 out of the 20 posts are of mood X (the mood was explicitely given), and was asked to select which ones they are. This process simulates the same test data used with the machine learning experiments: a 50% partition of test examples, and the same type of data provided to the human and the machine. The accuracy of the manual classification over these 800 posts was 63%—only slightly better than the average performance of the classification over all moods, 58%. Similar human assessment of 400 longer posts (over 250 words) yielded an average accuracy of 76%, better than the average automated accuracy (66%), but indicating that machine performance in this case too is reasonable.

To return to our research questions, it seems that the mood classification task is indeed a complex one, for humans or machines, and that methods and features used for other stylistic analysis—even when augmented with a range of additional features—do not provide the same accuracy levels as in related tasks. Part of the complexity can be explained in the relatively short length of many blog posts, resulting in sparse features to train and classify. A possible additional reason for low accuracy—both of the human and the machine—is the subjective nature of the "annotation" in the corpus: it depends not only on the text, but also on how the blogger perceives a given mood (as opposed to lab-controlled experiments, where annotators follow guidelines and try to be consistent).

One clear observation is that increasing the size of the training set affects favorably the performance in the vast majority of the cases, particularly for single-mood classification, and to a lesser extent also for mood-set classification. We believe this indicates that our results can still improve by simply further increasing the training data size; given the continuing growth of the blogspace, fairly large training sets can be accessed.

Finally, we note that classification accuracy for longer posts is substantially higher than that obtained for shorter ones—an expected result, given the statistical nature of most of our features.

4.1.7 Conclusions

We used a text classification approach to classify moods of blog posts, focusing on stylistic features of the text, and using some features which are unique to blogs. Evaluation on a large collection of blog posts in which the bloggers indicate their state-of-mind at the time of posting show a small, if consistent, improvement over a naive baseline; constraining the experiments to longer posts yields improved performance, approaching that of other stylistic analysis tasks. While the success rates are modest, human performance on this task is not substantially better: this is a difficult task for humans and machines alike, and the wealth of features available for the learning process does not ensure high performance. Furthermore, our results indicate that increasing the amount of training data results in a clear improvement in effectiveness; we have not reached a saturation point in this improvement in our experiments, meaning that further improvement is expected with more training data.

4.2 Automated Tagging

The previous section presented a classification task which is related to the particulars of blog language, namely, its personal, subjective language. For "standard," topical classification, off-the-shelf approaches are reported to work well: we have already mentioned work indicating similar accuracy levels of topical classification in blogs and in a news corpus [179]. Similarly, a simple tf·idf distance measure was used to obtain an accuracy level of 84% for a four-way blog classification task, between the types personal, news, politics, and sports [247]. The main issue with this approach to classification is that since a small, pre-determined set of categories is chosen in advance, the resulting abstraction level of categorization is high. High-level categorization may be useful for blog readers in some scenarios (i.e., limiting search results to a specific category of blogs), but other scenarios require a much more detailed classification scheme (e.g., clustering results by domain or suggesting "similar posts" for a given post).

In the work presented in this section, we revisit topical text classification in blogs from a different, more blogger-oriented angle: instead of classifying blogs into predefined classes, we address the task of determining useful tags for a blog post, given its text. In a nutshell, we aim at providing the blogger and her audience with a small set of concrete topics, concepts, and keywords which can be used to tag a post. More than just simplifying the tagging process, automated tagging also improves its quality: first, by increasing the chance that blog posts will be tagged in the first place; second, by offering relevant tags that may not have been applied otherwise; and finally, by reducing the number of different tags used for the same concept, assuming bloggers choose the most popular tag out of a few variants (e.g., "half marathon," "1/2 marathon," "half-marathon race"). This in turn improves the tasks for which tagging is aimed at, providing better search and browse capabilities.

We proceed by setting the background to automated tag assignment, and reviewing related work; we then present our approach to this task, based on collaborative filtering; finally, we describe experiments that evaluate our proposal.

4.2.1 Tagging and Folksonomies in Blogs

Tagging has already been introduced in Chapter 2. It is an old-new method for organizing data by assigning descriptors to various resource such as photographs or video clips. The descriptors—"tags"—are typically short textual labels, which provide an easy way to categorize, search, and browse the information they describe. Annotation of documents with keywords is nothing new by itself, but a collaborative form of this method with some unique properties is attracting a lot of attention on the web in recent years. The main characteristics of collaborative tagging differentiating it from traditional keyword annotation are its open-vocabulary, non-hierarchical nature, and the fact that tags are assigned by authors and users of the information rather than by professional annotators, with no rules or limitations [91, 186]. Contrasting this with taxonomies—hierarchical, structured categorizations managed by professionals, the collective of all tags assigned to different resources is called a *folksonomy*—a categorization scheme created by the people, for the people.

At the time the work presented here was made public (late 2005), no empirical work was available on automated tag assignment for blog posts; Avesani et al. [16] discussed in broad terms extraction of topical tags from blog posts, but with no experimental results. Studies of collaborative tagging itself were few despite the popularity of tags in the blogspace as well as in other domains (in mid-2006, half of all blog posts were assigned tags [67], and the rate was growing). A notable exception is the work of Golder and Huberman [91] on modeling tagging systems, examining data from a popular bookmark tagging site, `del.icio.us`. Later, academic interest in folksonomies picked up, with dedicated venues (e.g., [318]); in particular, automated tagging in blog posts was proposed using keyword extraction techniques [41] and machine learning (with a restricted set of tags) [232].

4.2.2 Tag Assignment

In the rest of this section, we describe a system that, given a blog post, offers a small set of tags which seem useful for it. The blogger can then review the suggestions, selecting those which she finds instrumental.

Our basic approach to automated tag assignment is that of a recommender system, a system designed to assist users in selecting items from large repositories [251]; in many cases, items are products, although recommender systems have also been applied in other domains, such as recommending Usenet articles [289] or sharing knowledge in a community [90].

Recommender systems work by using existing knowledge about the user (i.e., the previous items she selected) to offer items which are likely to appeal to her in the future. A prominent approach to recommendation is that of collaborative filtering [251], where the knowledge of items selected by other users is used to make the recommendations. Amazon.com, TiVo, Netflix and others are among the many successful commercial applications of recommender systems using collaborative filtering approaches.

How collaborative filtering systems work. In a nutshell, a recommender system based on collaborative filtering uses not only the previous choices of a user to recommend products or services to her, but the choices and ratings of other users too. The two main approaches to collaborative filtering are *model-based* and *memory-based*; model-based systems use the collective ratings of users to form a model of recommendation through a learning process (e.g., [117]). Memory-based approaches—in a manner similar to memory-based learning—store the raw ratings of users, using them to identify recommended items for another user. The memory-based approach is more common, and is also the one we follow; we will restrict our discussion to it.

Recommendations in memory-based collaborative filtering systems are based on similarity between items (item-based) or between users (user-based). Item-based systems predict a user's interest in an item according to a community-dependent similarity between the previous items the user expressed interest in and the new item. An example for such an approach is the Amazon recommendation system, which uses groups of items commonly purchased together by users to measure item similarity [174]. User-based systems, on the other hand, utilize the similarity between users rather than products. These approaches recommend to a user items that have been chosen by similar users to her, and by that model human word-of-mouth recommendations: people often consult others who they believe have similar interests and views when choosing products such as books and movies.

More formally, the components of a memory-based, user-based collaborative filtering system are a set of users U, a set of items I, and a set of rankings of items by users, often stored in a $|U| \times |I|$ matrix. Rankings are derived from the user's actions, such as purchasing a product, viewing it, explicitly rating it, and so on; they can be binary or real-valued. When asked to provide recommendations to a user $u \in U$, the system first identifies the user's neighborhood—the set of users in

$U \setminus \{u\}$ which are most similar to it, given some similarity function between users. Then, the item ratings of the neighborhood are aggregated, and the top-ranked items form the recommendation. The user similarity function itself is often based on the user item rankings, assuming that users who expressed interest in similar items in the past are themselves similar: in practice, a cosine similarity score is usually calculated between vectors representing the rankings of the users.

In some cases, additional knowledge is combined into this framework. For example, the global popularity of items across all users (and not only those similar to a given user) is sometimes used. One theoretically sound way of doing this is within a probabilistic framework, where the collaborative filtering-based recommendations are represented as probabilities and combined with other probabilities, such as priors assigned to all items [132].

Collaborative filtering and tagging. An application of collaborative filtering methods to automated tag assignment suggests itself when the "user" and "product" concepts are examined from a different angle. In our approach, the blog posts themselves take the role of users, and the tags assigned to them function as the products that the users express interest in. This is fitted into the user-based collaborative filtering approach: in its traditional form, similar users are assumed to purchase similar products; we make the same assumption, identifying useful tags for a post by examining tags assigned to similar posts from a large collection. Recommendations are further improved by incorporating knowledge about tags previously used by the blogger using a probabilistic approach.

We follow with details on the stages of the collaborative filtering approach: the similarity measure we use between posts, the manner in which tags used by similar posts are aggregated to a single list, and the usage of information about previously used tags.

Post similarity. As mentioned earlier, the standard approach to calculating similarity between two users in user-based recommender systems is based on the similarity between the sets of items the users ranked. This cannot be applied in our model, as no previous information is given about a blog post: it is not tagged, meaning that there are no "previously ranked items" to use as a basis for similarity. Since blog posts are text documents, we can instead use their content to calculate similarity, by applying standard text-based similarity measures. With a large repository of tagged posts, an information retrieval approach to similarity calculation can be used. In practice, this means the post collection is indexed by an IR engine, and a query generated from the original post is submitted to it. The k most similar posts are then taken to be the k highest-ranking posts retrieved from the collection using this query according to some retrieval model.

A posterior tag model. Once a set of tagged similar posts is obtained, we simply count, for each tag assigned to any of them, the number of times it occurs in the set. Experiments with more complex ways of scoring the tags, taking into account the retrieval rank or score, yielded only minor improvements in accuracy, and were not further pursued. We refer to this ranked list of tags as the posterior tag model, a model dependent on a given post. Formally, given a post x and the k most similar posts to it in a large collection, $S_k(x)$, we assign a posterior likelihood to each tag t as follows:

$$p_{\text{posterior}}(t) = \frac{count(t, S_k(x))}{|S_k(x)|}$$

Combining prior knowledge. When describing the similarity function between posts which we use, we noted that unlike most collaborative filtering settings, in our case there are no "previous rankings" for a post, and its contents are used for the similarity instead. While previous knowledge does not exist for a given post, it does exist at the blog level: given a blog post, we have access to other posts from the same blog, and the tags assigned to them. The fact that a tag has been used by a blogger does not necessarily make it suitable for future posts—obviously, posts from the same blogger dealing with different topics are unlikely to share many tags. However, if such a previously-used tag appears in the tag model derived from tags assigned to other, similar posts, it is likely to be relevant to the new, untagged post too.

To account for this within our model, we introduce a prior tag model, in addition to the posterior model derived from the similar posts. This model assigns each tag a post-independent prior likelihood of being used by a given blogger: the likelihood that this tag will be used by the blogger. Given a blog post x, the prior model is constructed using a maximum likelihood estimate from the blogger's previous tag usage: the likelihood assigned to a tag is its frequency in other posts from the same blog which predate x, normalized by the total number of such posts. Formally, the prior likelihood assigned to tag t, given the set X of posts from the same blog as x is

$$p_{\text{prior}}(t) = \frac{count(t, \{x' \in P \mid \text{date}(x') < \text{date}(x)\})}{|X|}$$

Finally, the prior and posterior likelihoods are combined linearly, so that the overall likelihood of tag t is

$$p_{\text{combined}}(t) = \lambda \cdot p_{\text{posterior}}(t) + (1 - \lambda) \cdot p_{\text{prior}}(t).$$

The tag suggestion process is summarized and demonstrated in Figure 4.1. Given a blog post, two sets of other posts are retrieved: posts from the same blog predating it, and the most similar posts to in a large collection of posts. Next, a

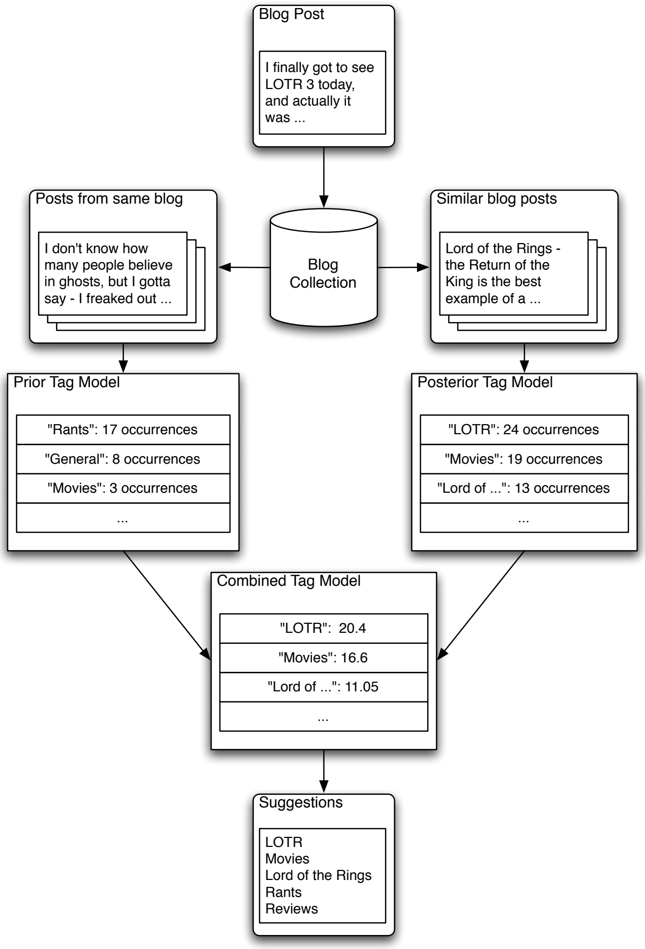


Figure 4.1: Flow of information for automated tagging.

maximum likelihood estimate is used to construct two tag models from these two sets of posts, a prior model and posterior one. The models are then combined using a linear combination, and the top-ranked tags according to the combination are offered to the user, who selects the tags to actually attach to the post.

4.2.3 Experimental Setting

We now describe the experiments with which we evaluated the proposed tag assignment method. Aside from testing the usefulness of our automated tagging concept, we explore the effect of varying the parameters of our model on performance, as well as compare it to human performance.

As a baseline, we use the tagging approach described in [41], according to which all terms appearing in a post are scored according to their tf·idf values, and the highest scoring terms are selected as tags. This approach was developed in parallel to ours.

Data. The blog post collection we used was the corpus distributed for the 3rd Annual Weblogging Workshop.[4] It consists of 10 million blog posts collected during a period of 3 weeks in July 2005; of these, 1.8 million posts are tagged, with a total of 2.3 million tags. To index the collection we used the open source retrieval system Lucene [106], which uses a rather simple vector space retrieval model; text was stemmed with an English Porter stemmer.

Test sets and evaluation metrics. To evaluate the tags suggested for a post, we compare them with actual tags assigned to it by its author. While this limits the evaluation to tagged posts, the amount of those in our collection is enough for a large test set with good coverage: we used a set of 4,000 randomly-selected posts, each tagged with five or more tags. To account for minor variations between assigned tags and "ground truth" tags (e.g., "blogs" and "blogging"), string distance is used to compare tags rather than exact string matching.

The evaluation metrics we use are precision and recall at 10, as well as R-precision [21]. The precision at 10 is the fraction of tags, out of the top-10 suggestions, which were used by the author of the post; recall at 10 is the fraction of tags used by the blogger, which also appear in the top-10 suggestions, out of the total number of tags used by her. Cutoff at 10 was selected as users are most likely interested in a handful of recommendations, and unwilling to examine long lists of possible tags. To account for the differences in the number of ground-truth tags between posts, and to provide a single metric for the performance, we use R-precision: the number of correct tags out of the top-ranked R ones, where R is the number of known tags according to the ground-truth.

Experiments and results. The three main parameters of the model we use are the following:

- The manner in which a query is generated from a post for the purpose of retrieving similar ones.
- The number of top-ranking similar posts from which tags are aggregated into the tag model (k).
- The linear combination parameter, λ, used to combine the prior and posterior models; when $\lambda = 0$ only the prior model is used, and when $\lambda = 1$ only the posterior one is used.

We experimented with different techniques for creating a query from a post, with a baseline of simply using the entire post as a query. Although minor improvements over this baseline were obtained, they were not significant. As this is not the main question we address, in the rest of the experiments described here we use the entire post text as a query. Best results were obtained with $k = 50$, $\lambda = 0.8$, and appear

[4] http://www.blogpulse.com/www2006-workshop

in Table 4.9 along with the scores of the tf·idf baseline; all improvements are statistically significant using the t-test. The variation of R-precision given for different values of λ and k are shown in Figure 4.2.

	Precision@10	**Recall@10**	**R-Precision**
Baseline (tf·idf, [41])	0.1624	0.2014	0.1393
Collaborative Filtering	0.3589 (+120%)	0.4946 (+145%)	0.4263 (+206%)

Table 4.9: Auto-tagging accuracy: our collaborative approach with $k = 50$, $\lambda = 0.8$ compared to a tf·idf baseline.

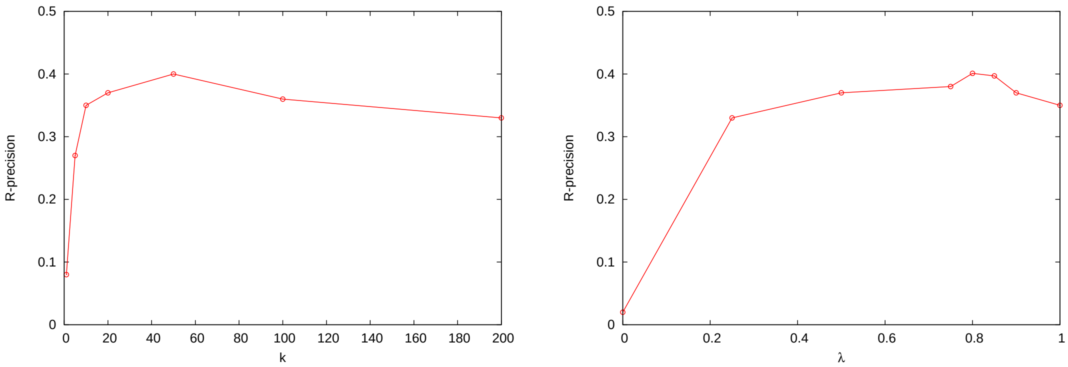


Figure 4.2: Auto-tagging accuracy for different values of k (left, λ fixed at 0.8): and λ (right, k fixed at 50).

It is important to note not only that our approach performs substantially better than a tf·idf one, but also that the resulting tags are in many cases more beneficial. Tags based on a tf·idf approach are terms originally appearing in the post; on the other hand, since the tags our method suggests are not necessarily found in the text of the post itself, they add new information to it—increasing their usefulness to retrieval systems and other automated access methods to the posts. Examples of tags suggested for a specific post are shown in Table 4.10, along with actual tags used by the blogger for that post.

On average, for a given post, more than a third of the suggestions generated by our method were actually used by the blogger. An examination of posts with low precision scores shows many non-English posts (for which much less data exists in the corpus, leading to lower success of data-driven methods), and many tags which are highly personal and used by few bloggers (such as names of family members).

The automated evaluation method we use is not an ideal one: tags which are useful for a post but were not originally used by its author are incorrectly judged as irrelevant; the resulting scores are artificially low due to this strictness. An ideal evaluation would be based on human judgments: the tags assigned to a

Post	`http://www.stillhq.com/diary/000959.html` *On pitching products to bloggers* Anil comments on how to pitch a product to him as a blogger, and makes good suggestions as Lindsay agrees before getting distracted by how this applies to press releases. I have to wonder though how much of this promotional pitching actually happens. I certainly haven't ever had a product pitched to me for this site. I've had people pitch advertising, and other spammy things, but not products. Does it really happen to us normal people bloggers?

Suggested tags	PR; blogging; weblogs; marketing;
	net; gripes; email;
	small business; life; Anil Dash

Actual tags	blog; pitch; product; marketing; communications

Table 4.10: Sample tags suggested for a post.

post are assessed as relevant or irrelevant though manual examination of the post. This form of evaluation is expensive, but we hypothesized it will prove that our tag assignment proposal performs better than the automated measure indicate. To demonstrate this, we manually evaluated a random set of 30 posts, comparing the precision at various levels as assessed by a human with that obtained using the automated evaluation (recall cannot be measured as the "complete" set of relevant tags is unknown to the human). The results of this comparison are shown in Table 4.11, clearly showing that a human judges additional tags as relevant—not only those that were actually chosen by a blogger, and therefore are available to the automated evaluation procedure. As the results of the automated evaluation on this small test set are similar to the results of it on the large, 4,000 post set, we believe that had we been able to manually assess the larger set we used for most of the experiment, higher scores would have been obtained: the number of relevant tags out of those suggested would be closer to two-thirds of the top suggestions than to one-third. Also interesting to note is that, both for automated and manual evaluation, the precision at 1 or 5 is substantially higher than the precision at 10, meaning that the top-suggested tags tend to be more relevant than lower-ranked ones (so, if a user wishes to be offered only a few tags rather than 10, the percentage of useful ones in them will be higher).

Evaluation type	**Precision@1**	**Precision@5**	**Precision@10**
Automated	0.4333	0.3983	0.3620
Manual	0.7333	0.6440	0.5992

Table 4.11: A comparison of manual and automated auto-tagging precision on a random subset of 30 posts.

4.2.4 Conclusions

We proposed and evaluated a method for tagging blog posts based on a collaborative filtering approach: given a blog post, tags for it are suggested according to tags assigned to other, similar posts. Applying this type of automated tagging mechanism can benefit both the bloggers themselves, as well as their readers and others making use of tags in blog posts. Despite a relatively simple approach, we obtain satisfactory results, with at least a third of the top 10 suggested tags being useful for the blogger—and, according to human evaluation, a substantially higher fraction. Furthermore, within these 10 tags, the top-ranked ones are more relevant, on average, than the lower-ranked ones, so a blogger examining only the first few suggestions is likely to find an even higher percentage of useful tags.

In terms of the model parameters, we witness that incorporating the prior knowledge about previous usage of tags by a blogger contributes only modestly to overall performance, and that usage of a moderate number of tagged posts for aggregating suggestions yields the best performance.

To summarize this Chapter, our text classification approaches aimed at the single blog level focused on two blog-specific aspects of the text. First, we studied emotions in blog posts, and, more specifically, how the mood reported by the blogger can be derived from the text. We found that while in some cases—for some moods and for relatively long blog posts—this is a tangible task, in the majority of the cases it proves more complex than similar, non-affect related text classification tasks, despite usage of state-of-the-art approaches. The second text classification technique for blogs we addressed is a topical classification task, but one focused on tags—open vocabulary categorization labels that are attached to some posts to facilitate their organization. Here, we found that a collaborative filtering approach which harnesses the decisions of other bloggers performs well, substantially improving a content-only based method. In both cases, we show that the properties of blogs as text documents give rise to new challenges in text classification, as well as offer ways to address these challenges.

Chapter 5

Comment Spam in Blogs

The final issue we address at the level of single blogs using text analytics concerns a specific type of spam prevalent in the blogspace: comment spam, already described in broad terms in Section 2.3.3. First, we provide more details about this form of spam.

5.1 Comment and Link Spam

The growing popularity of internet search as a primary access point to the web has increased the benefits of achieving top rankings from popular search engines, especially for commercially-oriented web pages. Combined with the success of link analysis methods such as PageRank, this led to rapid growth in link spamming—creating links which are "present for reasons other than merit" [68].

Comment spam is essentially link spam originating from comments and responses added to web pages which support dynamic user editing; blogs are a primary target of link spam through the commenting mechanism, along with other user-modifiable pages such as wikis and guest books. Blogs have made the life of comment spammers easy: instead of setting up complex webs of pages linking to the spam page or link exchange schemes, the spammer writes a simple agent that visits blog pages, posting comments that link back to her page. Not only is spamming easier, but the spammer also benefits from the relatively high prestige assigned by many search engines to blogs, stemming both from the rapid content change in them and the density of links between them. Comment spam, and link spam in general, poses a major challenge to search engines as it severely threatens the quality of their ranking. Commercial engines are seeking new solutions to this problem [108]; accordingly, the amount of research concerning link spam is increasing [79, 10, 68].

In this Chapter we follow a language modeling approach to detecting link spam in blogs and similar pages. Our intuition is simple: we examine the use of language in the blog post, comments posted to it, and pages linked from the comments. In the case of comment spam, these language models are likely to be substantially different: the spammer is usually trying to create links between sites that have no semantic relation, e.g., a personal blog and a gambling site. We exploit this divergence in the language models to effectively classify comments as spam or non-spam. The approach is aimed at the single blogger, who is often unable to utilize complex link spam detection mechanisms requiring web-scale connectivity knowledge. However, it can also be deployed by web search engines inspecting a blog page containing comments.

5.2 Related Work

5.2.1 Combating Comment Spam

Most approaches to comment spam filtering are technical in nature, and include:

- Requiring registration from comment posters;
- Requiring commenters to solve a *captcha*—a simple Turing test mechanism [302];
- Preventing links, or HTML code in general, in comments;
- Preventing comments on old blog posts;
- Using lists of forbidden or allowed IP addresses for commenters ("blacklists" and "whitelists");
- Using internal redirects instead of external links in comments;
- Limiting the number (or rate) of comments being added ("throttling").

While some of these mechanisms can be moderately effective, they also have disadvantages. Methods which require user effort such as registration reduce the number of spontaneous responses which are important to many blog maintainers. Additionally, they do not affect the millions of commented web pages already "out there," and only address new comments. Preventing commenting altogether, or limiting it to plain text, or enforcing redirects on links in it, limits also legitimate comments and links contained in them, reducing the effectiveness of link analysis methods. Blacklists and whitelists require constant maintenance, and are bypassed by spammers using proxies, hijacked machines, and spoofed legitimate IP addresses. Throttling can reduce the amount of spam from a single page, but not the phenomenon altogether; spammers will simply post to more blogs to compensate for the limited amount of comments per blog. All in all, spammers have found ways around any of these technical obstacles, and it is likely that, as in the case of email spam, technical solutions alone will not eliminate the problem completely.

In 2005, a number of major search engines including those operated by Yahoo, Microsoft and Google announced that they were collaborating with blogging software vendors and hosts to fight comment spam using a special attribute added to hypertext links [225]. This tag, `rel="nofollow"`, tells search engines that the links are untrusted, and allows them to exclude it or handle it differently when performing link analysis. However, the usage of this attribute is problematic for a number of reasons, including harming the inter-linked nature of blogs and possible abuse by webmasters; indeed, it is disputed within the blogging community [226, 18], including by those working for the search engines proposing it [19].

5.2.2 Content Filtering and Spam

A different set of approaches for fighting comment spam works by analyzing the content of the spam comment, and possibly also the contents of pages linked by the comment (e.g., [211]). All these techniques are currently based on detecting a set of keywords or regular expressions within the comments. This approach suffers from the usual drawbacks associated with manual sets of rules, i.e., high maintenance load as spammers are getting more sophisticated, and false alarms (e.g., not all pharmacy-related comments are spam). Typically, content-based methods require training with a large amount of spam and non-spam text, and correcting mistakes that are made; failure to continuously update the learner will decrease its accuracy, as it will create an inaccurate view of what is spam and what is legitimate. While regular expression based methods may provide relief from simple comment spammers, as they enhance their methods these rule-based approaches become less effective, and methods based on deeper analysis (such as the text classification methods successfully employed today for email spam detection) are required.

Published work on content-based spam filtering refers mostly to email spam, which existed long before comment spam, and was therefore targeted by industrial and academic research alike. In the email domain, machine learning and language modeling approaches have been very effective in classifying spam [92, 12]. But an important difference between email spam and comment spam stems from the fact that comment spam is *not intended for humans.* No comment spammer actually expects anyone to click on the link that was added: this link is meant solely for the purpose of being followed by web crawlers. Thus, the spammer can (and does) use any type of words or other features in his comment: the main goal is to have the link taken into account by search engine ranking schemes, and strings which have been reported as good discriminators of email spam such as over-emphasized punctuation [260] are not necessarily typical of comment spam.

5.2.3 Identifying Spam Sites

An altogether different approach to spam filtering is not to classify individual links as spam links or legitimate links, but to classify pages or sites as spam;

work in this area includes usage of various non-content features [228, 79], link analysis methods [10, 28, 31], and trust propagation [102]. However, effective applications of most of these methods requires full connectivity knowledge of the domain or access to a large training set—something which is beyond the abilities of most bloggers.

In comparison to other approaches, ours requires no training, no maintenance, and no knowledge of additional information except that present on the commented web page.

5.3 Comment Spam and Language Models

We now outline our language model based approach to identifying comment spam.

In the previous section, we noted that email spam is easier to classify than comment spam since it tends to have characterizing features, which are supposed to convince a human to respond to the spam mail. On the other hand, the task of filtering comment spam also has an advantage over email spam classification. While every email needs to be classified as spam in an isolated manner, blog comments are presented within a *context*: a concrete semantic model in which the comment was posted, namely, the blog post it is referring to. Our main assumption is that spam comments are more likely to violate this context by presenting completely different issues and topics, compared with legitimate comments. We instantiate the semantic models of the context, the comment and the page linked by the comment using language models.

5.3.1 Language Models for Text Comparison

We identify three types of languages, or language models, involved in the comment spam filtering task (see Figure 5.1). First, there is the model of the original blog post. Then, every comment added to the post adds two more models: the language used in the comment, and the language used in the page linked by the comment.

Intuition suggests that the contents of a post, a comment to it, and a page linked by the comment (if such a link exists) are related; the comment is, after all, a reaction to the post. To measure the similarity between the models we use KL-divergence, first smoothing the model of each language. The smoothing method we use is Jelinek-Mercer interpolation, a linear combination of the given language model with a background, more general one. After smoothing, the probability of an n-gram x is

$$p(x) = \lambda \cdot p(x|\Theta_{\text{orig}}) + (1 - \lambda) \cdot p(x|\Theta_{\text{background}}),$$

where Θ_{orig} and $\Theta_{\text{background}}$ are based on maximum likelihood estimates from the text being modeled and from a larger, more general corpus, respectively. In our

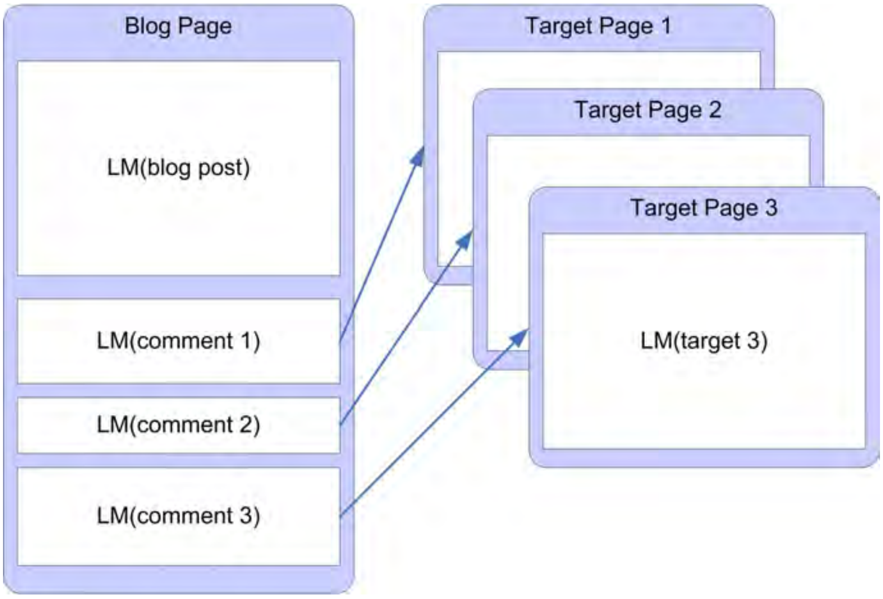


Figure 5.1: Three types of language models: the model of the blog post, the models of the comments, and the models of the pages linked by the comments.

case, the background model is based on word frequencies on the web; these were obtained from the Web Term Document Frequency Project [70]. Any pair of language models from the triplet ⟨blog post, comment, page linked by comment⟩ can be compared.

5.3.2 Spam Classification

Once we have language models for all components and their pairwise similarity scores, we can use these scores to classify the comments as spam or non-spam. A simple approach to this would be to set a minimal similarity value between a comment and a post (or between the page linked to it and a post, or a combination of them); but this is problematic, as the similarity threshold may change between posts. Because of this, we take a different approach, viewing the spam and legitimate comments as two clusters, and using a text clustering approach to distinguish between them. To visualize this, let us return to Figure 5.1: this is how a commented blog page appears to a search engine trying to assess which links (from the comments) should be used for link analysis methods and which not—what is the "legitimate" comment cluster, and what is the spam one.

To distinguish between the clusters, as a first stage, the KL-divergence between all comments (or linked pages) and the blog post are calculated as described earlier. The values obtained can then be seen as drawn from a probability distribution which is a mixture of Gaussians: each Gaussian represents a different language model. The Gaussian with the lowest mean—the least distance from the language model of the original blog post—represents the language model which is closest to the original post. Subsequent Gaussians represent language models which have a larger deviation from the original one, and are therefore more likely to constitute spam comments.

As outlined earlier, we assume the KL-divergence scores to be drawn from a 2-Gaussian distribution: the "legitimate" language model, and all other (spam) models; an example of the distribution in one of the blog pages from our corpus (which is described in Section 5.4) appears in Figure 5.2. To estimate the parameters of the Gaussians, we use the EM algorithm; this is similar to the clustering approach described in [183].

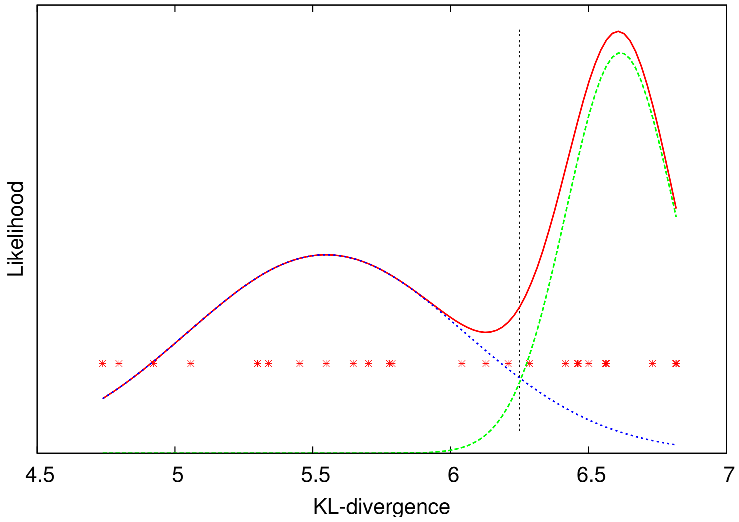


Figure 5.2: Gaussian mixture model estimated from the KL-divergence values of 28 comments on a blog post (solid line), and its underlying Gaussians (dashed lines); the KL-divergence values themselves appear as points.

Finally, a comment is classified as spam if its KL-divergence from the blog post is more likely to be drawn from the spam Gaussian than from the legitimate one. For this purpose, we calculate a discriminating value between the two Gaussians—a number for which lower values are more likely to be drawn from the Gaussian with the lower mean, and higher values are more likely to be drawn from the other Gaussian. Visually, this threshold can be viewed as the best vertical separator between the two distributions. Note that this threshold value provides us with an easy mechanism for changing the likelihood of identifying false positives ("legitimate" comments classified as spam) and false negatives (unidentified spam). Decreasing the threshold ("moving" the separator to the left) will result in a more strict requirement from the language model divergence between the comment and the post, effectively increasing the number of false positives and reducing false negatives; increasing the threshold value ("moving" the line to the right) will cause our method to be more tolerant to higher KL-divergence values, reducing false positives at the cost of increased false negatives. Usually, the cost of false positives is considered higher than that of false negatives; in general, we can use a *threshold multiplier* value to adjust the original threshold, with values lower than 1 “moving” the separator to the left and values over 1 “moving” it to the right. In Figure 5.2, the optimal separator (when the threshold multiplier is 1) appears as a vertical line.

5.3.3 Model Expansion

Blog comments can be very short, and this is true also for some blog posts. This results in sparse language models, containing a relatively low number of words. We therefore propose to enrich the models of both the post and the comment, to achieve a more accurate estimation of the language model. An intuitive way to do so is to follow links present in the post and the comment, and add their content to the post and the comment, respectively; in the case of the post, it is also possible to follow incoming links to the blog and add their content. This is similar to the expansion of a blog post with additional sources of data we discussed in Chapter 3, to improve ad matching. Taking this expansion a step further, it is also possible to continue following links up to depth N, although this potentially causes topic (and language model) drift.

5.3.4 Limitations and Solutions

An easy way for spammers to “cheat” our model (or any other model which compares the contents of the post and the comment) is to generate comments using a similar language as that used in the original blog post. This makes the link-spam bots slightly more complicated since they must identify the post contents and use its model for generating a close one (e.g., by copying terms or sentences from it)—but spammers have shown to overcome higher obstacles.

But while this approach reduces the divergence between the comment and the post, it increases the divergence between the comment and the page linked by it; when comparing the models of the comment and the landing page, language divergence will still distinguish between related and unrelated comments. Additionally, in this case, a new opportunity for spam filtering arises. Assuming the spammer posts multiple comments to many different blogs (as a means of increasing link-based authority scores), and in each case indeed matches the language of her comment to the language of the post to which it responds, there are now many comments with completely different language models linking to the same spam site. This is easily detectable at the search engine level, where full connectivity information is known; it can also be detected by an iterative HITS-like method by the blogger, following the link to the spam site and then its incoming links.

As any other spam filtering method, ours is not foolproof and can make mistakes; comments formulated with a sufficiently different vocabulary than the blog post might be mistaken for spam. However, this is precisely the reason for the robustness of our approach: it is very hard for the spammer to create comments that will be both similar to the blog language and to the spam site language. To account for these mistakes, an alternative to using this method as a binary spam/non-spam classifier is to use it to assign weights to links found in comments according to their language model divergence; the weight can be used to decrease link-analysis scores of malicious sites, using methods such as the one reported in [20].

5.4 Evaluation

We now discuss experiments performed to identify comment spam in blogs using our approach. Aside from evaluating the success of our approach, we are interested in the effects of the parameters of our models (the smoothing value and the threshold used to separate the Gaussians) on its performance.

We collected 50 random blog posts, along with the 1024 comments posted to them; all pages contained a mix of spam and non-spam comments. The number of comments per post ranged between 3 and 96, with the median being 13 (duplicate and near-duplicate comments were removed). We manually classified the comments: 332 (32%) were found to be "legitimate" comments; the other 692 comments (68%) were spam comments.[1]

The spam comments we encountered in our corpus are of diverse types; while some of them are simple keyword lists, accompanied by links to the spam sites, others employ more sophisticated language (see Figure 5.3 for some sample comments from the collection). A typical blog page from our corpus contains a mix of different types of comment spam.

In experiments reported here, we compared only the language models of the post and the comments, and did not take into account the model of the page linked to by the comments.

5.4.1 Results

We conducted two sets of experiments. The first set was aimed at assessing the overall effectiveness of our approach, and, in particular, measuring the effect of smoothing on the results; the second set of experiments examines the trade-off between reducing the number of false positives and reducing the number of false positives.

[1] This is a relatively high amount of comment spam, compared with an estimate of 40% comment spam reported by Akismet in [214]; however, as noted in [214], the sample is biased towards a particular platform, possibly skewing the reported figures.

```
6708 sports bettingonline sports bettingmarch
madnessbasketball bettingncaa bettingsports ...
```
Link: gambling site

(a)

```
%syn(Cool|Nice|Rulezz)% %syn(blog,|portal|site)% hope to
make %syn(my own|own weblog|my diary)%, not worse than
yours ;)
```
Link: adult site

(b)

```
A common mistake that people make when trying to design
something completely foolproof was to underestimate the
ingenuity of complete fools.
```
Link: pharmacy site

(c)

```
i was looking for plus size lingerie but google sent me
here
```
Link: fashion shop

(d)

Figure 5.3: Samples of different types of comment spam in our collection (top to bottom): (a) keyword based, with a random number to prevent duplicate detection; (b) revealing internal implementation of the spamming agent; (c) using quotes—in this case, from Douglas Adams—as "trusted" language; (d) disguising as random surfer.

As a naive baseline, we use the maximum likelihood probabilities for the comment type in our model; as noted earlier, 68% of the comments were spam, so we assume a fixed probability of 0.68 for a comment to contain link spam. This results in 581 of the 1024 comments (57%) being correctly classified: 112 out of the 332 legitimate comments (34%) and 469 out of the 692 spam ones (68%).[2] We proceed with describing both experiment sets.

Overall success and effect of smoothing. As noted in Section 5.3, we smooth the language models we construct by interpolating them with a background model—in our case, word frequencies on the web. This is done to handle the sparseness of the models, as many of them are based on fairly short texts. To examine how smoothing affects spam classification success, we varied the smoothing parameter λ between 0.5 (where the background model and comment or post models are equally important) and 1.0, (no background model is used at all); results are shown in Figure 5.4. Improvement over the baseline is clear: the optimal

[2] Using a maximum likelihood estimator identifying all comments as spam would yield a baseline with higher overall performance (68%), but with a 100% false positive rate.

smoothing parameter, 0.85, achieves 83.5% success over all comments—spam and non-spam. The effect of smoothing is typical; performance peaks with a smoothing parameter close to 1, and gradually decreases as more, or less, smoothing is used. In this set of experiments, the optimal separator was used between the two Gaussians (i.e., the threshold multiplier was set to 1).

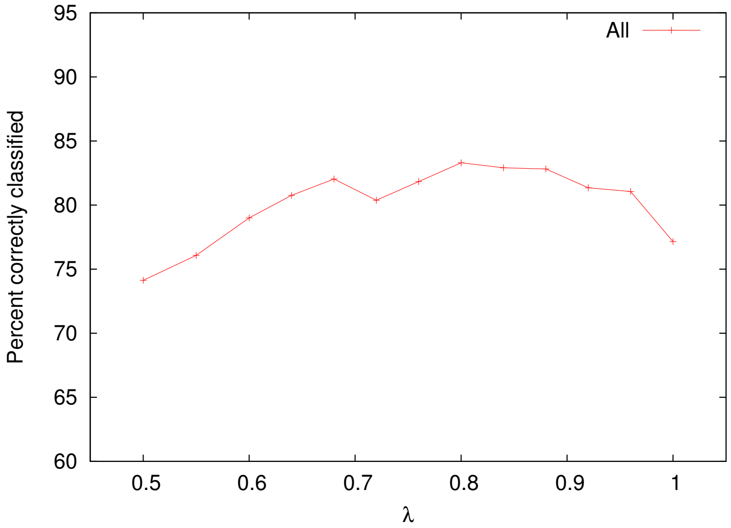


Figure 5.4: Effect of smoothing on comment spam classification.

Optimizing for different scenarios. The second set of experiments demonstrates the trade-off between minimizing the false negatives and minimizing the false positives. False negatives are spam comments which were not identified as spam by our method, while false positives are non-spam comments which were classified as spam. As noted in Section 5.3.2, the threshold multiplier is a value used to modify the separator between the two Gaussians estimated from the language models: a multiplier of 1 means the optimal theoretic separator is used. By multiplying the optimal threshold with a value lower than 1 we effectively move the separator towards the Gaussian with the lower mean, indicating a stronger restriction on the allowed divergence from the language model of the post. Similarly, a multiplier higher than 1 results in a more lenient approach. By varying this multiplier, we can decrease the number of false positives at the cost of increased false negatives, and vice cersa. Figure 5.5 shows the percentage of correctly-classified comments, as well as the number of false negatives and false positives, as obtained by varying the multiplier between 0.75 and 1.25 (the smoothing parameter used was 0.85). While overall performance remains more-or-less fixed, with accuracy rates in the low 80s, as the threshold multiplier increases the percent of correctly identified spam comments decreases, and the percent of correctly identified non-spam comments increases.

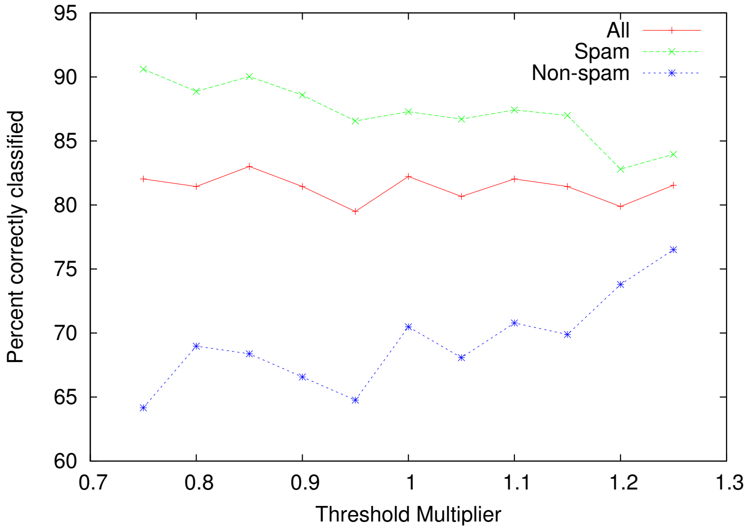


Figure 5.5: Trade-off between correctly classifying spam and correctly classifying non-spam, as the threshold multiplier varies.

5.4.2 Discussion

While the size of our test collection is relatively small, our results are encouraging and clearly show that our intuition is correct: the language used in spam comments does diverge from the language of the blog post substantially more than the language used in legitimate comments.

An analysis of the misclassified comments reveals that many of them are very short—containing 3 or 4 words, usually a non-content response to the post (e.g., "That sounds cool"). However, the vast majority of these comments contain no external links, or an email link only—so their misclassification will not result in actual search engine spam (in the case of false negatives) and not change the "true" link-analysis prestige of pages (in the case of false positives). While it is possible to integrate language divergence with comment length and other features into a hybrid comment spam classification system, we focused on the language aspect only and did not explore usage of additional knowledge.

5.4.3 Model Expansions

As mentioned earlier, a possible solution to the sparseness of some of the blog posts is to expand the language model in various ways. We experimented with such expansions by following all links present in the blog post and adding the content present in the target pages to the content of the blog post, before estimating the language model. Of the 50 blog posts in our corpus, 31 posts had valid links to other pages (some posts did not contain links at all, and some contained expired and broken links). The average number of links followed (for the 31 pages with expansions) was 3.4. However, usage of the expanded models did not improve overall classification accuracy. In fact, while for some blog posts—most notably shorter ones—the expansion helped substantially, we experienced a degradation of 2%-5% in the average performance over the entire corpus. This suggests a possible strategy of using expansion for short posts only, those that are likely to benefit from it.

5.5 Conclusions

We presented an approach to blog comment spam filtering which utilizes the difference between the language used in a blog post and the language used in the comments to that post (and, potentially, pages linked from those comments). Our method works by estimating language models for each of these components, and comparing these models using standard divergence methods. Experiments using our method to classify typical comment spam show promising results; while we apply this to blogs, the problem and the solution are relevant to other types of comment spam, such as wiki spam.

Going back to our research questions, we show that language modeling comparison is a viable method for spam detection in blog comments, as spam in this domain is characterized by diverging models. While this approach does not guarantee to completely block spam, it can be combined with other measures, such as those reviewed in Section 5.2, to form a more complete solution. Finally, varying the extent to which the model of a comment is allowed to diverge from the model of the post enables us to tune our method to minimize either false negatives or false positives, and adapt it to different scenarios.

Conclusions for Part I

This part set out to explore text analysis at the level of individual blogs, focusing on analytics tasks which utilize properties unique to the blogspace. We discussed blog profiling for product and ad matching, classification and categorization tasks of distinct blog features such as moods and tags, and, finally, a method for addressing a form of spam specific to blogs.

Two themes recurred throughout the work presented in this part: the use of statistical language models, and the view of blogs as semi-structured documents, consisting of different components. We demonstrated that simple text analysis approaches, such as language model divergence, prove powerful when combined with the semi-structured approach to blog content analysis, both when the different blog components are combined (e.g., for profiling) or contrasted (e.g., spam filtering).

An additional observation is that, at least on the individual level, text analysis of blogs is not trivial. Mood classification proved to be a complex task; other tasks such as product and contextual advertising, for which success was higher—even favorable compared to state-of-the-art approaches for the same tasks—still leave much to be desired.

The main reason for the complexity of tasks addressing a single blog is not necessarily the method of analytics being used, but the sparseness of data. In many cases, individual blogs, and certainly individual blog posts, simply do not have enough text for meaningful applications of some analysis approaches. In the next part we move from work on single blogs to the analysis of multiple blogs, substantially enriching the text models involved. As we will demonstrate, analysis at the aggregate level offers new possibilities for mining knowledge from blogs—and new challenges too.

Part II

Analytics for Collections of Blogs

In an analysis of a full year's worth of blog posts, Doran et al. state that "the vast majority of blogs are uninteresting to most people most of the time" [71]. While this may be the case, this part will show that whereas individual blogs are of limited interest to most people, aggregate knowledge that can be mined from collections of blogs, and sometimes from the entire blogspace, is potentially useful to large audiences.

As in the previous part, we focus our attention on properties of blogs which are relatively unique to the medium: the subjective, personal language used, and the ability of blog readers to interact with blog authors. There are two chapters in this part: Chapter 6 revisits affect analysis in blogs, this time at the aggregate level. We show that the blogspace provides insights into global emotional behavior, supporting observations which are difficult to obtain through other sources. Additionally, we demonstrate that mining this type of data is useful for commercial applications, and in particular, business intelligence. Chapter 7 examines comments in the blogspace—a domain often neglected in computational analyses of blogs; it shows that existing analytics for the blogspace are incomplete when comments are discarded, and demonstrates new types of analysis which are enabled with the presence of comment data.

Chapter 6

Aggregate Sentiment in the Blogspace

Sentiment analysis is a complex task; typical performance in this domain is lower than that achieved in other, more straightforward text classification tasks, such as topical categorization. We have already observed, in Chapter 4, that classifying the mood of a blog post is hard; more subtle expressions of sentiment—e.g., irony or sarcasm—remain a challenge for current technology (as well as for many humans). But, as with many other computational linguistics tasks, overall performance of sentiment analysis techniques increases as more data is available: more training examples contribute to a better model, and longer texts in the training or test sets contribute to stability and accuracy of the method.

In this Chapter we turn to mining the personal, sentiment-rich language of blogs—this time at an aggregate level. Instead of analyzing a single post or blog, we look at the sentiment as reflected in multiple blogs, or the entire blogspace. We begin by demonstrating why this is useful: Section 6.1 examines the relation between aggregate blogger sentiment and financial results of products, showing that taking the sentiment into account results in better models than those obtained when measuring only the volume of discussion related to a product. The rest of this Chapter continues to explore moods in the blogspace—an area we addressed at the level of single posts in Chapter 4. Section 6.2 demonstrates that important insights can be obtained by observing moods reported by bloggers over many blogs. In Section 6.3 we develop a method for predicting the global mood through the language used by multiple bloggers, and in Section 6.4 we focus on irregular temporal patterns of such moods, and how they can be explained—again using the bloggers' language. As the latter part of the chapter will show, analysis of mood at the aggregate level is more tractable than the corresponding analysis at the level of a single post.

6.1 Blog Sentiment for Business Intelligence

Earlier, we referred to a blog as the "unedited voice of an individual." The entire blogspace, then, can be viewed as the voice of the public: a massive collection of discussions and commentary reflecting people's opinion and thoughts. Part of this discussion is classified as Consumer Generated Media (CGM, [34])—experiences and recommendations expressed about products, brands, companies and services. CGM in blogs presents a double opportunity: for consumers, this type of advice provides direct, unsolicited information that is often preferred to more traditional channels; a survey of online consumers showed that people are 50% more likely to be influenced by word-of-mouth recommendations than by radio or television advertisements [126]. For companies, CGM helps to understand and respond to the consumer by analyzing this informal feedback. This Section focuses on the latter use of CGM.

A relation between the volume of discussion in blogs (the "buzz") and commercial performance of a product has already been observed (e.g., [97]). In addition, sentiment analysis methods for analyzing typical CGM content have improved substantially in recent years, based on the availability of large-scale training data and resources. The main question addressed in this section is whether these two aspects can be combined: more specifically, we aim to discover whether usage of sentiment analysis on blog text in a CGM context results in a better predictor of commercial performance than simple buzz count. To this end, we analyze the sentiment expressed in blogs towards a particular product type—movies—both before the movie's release and after, and test whether this sentiment correlates with the movie's box office information better than a simple count of the number of references to the movie in blogs does. We proceed by describing related work, the data we used in our experiments, and their results.

6.1.1 Related Work

Product reviews are frequently used as the domain in sentiment analysis studies (e.g., [238, 61]); they are focused, easy to collect, and often provide meta-data which is used as ground truth: a predefined scale which summarizes the level and polarity of sentiment ("4 out of 5 stars"). Blogs differ from these studies in that they tend to be far less focused and organized than the typical product review data targeted by sentiment analyzers, and consist predominantly of informal text. Often, a reference to a movie in a blog does not come in the context of a full review, but as part of a post which focuses on other topics too.

A number of studies are closer to the work described here than most product review oriented sentiment work. Good correlation between movie success and blog posts mentioning the movie was established in [291]. However, this study was based on an analysis of five blogs only; furthermore, the tracked blogs are dedicated to movie reviews, and their content resembles professional product review sites rather than typical blog content. It is unclear whether the methodology described scales up to other products and blogs, or whether it is different from simply tracking non-blog product reviews. A large-scale study of blogs and business data, measuring the correlation between the number of blog posts mentioning a product (books, in this case) and its sales rank, is described in [97]. Here, the raw number of product mentions in the blogspace was shown to be a good predictor of sales, but no sentiment analysis was used. Tong [292] and Tong and Snuffin [293] do describe systems which incorporate sentiment analysis for measuring correlation between business information and product mentions, but do not report on empirical results. Finally, in work done in parallel to that reported here, Liu analyzed references to movies in message boards, finding that sentiment is not particularly beneficial as an indicator of movie success [177]. In this study, most movie mentions were observed after the release of a movie; this correlates with our own observation regarding post-release discussions. However, we also analyze, separately, pre-release references to movies—reaching a different conclusion in this case.

6.1.2 Data and Experiments

Our task, then, is to examine the correlation between sentiment expressed in blogs towards a product, and the product's financial success; our product of choice is movies—as mentioned earlier, a popular domain for sentiment analysis studies. We begin by presenting the data and the methodology used for examining this correlation.

Product Set

To measure aggregate levels of sentiment, we first require the studied movie to have some minimal discussion volume in the blogspace: in cases where only a handful of references to a movie exist, there is little meaning to aggregating their infomation. For this reason, we limited our analysis to high-budget movies—requiring a budget higher than 1 million U.S. dollars. During the period between February and August 2005, which is the time span we study, 49 movies with publicly-available financial information meet this criterion; these are the movies used in the work that follows.

Financial Data

Given a particular movie, we used IMDB—the Internet Movie Database[1]—to obtain the date of its "opening weekend" (the first weekend in which the movie played in theaters), as well as the gross income during that weekend and the number of screens on which the movie premiered. We focus on the opening

[1] http://imdb.com

weekend data rather than total sales since this normalizes the figure across movies that were released on different dates, preventing earlier movies from having a higher total income just because they have been "out there" longer, have been already released on DVD, etc. Opening weekend income correlates highly with total movie income, accounting for an estimated 25% of the total sales [283]. The number of screens the movie premiered on was used to normalize the opening weekend income, producing an "Income per Screen" figure for each movie. This allows comparing sales of summer blockbuster movies, sometimes released to 4,000 screens simultaneously in the opening weekend, with lower-profile movies released to 1,000–2,000 screens.

Blog Data

For each movie, we collected all related blog posts appearing in the Blogpulse [89] index, a large-scale index covering the majority of the blogspace. A post was considered related to a movie if the following conditions were true:

- The date of the post is within a window starting a month prior to the movie's opening weekend date and ending one month after it.
- The post contained a link to the movie's IMDB page, *or* the exact movie name appeared in the post in conjunction with one of the words ⟨movie, watch, see, film⟩ (and their morphological derivatives).[2]

Methodology

For each relevant post, we extracted the contexts in which the movie was referenced by taking a window of k words around the hyperlinks to the movie's IMDB page, or around exact matches of the movie name; we used varying values for k, from 6 words to 250. Then, we calculated the sentiment polarity and intensity of the contexts using the state-of-the-art method described in [222]. In this approach, a domain-specific lexicon of sentiment values is used to tag each token in the text; we used a lexicon developed for the movie domain. Additionally, the text is chunked, and predefined expressions of polarity are located in the higher-level groups of tokens; finally, polarity information is propagated to the entire sentence level, using heuristics to handle negation, subsumption, and so on. This method has been shown to perform only a few percent worse than human performance on the same task [222].

Examples of different context lengths for the same reference to a movie are shown in Table 6.1, along with the computed polarity; this demonstrates the

[2] This strategy aims for high precision without overly sacrificing recall. An examination of the retrieved posts shows a high degree of precision. We did not explicitly measure recall, but did find that using a combination of an IMDB link query and text queries increased recall by a factor of 500% over simply using the IMDB link query, which has near-perfect precision but limited recall.

Length (k)	**Content**	**Sentiment**
8	Rach went to see “The Sisterhood of the Traveling Pants”, we both read the	Neutral
40	gym or work. 2. It was the first time I laughed since I’ve been home. Me and Rach went to see “The Sisterhood of the Traveling Pants”, we both read the book and the girl from Gilmore Girls was in it. I had the best time	Positive
120	Tonight was a night of firsts. 1. I’m pretty sure it was the first time I left the house and didn’t go to the gym or work. 2. It was the first time I laughed since I’ve been home. Me and Rach went to see “The Sisterhood of the Traveling Pants”, we both read the book and the girl from Gilmore Girls was in it. I had the best time with her. We laughed, were both such dorks. The movie was SOOO sad. Like true “The Notebook” quality. I enjoyed it and it got me thinking. I need to stop being so miserable. I make my time here suck. I	Negative

Table 6.1: Polarity of different contexts.

possible differences in polarity estimation when using “too much” or “too little” context.

In summary, for every item m in the set of 49 movies we have the following information:

- d_m: the opening weekend date of m;
- e_m: earnings from ticket sales for m during the opening weekend;
- s_m: the number of screens m featured on in the opening weekend;
- R: a collection of references to m in blog posts. For each $r \in R$, we also have d_r—the date of the post containing r; and $p_{k,r}$, the polarity value of the k words surrounding r, where k values vary between 6 and 250.

A sample item is shown in Table 6.2. Note that the polarity score is fitted to a log-linear distribution, with the majority of scores falling within a range of 4 to 7 [223]. Thus, the average polarity score of 5.5 for the movie in the table indicates significant positive overall sentiment.

Using the d_r values, we can partition R into two subsets: R_{pre} which is all references made to m prior to its release (i.e., $d_r < d_m$), and R_{post}—all references made after the release ($d_r \geq d_m$). Then, we can measure the correlation between $|R_{pre}|$ or $|R_{post}|$ and the “Income per Screen,” e_m/s_m, as well as measure sentiment-related correlations that take into account the polarity values, p_r.

Movie	The Sisterhood of the Traveling Pants
Opening Weekend (d_m)	5 June 2005
Opening Weekend Sales (e_m)	\$9.8M
Opening Weekend Screens (s_m)	2583
Income per Screen (e_m/s_m)	\$3800
Pre-release Data	
References in blogs ($\|R_{pre}\|$)	1773
Context Length: 10 words	
- Positive references	329
- Negative references	50
- Mean sentiment polarity	5.5 / 10
Context Length: 20 words	
...	...
Post-release Data	
References in blogs $\|R_{post}\|$	1618
...	...

Table 6.2: Sample data from our collection.

Experiments

At this stage, we have some indicators of the movie's success (Income per Screen and raw sales figures), as well as a range of sentiment-derived metrics such as the number of positive contexts, the number of negative ones, or the total number of non-neutral contexts. The natural way to determine whether the two sets of information are related is to measure the statistical correlation between them; we use Pearson's r-correlation for this. In addition to measuring the statistical correlation between the sentiment-related measures and the movie success information, we measure the correlation between the raw counts of occurrences in blogs (the "buzz") and the financial information: comparing the two correlations will address the main question in this section: whether sentiment information improves on volume counts only for this type of task. Measurement was done separately for pre-release contexts and post-release ones.

Raw counts vs. sentiment values. Our first observation is that usage of the sentiment polarity values, given an optimal context length, results in better correlation levels with movie success than the raw counts themselves for data gathered *prior* to the movie's release. For data gathered *after* the movie's release, raw counts provided a better indicator. Of the different polarity-based measures used in our experiments, those yielding the best correlation values were as follows:

- Prior to the movie release: the number of positive references, within a relatively short context (the optimal value was 20 words).
- After the movie release: the number of non-neutral references within a relatively long context (the optimal length was 140). Using the number of

positive references achieved very close results to this.

Table 6.3 compares the correlation between movie business data for raw counts and for the best performing polarity-related metrics. Clearly, the sentiment-based correlation improves substantially over the raw counts for pre-release data, whereas for post-release data the effect is negative (but minor).

Correlation	**Between . . .**	**Period**
0.454	Raw counts and income per screen	Pre-release
0.509 (+12%)	Positive contexts and income per screen	
0.484	Raw counts and sales	
0.542 (+12%)	Positive contexts and sales	
0.478	Raw counts and income per screen	Post-release
0.471 (-1%)	Non-neutral contexts and income per screen	
0.614	Raw counts and sales	
0.601 (-2%)	Non-neutral contexts and sales	

Table 6.3: Comparison of correlation between movie business data and blog references, with and without use of sentiment. Context sizes used: 20 (pre-release), 140 (post-release).

While the improvement using sentiment values on pre-release data is in-line with intuition, it is unclear to us why it does not have a similar effect for post-release data. One possible explanation is that post-release contexts are richer and more complex, decreasing the accuracy of the sentiment analysis.

Context length. Our next observation is that constraining the context being analyzed to a relatively small number of words around the movie "anchor" is beneficial to the analysis of pre-release polarity metrics, but reduces the effectiveness of the post-release metrics. Figure 6.1 displays the relation between the correlation values and the context length for two particular instances of analysis: the correlation between the number of positive contexts before the movie release and the income per screen, and the correlation between the number of non-neutral contexts after the release and the opening weekend sales for the movie (note that the context length is plotted on a log-scale).

Examining the contexts extracted both before and after the movie's release, we observed that references to movies before their release tend to be relatively short, as the blogger typically does not have a lot of information about the movie; usually, there is a statement of interest in watching (or skipping) the movie, and possibly a reaction to a movie trailer. References to movies after their release are more often accounts of the blogger's experience watching the movie, containing more detailed information—see an example in Table 6.4. We hypothesize that this may be an explanation for the different effect of context length on the correlation quality.

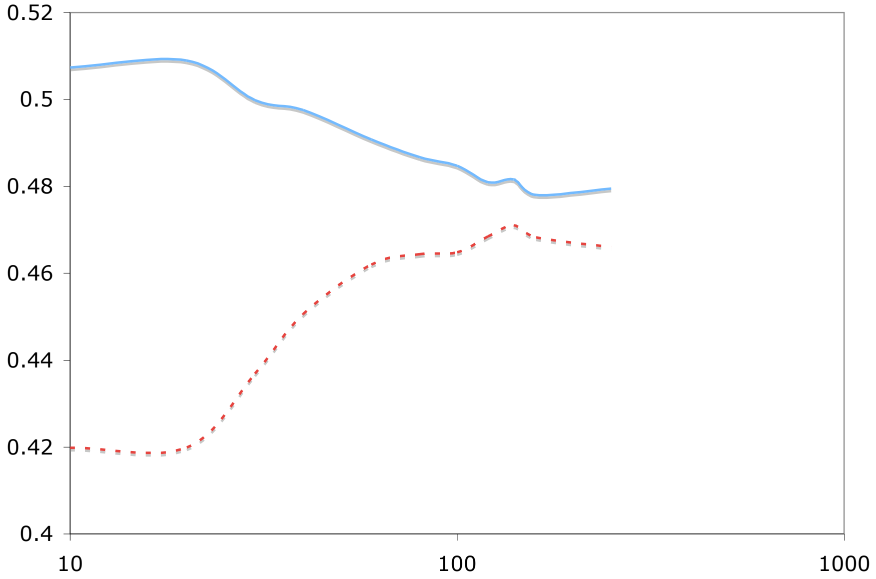


Figure 6.1: Relation between context length and correlation to income per screen: positive references, pre-release (blue, solid line) and non-neutral references, post-release (red, dashed line). The X-axis shows the context length (on a log-scale), and the Y-axis shows the level of correlation.

Breakdown

Out of the 49 movies in our study, over half have very good correlation between pre-release positive sentiment and sales. Less than 20% can be viewed as outliers: movies whose average Income per Screen was poorly predicted by pre-release sentiment. How can the low correlation between blog opinion and business data be explained for these outliers? Movie sales have been shown to be affected by many factors unrelated to online discussion, such as genre, Motion Picture Association of America rating, other movies released at the same time, and so on [283]. On top of that, noise originating from different components of our analysis—the retrieval of posts from the collection of all posts, the polarity analysis, and so on—accumulates, and may destabilize the data.

Cursory examination of outliers in our experiments, both those that overestimate sales and those that underestimate them, did not yield any obvious feature shared by the irregular data points.

6.1.3 Conclusions

The purpose of this Section was to motivate our work on aggregate sentiment analysis in blogs. To this end, the question we set out to investigate was whether taking into account the language used in blogs towards products—and, more

apparently an early easter is bad for apparel sales. who knew? i'll probably go see "guess who?" this weekend. i liked miss congeniality but the sequel [link to IMDB's page for "Miss Congeniality 2"] looks *awful*. and seattle's too much of a backwater to be showing D.E.B.S. i had to wait forever to see saved! too. mikalah gordon got kicked off american idol last night. while she wasn't the best singer, i wish ...
Monday, March 28, 2005 - Miss Congeniality 2: Armed and Fabulous. I know this is overdue, but I wanted to use this opportunity to discuss an important topic. The issue at hand is known as the Sandra Bullock Effect (SBE). This theorem was first proposed by my brother, Arthur, so he is the real expert, but I will attempt to explain it here. The SBE is the degree to which any movie becomes watchable simply by the presence of a particular actor or actress who you happen to be fond of. For example, if I told you that someone made a movie about a clumsy, socially awkward, dowdy female police officer who goes undercover as a beauty pageant contestant to foil some impenetrable criminal conspiracy, you'd probably think to yourself, "Wow that sounds pretty dumb." And you'd be right. However ...

Table 6.4: Typical references to movies in blogs: pre-release (top), and post-release (bottom).

concretely, using sentiment analysis to classify this language—results in a better prediction of the financial success of the products than measuring only the amount of discussion about them in the blogspace. The answer to this question is positive: in the domain of movies, there is good correlation between references to movies in blog posts—both before and after their release—and the movies' financial success; but shallow sentiment analysis methods can improve this correlation. Specifically, we found that the number of positive references to movies in the pre-release period correlates better with sales information than the raw counts in the same time period.

By itself, the correlation between pre-release sentiment and sales is not high enough to suggest building a predictive model for sales based on sentiment alone. However, our results show that sentiment might be effectively used in predictive models for sales in conjuction with additional factors such as movie genre and season. More generally, we conclude that aggregate sentiment analysis in the blogspace is useful for this type of tasks; blogs provide a unique source of CGM information through their personal, unmediated nature, and their sheer scale.

6.2 Tracking Moods through Blogs

We have already discussed moods in blogs in Section 4.1, which focused on identifying the mood of a given blog post. In the rest of this Chapter we return to blog moods, this time at the aggregate level. The mood assigned to a single post gives an indication of the author's state of mind at the time of writing; a collection of such mood indications by a large number of bloggers at a given point in time provides a "blogspace state-of-mind," a global view of the intensity of various feelings among people during that time. The "long tail" of the blogspace was described in Section 2.2.2: it is the body of blogs which comprises the majority of the blogspace. This section examines the the mood indications of the individuals which are part of this long tail, as expressed through their blogs over time.

Tracking and analyzing this global mood is useful for a number of applications. Marketers and public relation firms would benefit from measuring the public response to introductions of new products and services; political scientists and media analysts may be interested in the reaction towards policies and events; sociologists can quantify the emotional echo of an event throughout the crowds.

The rest of this Chapter offers methods for analyzing blog content to predict mood changes over time and reveal the reasons for them. But before applying text analytics to mine mood-related knowledge, we motivate our work by demonstrating the type of insights gained. All the examples we give, as well as later work in this Chapter, are based on mood-tagged blog posts from LiveJournal; additional information about this collection and the means of obtaining it is given in Appendix B.

6.2.1 Mood Cycles

One clear observation about fluctuation of moods over time is the cyclic, regular nature of some moods. Figure 6.2 shows three examples, plotting the prevalence of three moods over time: *sleepy*, which has a clear 24-hour cycle;[3] *drunk*, with a weekly cycle (drunkenness peaks on weekends); and the yearly cycle of *stress*.[4] In the latter example, stress is low during the summer, and increases substantially throughout the autumn (as many of the bloggers are high-school or college aged [178], this correlates well with the start of the school and academic year). In the period prior to Christmas, stress increases substantially; reasons

[3] Although the blog posts in the collection come from various time zones around the globe, in 2005—the period from which most data in this Chapter is used—more than 80% of the LiveJournal users for which country information was available were from North America [178]. This means that the vast majority of posts use the same 4-5 time zones, and that skew relating to differences in zones is limited.

[4] In these and subsequent graphs of moods in the blogspace, the X-axes mark the time, and the Y-axes show the percentage of blog posts manually annotated by their authors with a given mood.

may include exams and end-of-year preparations. During the holiday period itself, stress decreases to sub-summer levels, but quickly returns to earlier levels. Note the pulsating nature of stress throughout the entire period: this is due to a secondary cycle in stress, a weekly one (stress levels decrease substantially during the weekends).

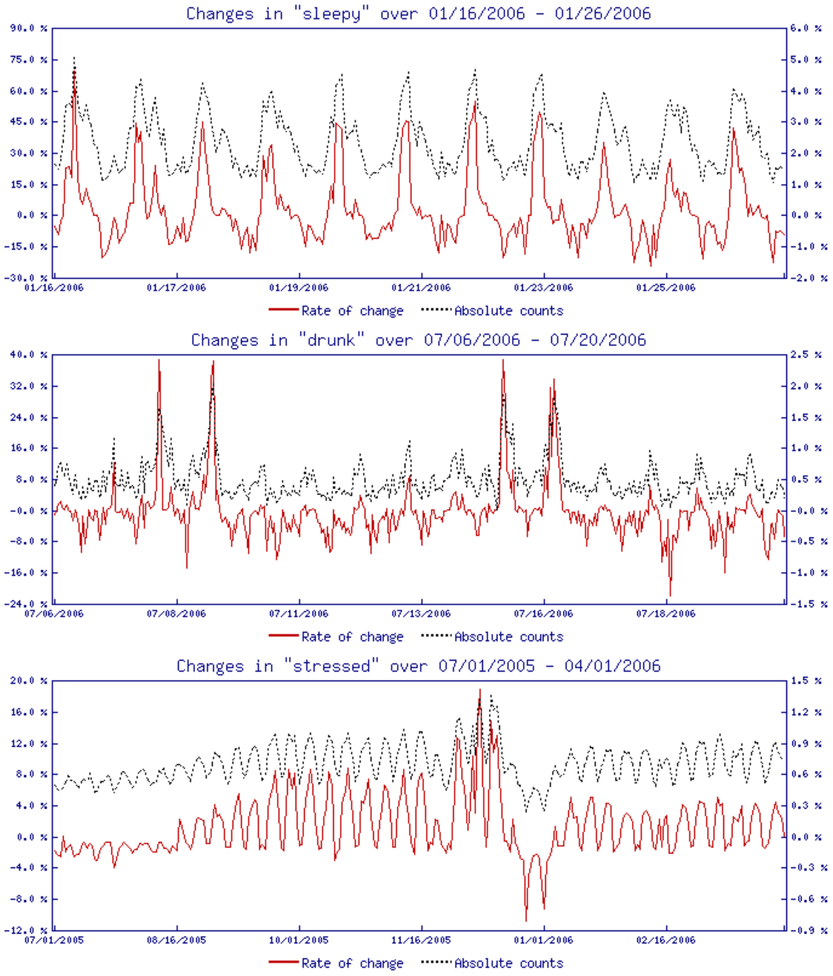


Figure 6.2: Regularities of mood fluctuation. Top: daily cycle of *sleepy* over a period of 10 days. Center: weekly cycle of *drunk* over a period of 14 days. Bottom: yearly pattern of *stress* over 8 months, slowly increasing toward the end of the year and rapidly decreasing as the new year starts. Note that the time span between tick marks is different in the different graphs.

Other moods show a less regular behavior over time. For example, Figure 6.3 shows the changes in the mood *cold*, peaking over the winter during particularly cold periods.

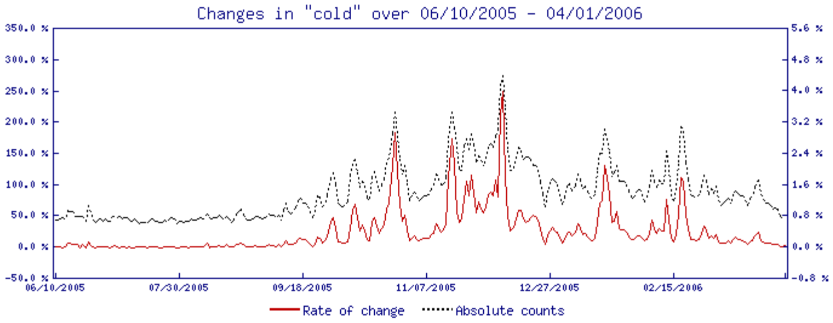


Figure 6.3: Long-term irregularities of mood behavior: *cold* peaks occasionally appear during the winter.

6.2.2 Events and Moods

Global events are clearly reflected in the mood of the blogspace. Figure 6.4 shows examples for the reaction of bloggers to two considerably major events: a peak of *sympathy* towards victims of bombings in London on July 7th, 2005, and a similar, prolonged sense of *worry* after Hurricane Katrina hit New Orleans two months later.

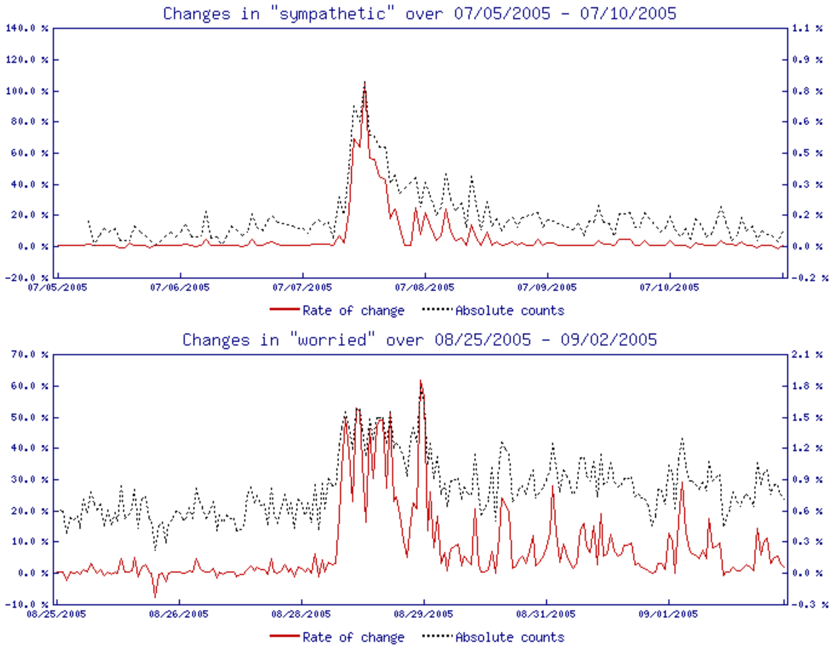


Figure 6.4: Global moods respond to global events. Top: *sympathy* spikes after bombs explode in London on July 7th, 2005. Bottom: elevated levels of *worry* as Hurricane Katrina strikes the Gulf Coast of the United States.

Placing the behavior of different moods side by side emphasizes relations between them: in a rather esoteric example, Figure 6.5 shows how a feeling of *hunger* growing as the American Thanksgiving meal approaches is replaced by a satisfied post-dinner *full* feeling.

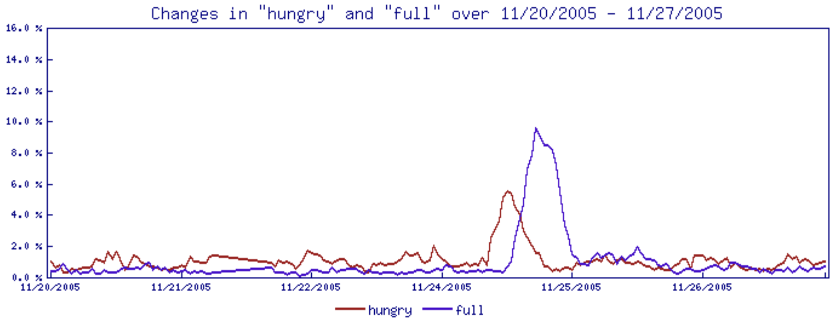


Figure 6.5: Comparing mood changes: *hungry* vs. *full* around Thanksgiving 2005.

While these may look like straightforward examples—it is clear that people are moved by tragedies and excited about holidays—they hint at the power of aggregate mood analysis. As we will see later, some mood changes are not trivial to understand without some cultural context; and even "anticipated" mood behavior can provide important observations. For example, a comparison of the level of shock or surprise of bloggers towards different events reveals their relative importance to people. Although our data comes primarily from U.S.-based bloggers, the level of sadness following the bomb attacks in London mentioned earlier was higher than the sadness level after Hurricane Katrina hit the United States; both were yet lower than the level of shock following the premature death of a television persona, Steve Irwin, in an underwater accident in September 2006.

We follow by introducing the two text analysis tasks we explore in the domain of aggregate moods in the blogspace: predicting the aggregate mood behavior from the contents of blogs, and identifying the reasons for unusual mood changes.

6.3 Predicting Mood Changes

Many blog posts are "annotated" with moods—but many, many more are not, either because the platform used does not support this annotation or because the blogger chooses not to use it. To understand the true mood of the blogspace—the one reflecting all bloggers, not only those with an explicit mood indications, we need to turn to that component of blog posts which exists for all posts—the contents of the posts. This is the task addressed in this section: not to classify the mood of individual posts, but to determine the aggregate mood levels across the entire blogspace at a given time: the intensity of "happiness" or "excitement" as reflected in the moods of bloggers during a particular time period. The relative intensities of moods as observed in those blog posts which do include a mood indication serve as the ground truth: in a nutshell, we are trying to derive the graphs shown in the previous section from the content of blog posts (not only the manually annotated ones, but a larger set). Our research question is whether this can be done effectively and efficiently.

6.3.1 Related Work

Clearly, the work discussed here continues one of the tasks addressed in Chapter 4, namely, classifying moods of single posts. But the task we face now is different not only in the amount of data analyzed, but also in its transient nature. Moods are a fast-changing attribute, and we focus on estimating the mood levels in a certain time slot.

While research on sentiment analysis is plentiful, work on change of sentiment over time as reflected in a text corpus is scarce. A notable exception is the work of Tong on sentiment timelines, where positive and negative references to movies are tracked in online discussions over time [292]. Outside the sentiment analysis area, work on timelines in corpora is mature, mostly driven by the Topic Detection and Tracking efforts (TDT, [8]) mentioned in Section 2.3.1. Non-sentiment-related trends over time in the blogspace are exposed by some blog search engines such as BlogPulse and Technorati which provide daily counts of terms in their indices.

At the time it was made public (mid 2005), the analysis presented here was the first available work on affect behavior in blogs over time.

6.3.2 Capturing Global Moods from Text

Formally, given a set of blog posts and a temporal interval, our task is to determine the prevalence of each one of a list of given moods in the posts; evaluation is done by comparing the relative mood levels obtained from the text with the relative mood levels as reflected in the mood indicators existing in some of the posts. We approach this as a multivariate regression problem, where the response variables are the mood levels and the predictors need to be derived from the text of the posts. The task is to find the relation between these predictors and the mood intensities; we follow with details on extracting the predictors and building the regression models.

Mood predictors. Our first goal is to discover textual features that are likely to be useful in estimating prevalence of moods in a given time slot. In Section 4.1 we introduced a wide range of text-based features for classifying moods in blogs, including word and POS n-gram frequencies, special characters, PMI values, and so on. As we are now dealing with a much higher volume of text—thousands of blog posts are written every minute—we limit ourselves to the basic features,

- The hour of the day from which the data in this instance came (between 0 and 23).
- A binary indication of whether the day of the week to which this instance relates is a weekend day (i.e., Saturday or Sunday).
- The total amount of blog entries posted in this hour.
- For each discriminating term, its frequency: the percentage of blog posts containing it.

Figure 6.6: Predictors of mood intensities used for the regression analysis.

which are computationally inexpensive and easy to calculate even for large volumes of text: frequencies of word n-grams.

Rather than using the frequencies of all n-grams as predictors, we use a limited list of n-grams: those that are most likely to be associated with moods. To select which n-grams appear on this limited list, we employ the same "indicative term" extraction methods we have already used on several occasions in Chapters 3 and 4. For our current task, we are interested in terms which are indicative of multiple moods. For this, we first create lists of indicative terms for each mood we intend to predict, then merge the top terms from each of these lists.

Since our prediction is for mood intensities at a given time point, we add the time of day, as well as an indication of whether the time slot occurs on a weekend to the set of predictors. The final set of predictors is summarized in Figure 6.6.

Modeling mood levels. Once the set of predictors for mood detection is identified, we need to learn models that predict the intensity of moods in a given time slot from these predictors. Training instances are constructed from the mood-annotated data for each mood m; every training instance includes the attributes listed in Figure 6.6, as well as the "intensity" of mood m, the actual count of blog posts reported with that mood at the given time point hour.

We experimented with a number of learning methods, and decided to base our models on Pace regression [306], which combines good effectiveness with high efficiency. Pace regression is a form of linear regression analysis that has been shown to outperform other types of linear model-fitting methods, particularly when the number of features is large and some of them are mutually dependent, as is the case in our data. As with other forms of linear regression, the model we obtain for the level of mood m is a linear combination of the features, in the following format:

$$\begin{array}{rcll}
\text{MoodIntensity}_m & = & \alpha_1 \cdot \text{total-number-of-posts} & + \\
 & & \alpha_2 \cdot \text{hour-of-day} & + \\
 & & \alpha_3 \cdot \text{freq}(t_1) & + \\
 & & \alpha_4 \cdot \text{freq}(t_2) & + \\
 & & \ldots, &
\end{array}$$

where t_i are the discriminating terms, and the values of α_i are assigned by the regression process.

It is important to note that both stages of our method—identifying the discriminating terms and creating models for each mood—are performed offline, and only once. The resulting models are simple, computationally cheap, linear combinations of the features; these are very fast to apply on the fly, and enable fast online estimation of "current" mood levels in the blogspace.

6.3.3 Evaluation

We now describe the experiments we performed to answer our research question—whether global mood levels can be estimated from blog text with the proposed estimation method. First, we provide details about the corpus we use; we follow with details about the discriminating terms chosen and the regression process.

Corpus. Our data consists of all public LiveJournal blog posts published during a period of 39 days, from mid-June to early-July 2005. For each entry, we store the entire text of the post, along with the date and the time of the entry. If a mood was reported for a certain blog post, we also store this indication. As described in Section 4.1, the moods used by LiveJournal users are either selected from a predefined list of 132 moods, or entered in free-text.

The total number of blog posts in our collection is 8.1 million, containing over 2.2GB of text; of these, 3.5 million posts (43%) have an indication of the writer's mood.[5]

One issue to note regarding our corpus is that the timestamps appearing in it are server timestamps—the time in which the U.S.-located server received the blog post, rather than the local time of the blogger writing the entry. While this would appear to introduce a lot of noise into our corpus, the actual effect is mild since, as we mentioned earlier, the vast majority of LiveJournal users are located in North America, sharing or nearly-sharing the time-zone of the server.

Discriminating terms. We used the text of 7 days' worth of posts to create a list of discriminating terms as described in Section 3.2; this time, we are searching for terms indicative of a specific mood rather than a given blogger. For this, we

[5] As an aside, the percentage of LiveJournal posts annotated with moods has slowly, but constantly, been decreasing since the experiments reported here; in late 2006, it was about 35%.

need to compare text associated with a given mood with more general text; in our case, the general text is the text of all posts, regardless of mood.

More specifically, for each mood m of the most popular 40 moods we aggregate the text of all posts annotated with this mood, T_m, and compare the frequencies of all unigrams and bigrams in it with their frequencies in the text of the entire corpus, T, using the log-likelihood measure described in Section 3.2. Ranking the terms by these LL values, we obtain, for each mood m, a separate list of indicative terms of that mood. This is identical to the process described when creating the indicative features for mood classification at the single blog level described in Section 4.1; examples of top-ranked indicative terms for some moods are shown in Table 6.5.[6]

Mood	**Indicative unigrams**	**Indicative bigrams**
hungry	hungry eat bread sauce	am hungry hungry and some food to eat
frustrated	n't frustrated frustrating do	am done can not problem is to fix
loved	love me valentine her	I love love you love is valentines day

Table 6.5: Most discriminating word n-grams for some moods.

Next, we combine the separate lists to a single list of indicative terms. As the amount of data we deal with is large, we aim to produce a relatively short list; the large amount of data requires a relatively simple set of features for fast analysis. For this reason, we focus only on the top-10 terms from the indicative list of each mood m; after manually filtering it to remove errors originating from technical issues (mostly tokenization problems—in total less than 10 terms were removed from all lists combined), we merge all top terms to a single list of terms indicative of moods in blogs. In total, this list contained 199 terms, of which 167 are single words and the rest word bigrams. Some examples of the discriminating terms in this list are shown in Table 6.6, along with the moods including them on their top-10 indicative term list.

Instances. The posts included in the 7 days that were used to identify the discriminating terms were removed from the corpus and not used for subsequent parts of the experiments. This left us with 32 days' worth of data for generating

[6] This is a repetition of the information in Table 4.2, and appears here for convenience.

Term	Source moods
love	cheerful, loved
envy	busy, sad
giggle	contemplative, good, happy
went to	contemplative, thoughtful
work	busy, exhausted, frustrated, sleepy, tired

Table 6.6: Examples of discriminating terms in our feature set.

the models and testing them. Instances were created by collecting, for every hour of those 32 days, all posts time-stamped with that hour, yielding a total of 768 instances. The average length of a single post in this collection is 140 words, or 900 bytes; the distribution of posts during a 24-hour period is given in Figure 6.7; each single-hour instance is therefore based on 2,500–5,500 individual posts, and represents 350K–800K words.

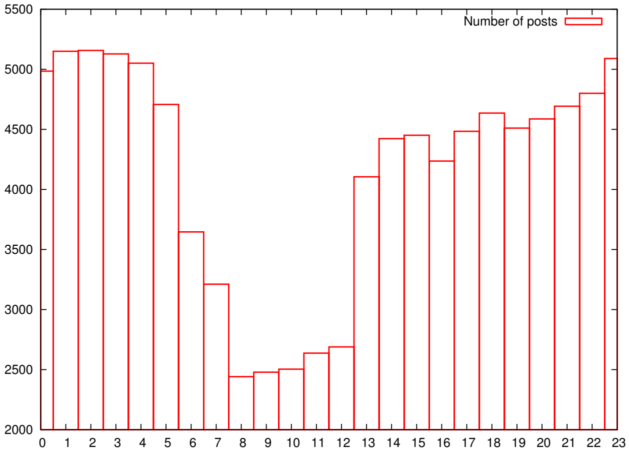


Figure 6.7: Average number of posts throughout the day. X-axis shows the hour of the day (GMT).

Generated models. We used the Pace regression module from the WEKA toolkit [314] to create our models. Since the models we create are linear regression models, they strongly exhibit the importance of features as positive and negative indicators of moods. Table 6.7 shows examples of the regression results for a couple of moods.[7]

[7] Pace regression includes a form of feature selection, therefore not all features are actually used in the resulting models.

Mood		Linear Model		
depressed	=	0.0123 ·	total-number-of-posts	+
		-523.777 ·	freq(“accomplished”)	+
		-367.5239 ·	freq(“confront”)	+
		-88.5883 ·	freq(“crazy”)	+
		-52.6425 ·	freq(“day”)	+
		90.5834 ·	freq(“depressed”)	+
		154.3276 ·	freq(“die”)	+
		-50.9185 ·	freq(“keep”)	+
		-147.1118 ·	freq(“lol”)	+
		-1137.6272 ·	freq(“old times”)	+
		283.2972 ·	freq(“really sick”)	+
		-235.6833 ·	freq(“smoke”)	+
		59.3897 ·	freq(“today”)	+
		195.8757 ·	freq(“tomorrow”)	+
		552.1754 ·	freq(“violence”)	+
		81.6886 ·	freq(“went”)	+
		-118.8249 ·	freq(“will be”)	+
		191.9001 ·	freq(“wish”)	+
		-19.23		
sick	=	-0.046 ·	hour-of-day	+
		0.0083 ·	total-number-of-posts	+
		20.3166 ·	freq(“cold”)	+
		-287.3355 ·	freq(“drained”)	+
		-91.2445 ·	freq(“miss”)	+
		-196.2554 ·	freq(“moon”)	+
		-67.7532 ·	freq(“people”)	+
		357.523 ·	freq(“sick”)	+
		615.3626 ·	freq(“throat”)	+
		60.9896 ·	freq(“yesterday”)	+
		1.6673		

Table 6.7: Examples of mood level models.

Experiments. All 768 instances of data were used to perform a 10-fold cross-validation run. The performance measures we use for our estimation are *correlation coefficient* and *relative error*. The *correlation coefficient* is a standard measure of the degree to which two variables are linearly related, and is defined as

$$\text{CorrCoefficient} = \frac{S_{PA}}{S_P \cdot S_A},$$

where

$$S_{PA} = \frac{\Sigma_i (p_i - \overline{p}) \cdot (a_i - \overline{a})}{n-1}$$

$$S_P = \frac{\Sigma_i (p_i - \overline{p})^2}{n-1}, \quad S_A = \frac{\Sigma_i (a_i - \overline{a})^2}{n-1},$$

and p_i is the estimated value for instance i, a_i is the actual value for instance i, $\overline{x}$ is the average of x, and n is the total number of instances. The *relative error* denotes the mean difference between the actual values and the estimated ones, and is defined as:

$$\text{RelError} = \frac{\Sigma_i (|p_i - a_i|)}{\Sigma_i (|a_i - \overline{a}|)}.$$

The correlation coefficient indicates how accurate the mood estimation is *over time*, showing to what degree the fluctuation patterns of a mood are predicted by the model. This is our primary metric, since we view estimation of the mood's behavior over time (e.g., detection of peaks and drops) as more important than the average accuracy as measured at each isolated point in time (which is given by the relative error). A correlation coefficient of 1 means that there is a perfect linear relation between the prediction and the actual values, whereas a correlation coefficient of 0 means that the prediction is completely unrelated to the actual values.[8]

As a baseline, we perform regression on the non-word features only, i.e., the hour of the day, the total amount of posts in that hour, and whether the day is a weekend day or not. As we demonstrated in the previous section, many moods display a circadian rhythm; because of this, and the strong dependence on the total amount of moods posted in a time slot, the baseline already gives a fairly good correlation for many moods (but the error rates are still high).

Table 6.8 shows the results of our experiments for the 40 most frequent moods. The relative high relative error rates reiterate our findings from Chapter 4 regarding the difficulty of the mood classification task itself; isolated from the temporal context, the accuracy at each point of time is not high. However, the high correlation coefficients show that the temporal behavior—the fluctuation over time—is

[8] More generally, the square of the correlation coefficient is the fraction of the variance of the actual values that can be explained by the variance of the prediction values; so, a correlation of 0.8 means that 64% of the mood level variance can be explained by a combination of the linear relationship between the prediction, and the acutal values and the variance of the prediction itself.

well predicted by the model. Also shown in Table 6.8 are the improvements of the regression over the baseline: in almost all cases the correlation coefficient increased and the relative error decreased, with substantial improvements in many cases. Note that the range of the changes is quite broad, both for the correlation coefficient and for the relative error. The average and median increase in correlation coefficient are 19% and 5.3%, respectively, and the average and median decrease in relative error are 18% and 9.7%, respectively.

The correlation levels achieved are fairly high: to demonstrate this, Figure 6.8 shows the estimated and actual levels of the mood *good* over a period of 6 days, with a correlation of 0.84—slightly higher than the average correlation achieved for all moods, 0.83.[9]

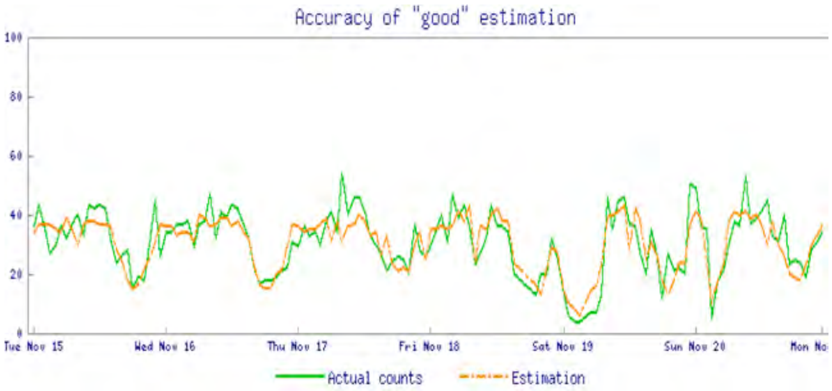


Figure 6.8: Example of mood prediction over time: prevalence of *good* (green, solid line) and the estimation based on the proposed method (orange, dashed line); the correlation in this case is 0.84.

What causes the difference in performance of our estimator across different moods? One hypothesis could be that moods for which our estimator scores higher (e.g., "bored," "happy") tend to be expressed with a small number of fairly specific words, whereas moods on which our estimator scores lower (e.g., "cold," "touched") are associated with a far broader vocabulary.

6.3.4 Case Studies

We now present two particular test cases, exhibiting particular mood prediction patterns. For these test cases, we divided our 32-day corpus into two parts: just over 24 days (585 hours) during June 2005, and just over 7 days (183 hours)

[9] This graph was created using MoodViews, a tool based on the method presented in this section and described in more detail in Appendix B.

	Correlation Coefficient			**Relative Error**		
Mood	**Baseline**	**Regression**	**Change**	**Baseline**	**Regression**	**Change**
drunk	0.4070	0.8611	+111.57%	88.39%	53.20%	−39.81%
tired	0.4882	0.9209	+88.63%	88.41%	37.09%	−58.04%
sleepy	0.5157	0.9106	+76.57%	80.46%	39.46%	−50.94%
busy	0.5346	0.8769	+64.02%	82.46%	45.15%	−45.24%
hungry	0.5601	0.8722	+55.72%	78.56%	44.06%	−43.91%
angry	0.5302	0.7944	+49.83%	73.70%	70.13%	−4.84%
exhausted	0.6212	0.9132	+47.00%	77.68%	39.32%	−49.38%
scared	0.4457	0.6517	+46.21%	80.30%	84.07%	+4.70%
distressed	0.5070	0.6943	+36.94%	77.49%	76.95%	−0.69%
sad	0.7243	0.8738	+20.64%	55.53%	49.91%	−10.12%
excited	0.7741	0.9264	+19.67%	61.78%	36.68%	−40.62%
horny	0.6460	0.7585	+17.41%	75.63%	63.44%	−16.11%
bored	0.8256	0.9554	+15.72%	54.22%	26.08%	−51.89%
drained	0.7515	0.8693	+15.67%	65.51%	49.50%	−24.44%
cold	0.5284	0.5969	+12.96%	87.02%	82.94%	−4.69%
depressed	0.8163	0.9138	+11.94%	57.45%	39.47%	−31.28%
anxious	0.7736	0.8576	+10.85%	60.02%	49.67%	−17.23%
loved	0.8126	0.8906	+9.59%	57.86%	44.88%	−22.43%
cheerful	0.8447	0.9178	+8.65%	50.93%	37.67%	−26.04%
chipper	0.8720	0.9212	+5.64%	47.05%	37.47%	−20.36%
bouncy	0.8476	0.8924	+5.28%	50.94%	41.31%	−18.9%
satisfied	0.6621	0.6968	+5.24%	72.97%	70.42%	−3.50%
sick	0.7564	0.7891	+4.32%	64.00%	60.15%	−6.01%
thankful	0.6021	0.6264	+4.03%	78.07%	77.48%	−0.75%
okay	0.8216	0.8534	+3.87%	54.52%	50.23%	−7.86%
ecstatic	0.8388	0.8707	+3.80%	52.35%	47.27%	−9.71%
amused	0.8916	0.9222	+3.43%	43.55%	37.53%	−13.8%
aggravated	0.8232	0.8504	+3.30%	54.91%	50.32%	−8.36%
touched	0.4670	0.4817	+3.14%	86.11%	85.39%	−0.83%
annoyed	0.8408	0.8671	+3.12%	52.28%	48.30%	−7.61%
thoughtful	0.7037	0.7251	+3.04%	69.38%	67.83%	−2.23%
crazy	0.8708	0.8932	+2.57%	46.87%	42.84%	−8.58%
cranky	0.7689	0.7879	+2.47%	63.01%	60.89%	−3.36%
happy	0.9293	0.9519	+2.43%	34.72%	28.86%	−16.86%
calm	0.8986	0.9146	+1.78%	41.89%	38.20%	−8.81%
curious	0.7978	0.8110	+1.65%	57.30%	55.69%	−2.82%
hopeful	0.8014	0.8139	+1.55%	58.79%	57.40%	−2.37%
good	0.8584	0.8714	+1.51%	51.30%	48.86%	−4.75%
optimistic	0.5945	0.6024	+1.32%	80.60%	80.25%	−0.44%
confused	0.8913	0.9012	+1.11%	44.96%	42.99%	−4.37%
average	0.7231	0.8320	+18.92%	63.56%	51.67%	−17.69%

Table 6.8: Mood level estimation for the 40 most frequent moods: 10-fold cross-validation over data from 32 days.

during July 2005.[10] The 24-day period was used for creating models, and the 7-day period for the actual case studies.

Terror in London. On the 7th of July 2005, a large-scale terror attack took place in London, killing dozens and wounding hundreds; this attack was strongly reflected in the mass media during that day, and was also a primary topic of discussion for bloggers. Following the attack, the percentage of bloggers reporting moods such as "sadness" and "shock" climbed steeply; other moods, such as "amused" and "busy," were reported with significantly lower levels than their average. An example of the reaction of bloggers can be seen in Figure 6.4 in the previous section.

Our method failed to predict both of these phenomena: the rise of negative moods and the fall of positive ones. Figure 6.9 shows two examples of the failure, for the moods "sadness" and for "busy." The correlation factors for some moods, such as these two, drop steeply for this period.

An examination of the blog posts reported as "sad" during this day shows that the language used was fairly unique to the circumstances: recurring words were "terror," "bomb," "London," "Al-Qaeda," and so on. Since these words were not part of the training data, they were not extracted as indicative features for sadness or shock, and were not included in our estimation method.

We hypothesized that given the "right" indicative words, our method would be able to estimate also these abnormal mood patterns. To test our hypothesis, we modified our data as follows:

- Add the two words "attack," and "bomb" to the list of words used as discriminating terms. These were the top overused terms during this time period, according to the log-likelihood measure we use to identify indicative words.

- Move two instances from the test data to the training data; these two instances reflect two hours from the period of "irregular mood behavior" on July 7th (the hours selected were not the peak of the spikes).

This emulates a scenario where the language used for certain moods during the London attacks has been used before in a similar context; this is a likely scenario if the training data is more comprehensive and includes mood patterns of a larger time span, with more events.[11]

[10] These consist of July 1st to July 3rd, and July 6th to July 9th. We have no data for two days—July 4th and 5th—due to technical issues.

[11] In the particular case where there is a stream of data updated constantly, some of it annotated—as is the case with blog posts—this can be done automatically: the quality of the estimation is measured with new incoming annotated data, and when the quality drops according to some critera, the models are retrained.

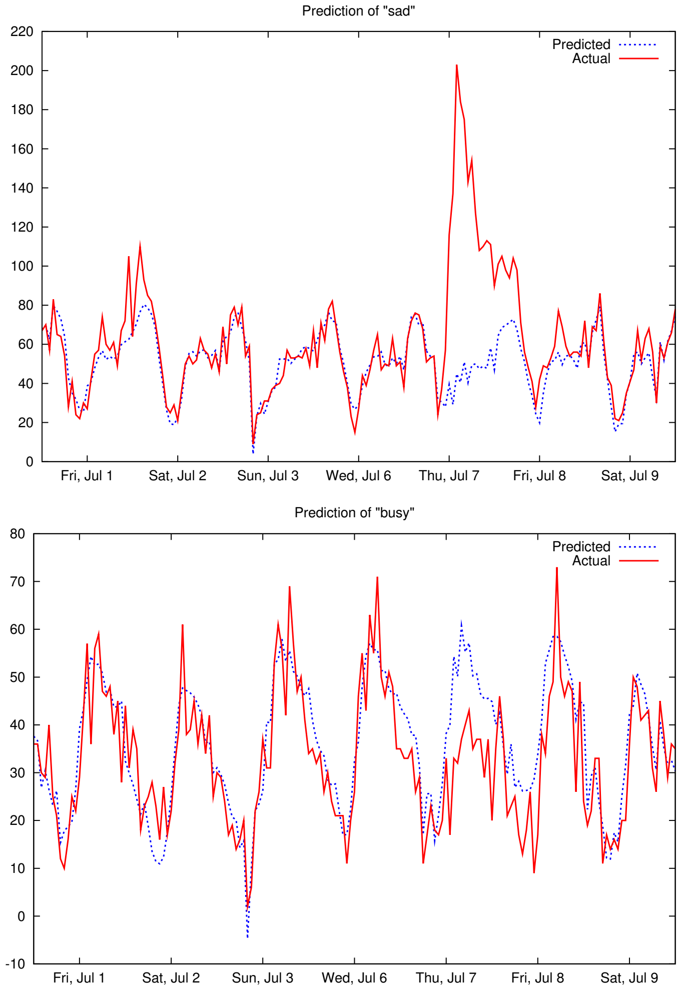


Figure 6.9: Failure to predict a sadness spike following the terror attacks in London (top), and the accompanying decrease in busyness (bottom). Counts of posts are indicated on the Y-axis; the red, continuous line marks actual counts, and the blue, dashed line is the prediction.

We then repeated the estimation process with the changed data; the results for "sadness" are shown in Figure 6.10. Accordingly, the correlation values climb back close to those achieved in our 10-fold cross-validation.

Weekend drinking habits. Our next test case is less somber, and deals with the increased rate of certain moods over weekends, compared to weekdays—already mentioned when discussing cyclic mood patterns in the previous section.

Figure 6.11 shows our estimation graphs for the moods "drunk" and "excited" for the same period as the previous London bombing test case—a period including

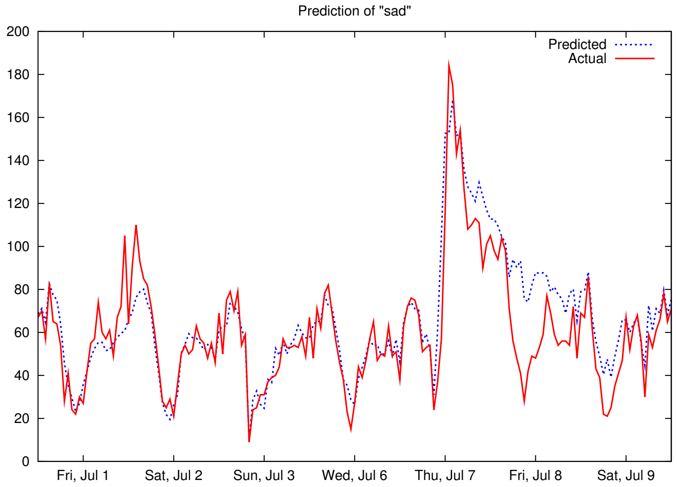


Figure 6.10: Successful prediction of the sadness peak with modified data. Counts of posts are indicated on the Y-axis.

two weekends. Clearly, both moods are successfully predicted as elevated during weekends, although not at the full intensity.

6.3.5 Conclusions

The work we presented aims at identifying the intensity of moods in the blogspace during given time intervals. Using a large body of blog posts manually annotated with their associated mood, we achieve high correlation levels between predicted and actual moods by using words which are indicative of certain moods. Our main finding is that while prediction of mood at the individual blog post level is a hard task, as shown in Section 4.1, at the aggregate level, predicting the *intensity* of moods over a time span can be done with a high degree of accurracy, even without extensive feature engineering or model tuning. Having said that, we believe that further expansions of the predictor set, i.e., using a larger amount of discriminating terms, and using any of the features used for single-post mood classification, will improve the results further.

6.4 Explaining Irregular Mood Patterns

The previous section included some examples of irregular mood behavior in the blogspace, such as peaks of certain moods after large-scale global events. In many of these cases, the reason for the irregular behavior is clear to the observer, assuming she shares the same cultural context as the bloggers. If we know that there has been a major tragedy we expect people to be shocked or sad; we assume

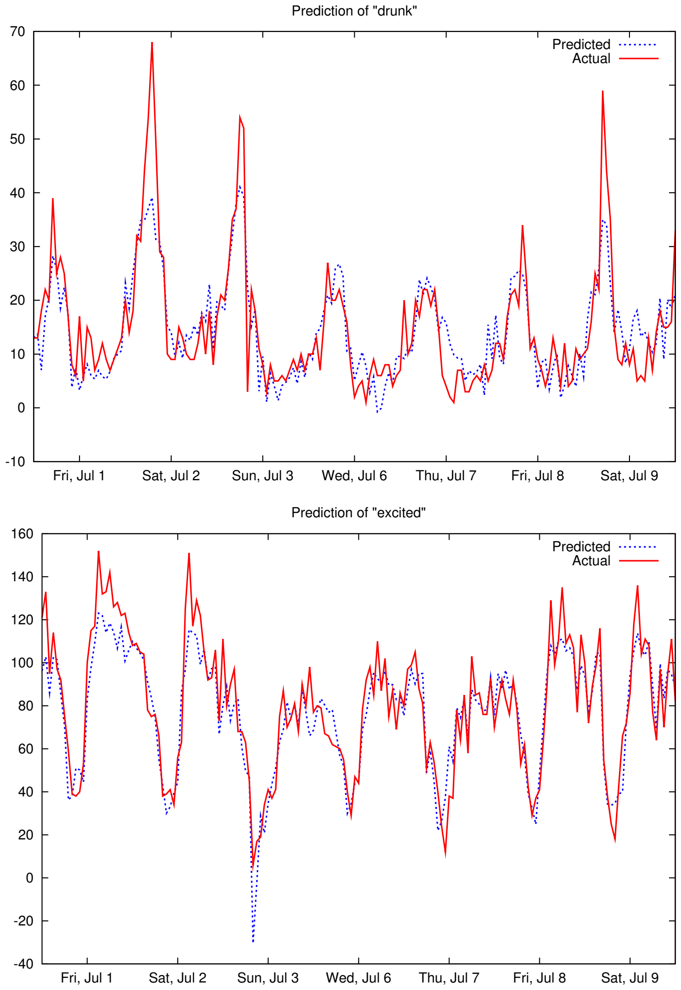


Figure 6.11: Prediction of weekend-related moods: “drunk” (top) and “excited” (bottom). Counts of posts are indicated on the Y-axis.

elevated relaxation during holidays, and so on.

However, not all irregularities in mood fluctuations are easily explained, and some require very specific context to understand. Consider, for example, the spike in excitement experienced in the blogspace in mid-July 2005 (Figure 6.12): what has happened on this day to make bloggers react so strongly? When we first encountered this peak, we were not aware of any large-scale event with such an expected effect.

In this section, we develop a method to address this and similar questions. Our approach identifies unusual changes in mood levels in blogs and locates an explanation for the underlying reasons for these changes: a natural-language text describing the event that caused the unusual mood change.

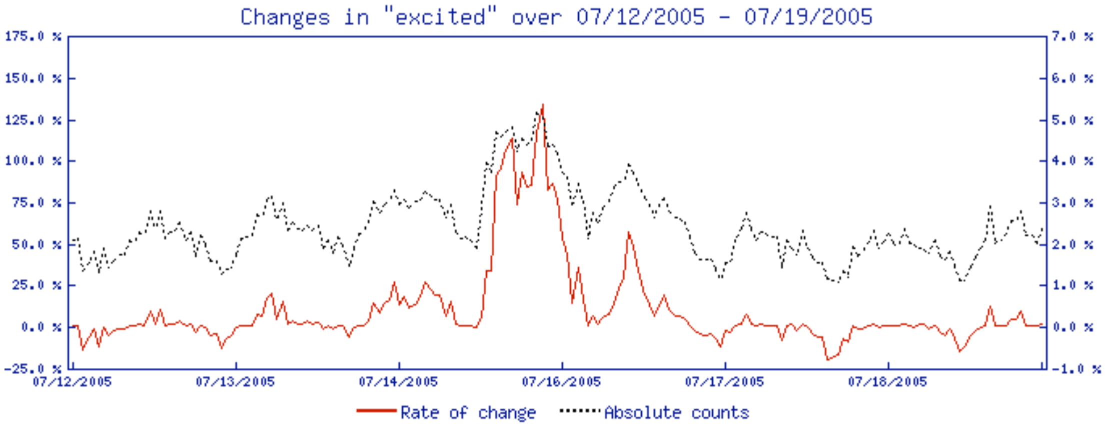


Figure 6.12: Surge in excitement in blogs on July 16th, 2005.

The method we use to produce such explanations is as follows. First, we compare the expected mood levels and the actual ones to identify irregular behavior. If unusual spikes occur in the level of mood m, we examine the language used in blog posts labeled with m around and during the period in which the spike occurs. We compare this language to the long-term language model for m, using overused terms for the irregular period as indications for the mood change. Once these terms are identified, we use them to consult a collection of global events—a news corpus—from which we retrieve a small text snippet as the desired explanation.

Our work is related to the burstiness models described by Kleinberg in time-lined corpora such as email and research papers [147]; there, irregularities are identified by applying probability models used to analyze communication networks. The same model is applied to discover dense periods of "bursty" intra-community link creation in the blogspace [156] and to identify topics in blogs over time [217].

6.4.1 Detecting Irregularities

Our first task, then, is to identify spikes in moods reported in blog posts: unusually elevated or degraded levels of a mood in a particular time period. As shown earlier, many moods display a cyclic behavior pattern, maintaining similar levels at a similar time-of-day or day-of-week (see Section 6.2). Our approach to detecting spikes addresses this by comparing the level of a mood at a given time point with the "expected" level—the level maintained by this mood during other, similar time points. Formally, let POSTS($mood, date, hour$) be the number of posts labelled with a given mood and created within a one-hour interval at a specified date. Similarly, ALLPOSTS($date, hour$) is the number of all posts created within the interval specified by the date and hour. The ratio of posts labeled with a given mood to all posts for a day of a week (Sunday, . . . , Saturday) and for a one-hour intervals (0, ..., 23) is given by:

$$R(mood, day, hour) = \frac{\sum_{\text{DW}(date)=day} \text{POSTS}(mood, date, hour)}{\sum_{\text{DW}(date)=day} \text{ALLPOSTS}(date, hour)}$$

where $day = 0, \ldots, 6$ and DW($date$) is a day-of-the-week function that returns 0, ..., 6 depending on the date argument.

The level of a given mood is *changed* within a one-hour interval of a day, if the ratio of posts labelled with that mood to all posts, created within the interval, is significantly different from the ratio that has been observed on the same hour of the similar day of the week. Formally:

$$D(mood, date, hour) = \frac{\frac{\text{POSTS}(mood,date,hour)}{\text{ALLPOSTS}(date,hour)}}{R(mood, \text{DW}(date), hour)}$$

If $|D|$ exceeds a threshold we conclude that an unusual spike occurred, while the sign of D makes it possible to distinguish between positive and negative spikes. The absolute value of D expresses the degree of the peak. Consecutive hours for which $|D|$ exceeds the thresholds are grouped into a single interval, where the first hour marks the start of the peak and the last one is the end of it.

6.4.2 Explaining Irregularities

Once an irregular interval is identified, we proceed to the next stage: providing an explanation to the irregular behavior. First, we identify terms which are indicative of the irregularity. For this, we follow the same language-model-based keyword extraction approach used to identify terms associated with a particular mood in the previous Section; however, we now attempt to identify indicative terms for a given mood *during a given period*, rather than terms indicative of it regardless of time. To this end, we compare the language model associated with the mood m during the irregular period with the language model associated with m in other periods, generating a ranked list of indicative terms.

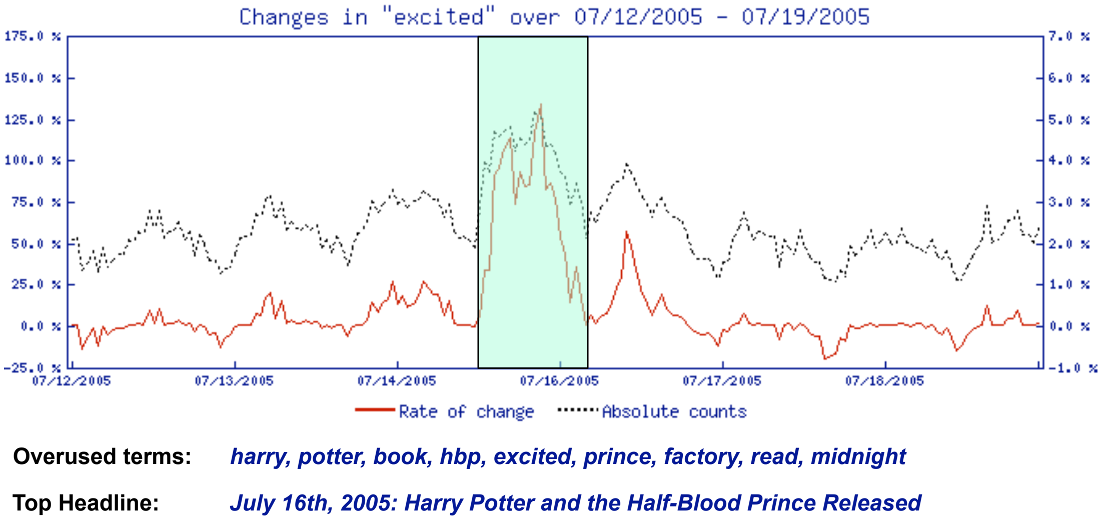
After identifying a set of terms related to the event behind the unusual pattern, the next step is straightforward: the top indicative terms are matched with a set of descriptions of events that took place during a time corresponding to the irregularity. Such timestamped descriptions are easy to obtain, e.g., through newswire corpora or streamed headlines.

Due to the length and high quality of the "queries" used—each query term is typically highly-related to the event—the effectiveness of this retrieval process is high, particularly for early precision. Different, unrelated events taking place within the same short time period share little vocabulary: we found that a few query terms, and a simple ranking mechanism (measuring the overlap between the top overused terms and the title of the event) provide good results.

6.4.3 Case Studies

Evaluation of the method described here is non-trivial. Instead, we show a few test cases demonstrating its usefulness. In these examples, the corpus used is identical to the one from the previous section—public blog posts of LiveJournal, starting from July 2005. As a news corpus, we used the English edition of Wikinews (`http://en.wikinews.org`), a collaborative site offering syndicated, royalty-free news articles.

For the first example, we return to the unusual peak of excitement appearing in Figure 6.12—the one which was unclear to us and prompted the development of the method this Section describes. Figure 6.13 shows this peak again; this time, the interval identified as unusual by our method is highlighted. The top overused terms during this period were "harry," "potter," "book," "hbp," "excited," and "prince." The headline of the top Wikinews article retrieved for the date of the peak using these terms is "July 16: Harry Potter and the Half-Blood Prince released" (recall that the average blogger is in her teens—an age group where this particular book series is extremely popular). Clearly, this is a simple, short, and effective explanation for excitement among bloggers .



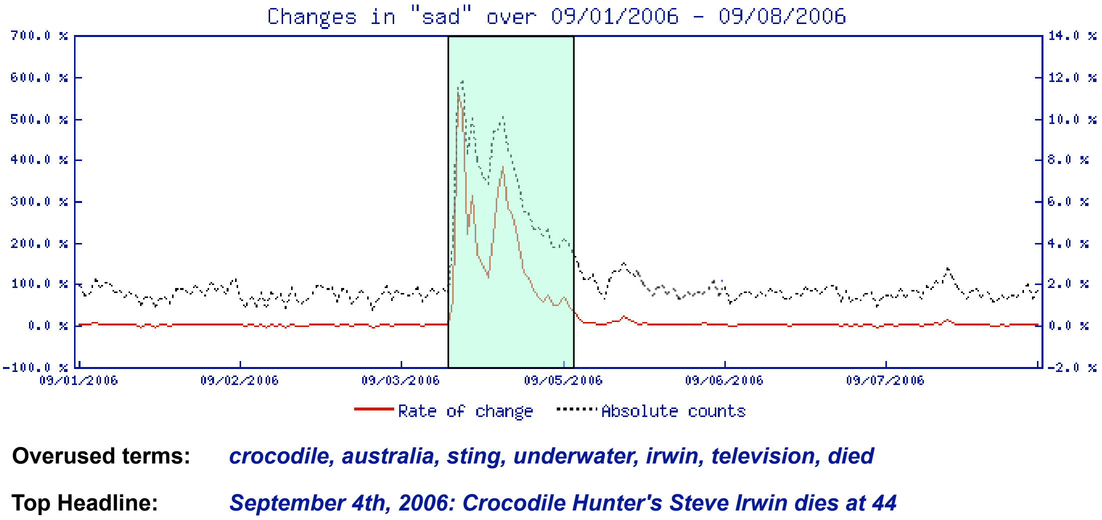
Overused terms: *harry, potter, book, hbp, excited, prince, factory, read, midnight*

Top Headline: *July 16th, 2005: Harry Potter and the Half-Blood Prince Released*

Figure 6.13: Surge in excitement in blogs on July 16th, 2005: surge interval identified, overused terms extracted, and top headline retrieved.

The next example concerns another event with a much stronger influence on bloggers than anticipated. During September 4th, 2006 (and, to a lesser extent, September 5th), significant peaks of sadness, shock and sympathy were registered, again with no obvious explanation to us as observers. The top overused terms for these moods during this time were "crocodile," "australia," "sting," "underwater," and "irwin"; the top retrieved Wikinews article for these terms described the death of Steve Irwin, the star of a television show called "The Crocodile

Hunter," in an underwater accident off the coast of Australia. Figure 6.14 shows the results of our method for this case: the highlighted peak, the overused terms extracted and the top headline; again, the resulting explanation clearly provides context for the irregularity.



Overused terms: *crocodile, australia, sting, underwater, irwin, television, died*

Top Headline: *September 4th, 2006: Crocodile Hunter's Steve Irwin dies at 44*

Figure 6.14: Surge in sadness in blogs on September 4th, 2006: surge interval identified, overused terms extracted, and top headline retrieved.

In addition to these two examples, the method we proposed was used to identify causes for other unusual mood phenomena including those shown on Figure 6.4, caused by major news events. More details about a web service built around this method are provided in Appendix B.

6.4.4 Conclusions

We described a method for relating changes in the global mood, as reflected in the moods of bloggers, to large-scale events. The method is based on identifying changes in the vocabulary bloggers use over time, and, particularly, identifying overused terms during periods in which the reported moods by bloggers differ substantially from expected patterns. Overused terms are then used as queries to a collection of time-stamped events, resulting in "annotations" of irregularities in global moods with human-readable explanations. While rigorous evaluation of such a task is complex, anecdotal evidence suggests that the resulting explanations are useful, and answer the question we set out to explore earlier in this Section.

More generally, the process of selecting overused terms during a specific time period in a timelined corpus (and, possibly, matching them with a corpus of events) can be used in other, non-mood-related scenarios. For example, it can be used to annotate peaks in occurrences of terms in a corpus over time,[12] or to track the changes in interests and opinions of a blogger towards a given topic by following the changing language models associated with this topic in her blog.

To summarize, this Chapter focused on aspects of aggregate sentiment analysis in blogs. First, we motivated this work by demonstrating its usefulness: we show that for analysis of Consumer Generated Media, sentiment analysis provides better prediction of financial success of products than the volume of discussion only, and discussed the relation between the accuracy of the prediction and the level of context used for the sentiment classification. We then demonstrated that the collective mood reports of bloggers provide insight into more than commercial data only, namely, into global behavior patterns. This latter observation has led to two research questions, explored in the subsequent parts of the Chapter. The first was whether the global mood can be inferred from the aggregate text of bloggers. To answer this, we coupled a regression-based approach with indicative term mining, showing that the answer is positive: global moods can be approximated with high degrees of accuracy using simple methods. This demonstrated the power of using large amounts of data for this task: as shown in Chapter 4, a similar task at the level of single posts resulted in substantially less accurate predictions. The second question we asked was whether irregularities in global mood behavior can be explained in an automated manner by monitoring the language use in the blogspace. Here, too, the answer is positive; to provide such explanations, we offer a method linking temporal changes in discriminative terms used by bloggers reporting a certain mood to a corpus of global events.

Taking a step back, we started with a somewhat traditional task for sentiment analysis: finding the relation between consumer opinion and product success, showing that known sentiment analysis methods are more beneficial in this domain than in others. The work discussed in the rest of the Chapter involved more novel information needs and tasks, demonstrating the type of knowledge that can be found in the blogspace, as a unique collection of people's emotions. Global mood patterns are one type of such knowledge (and a particularly singular one); we will return to non-factual aspects of blog content in Chapter III, where we address the task of locating sentiment in a large collection of blogs.

[12] Google Trends (`www.google.com/trends`) is an application which appears to do this for occurrences of terms in Google's search log; it was introduced after the work presented here was made public.

Chapter 7

Blog Comments

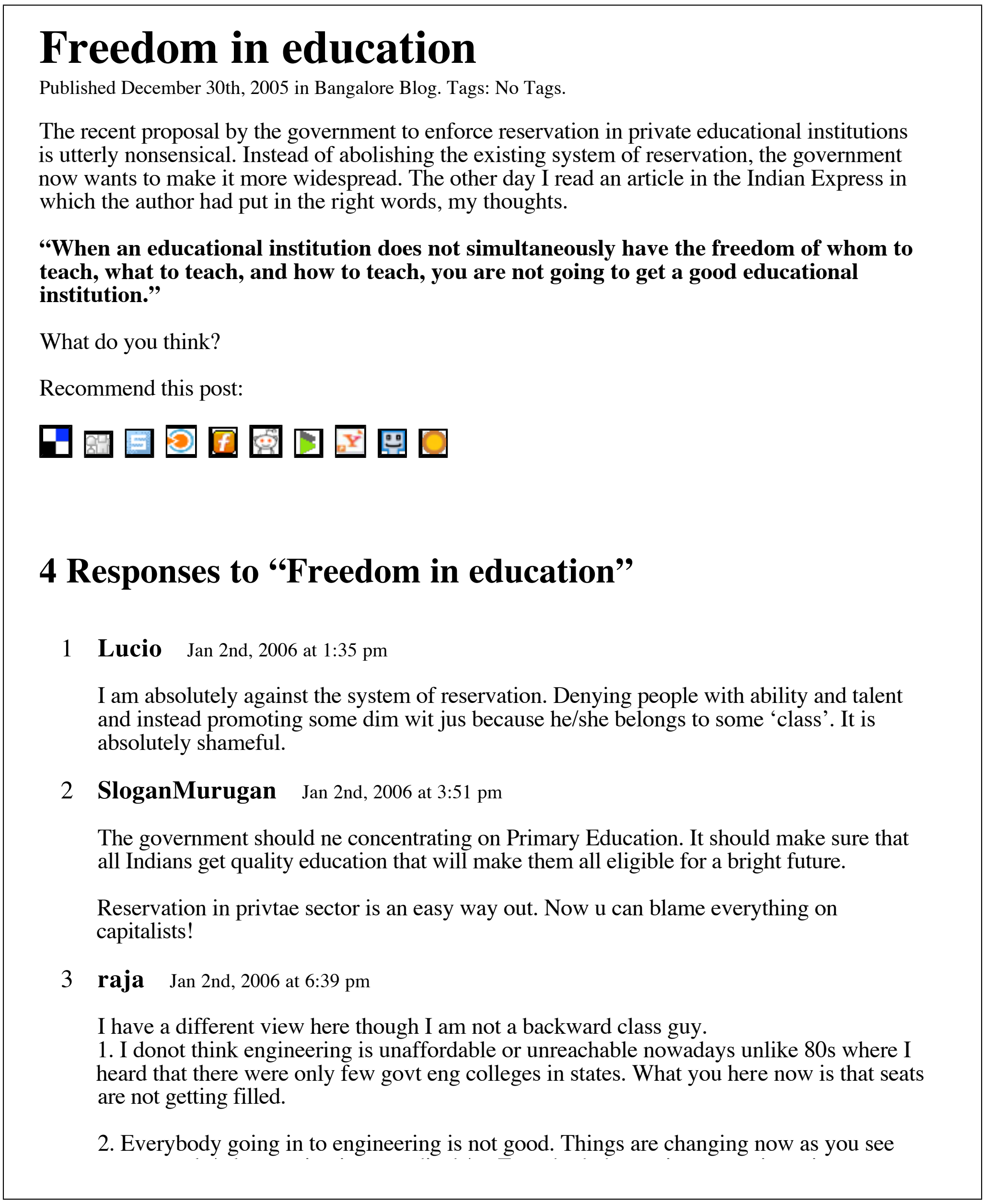
In the previous Chapter, we witnessed the power of extracting information from a large collection of blogs, where observations can be made about a large population, rather than about individuals. But while we aggregated information from multiple blogs, we did not look at the interactions between the individuals behind the blogs.

One manifestation of such interaction are comments posted to blogs. The commenting mechanism in blogs serves as "a simple and effective way for bloggers to interact with their readership" [184]; comments are considered one of the defining set of blog characteristics [313], and most bloggers identify them as an important motivation for their writing [294, 100]. Demonstrated in Figure 7.1, comments turn one-way publishing into a discussion between the blogger and readers of the blog.

Despite the possible usefulness of analyzing comments to mine information from blogs, comments are largely ignored in current large-scale studies of blog data, mostly because extracting and processing their content is somewhat more complex than extracting the content of the posts themselves. We have already discussed one particular aspect of blog comments, comment spam, in Chapter 5. In this Chapter, we present a large-scale study of comments in the blogspace, seeking basic knowledge of this domain—as well as identifying comment features which can be used in text analysis settings. At the time it was made public (early 2006), this study was the first to address blog comments at this scale.

The research questions we address in this work are these:

1. What is the volume of blog comments? How does it compare to the volume of blog posts?
2. To what extent does usage of comments improve access to blogs in typical tasks, such as blog search?
3. What relation exists between the amount of comments on a particular blog or post and its popularity, as measured by traditional influence metrics?
4. What knowledge can be mined from comments and the discussions taking place within them?



Freedom in education

Published December 30th, 2005 in Bangalore Blog. Tags: No Tags.

The recent proposal by the government to enforce reservation in private educational institutions is utterly nonsensical. Instead of abolishing the existing system of reservation, the government now wants to make it more widespread. The other day I read an article in the Indian Express in which the author had put in the right words, my thoughts.

"When an educational institution does not simultaneously have the freedom of whom to teach, what to teach, and how to teach, you are not going to get a good educational institution."

What do you think?

Recommend this post:

4 Responses to "Freedom in education"

1 **Lucio** Jan 2nd, 2006 at 1:35 pm

I am absolutely against the system of reservation. Denying people with ability and talent and instead promoting some dim wit jus because he/she belongs to some 'class'. It is absolutely shameful.

2 **SloganMurugan** Jan 2nd, 2006 at 3:51 pm

The government should ne concentrating on Primary Education. It should make sure that all Indians get quality education that will make them all eligible for a bright future.

Reservation in privtae sector is an easy way out. Now u can blame everything on capitalists!

3 **raja** Jan 2nd, 2006 at 6:39 pm

I have a different view here though I am not a backward class guy.
1. I donot think engineering is unaffordable or unreachable nowadays unlike 80s where I heard that there were only few govt eng colleges in states. What you here now is that seats are not getting filled.

2. Everybody going in to engineering is not good. Things are changing now as you see

Figure 7.1: Example of comments on a blog post. Source: http://www.vinayahs.com/archives/2005/12/30/freedom-in-education.

We proceed by describing related work in Section 7.1. Then, in Section 7.2, we describe the comment collection used in this study, and how it was created. In the rest of the Chapter we analyze the contribution of comments to text analysis of blogs: Section 7.3 examines the contribution of comment content to blog search, Section 7.4 looks at the relation between comments and blog popularity or authority, and Section 7.5 identifies a new type of knowledge that can be mined from comment information, namely, detection of disputes within the discussions taking place in them. Section 7.6 concludes this Chapter.

7.1 Related Work

As noted, quantitative studies of blogs focus on post data, leaving out the comments; an exception is the work of Herring et al. [112], studying the content and structure of a random sample of 203 blogs. In this sample, a relatively small amount of comments is found (average of 0.3 comments per post); however, the sample is too small to draw conclusions about the entire blogspace. Additionally, the content of the comments themselves, or the relations to their corresponding blog posts, are not further analyzed.

Qualitative studies of blogs, on the other hand, sometimes do refer to comments explicitly. Both Trevino [294] and Gumbrecht [100] study the importance of blog comments to the "blogging experience," and reach similar conclusions: comments are regarded by most bloggers as vital to the interactive nature of blogs.[1] Krishnamurthy studies the posting patterns to a specific blog following the September 11 events, finding that insightful posts attract the largest number of comments [154]. De Moor and Efimova [69] discuss blog comments in a larger context of blog conversations; among their findings is user frustration about the fragmentation of discussions between various blog posts and associated comments, indicating that, for users, the comments are an inherent part of the blog text, and they wish to access them as such.

In some cases, links found in comments are used to enrich the standard link-model of blogs: Chin and Chignell use them to identify communities, following existing work on link-based community extraction in the blogspace [53]. Nakajima et al. [216] use a combination of link and content analysis to analyze discussions taking place in blogs, aiming to locate prominent bloggers in these discussions. Similarly, Ali-Hasan and Adamic [7] use comments to identify two classes of bloggers in a community: "conversation starters," who create high-level content, and "conversation supporters," who generate discussion around it.

In terms of comment prevalence, a 2004 survey showed that 12% of internet users commented on a blog post at least once—amounting to about half of the number of bloggers [248]. A follow-up survey in 2006 found that 82% of bloggers posted a comment at least once, and that the vast majority of bloggers—87%—allow comments on their blog, with even higher percentages among young bloggers [165].

[1] Gumbrecht distinguishes between two types of bloggers—those who seek community interaction and appreciate comments, and those for whom the blog is a "protected space," where discussions are less important

7.2 Dataset

We now describe the comment corpus we studied and how it was built; as we will see, a test collection of comments is more complex to create than a blog post collection.

7.2.1 Comment Extraction

With the exception of a small number of blog software vendors and blogging hosts, comments are currently largely unsyndicated. We examined a random set of close to 1,000 blogs, and found less than 2% of them to contain comment content in syndicated form.[2] For this reason, we base our corpus on a comment mining mechanism, extracting comments from the blog HTML permalinks.

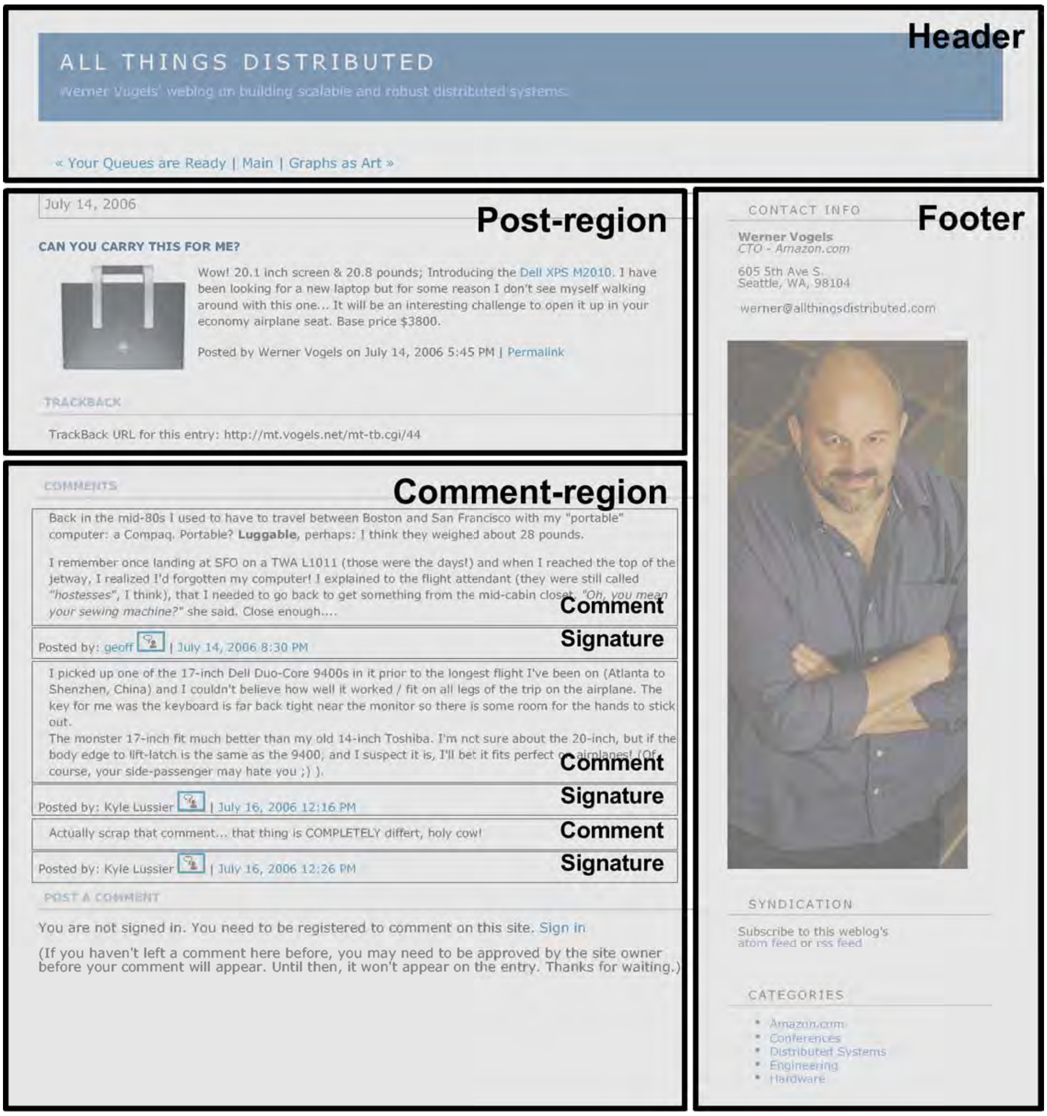
To extract comments from HTML content, we follow the model-based methodology used for extracting blog posts from HTML content used in [87, 217], adapting it to comments instead of posts.

Following this approach, we model the content of a permalink page as consisting of four parts: a header, a blog post region, a comment region, and a footer. The header and footer typically contain elements such as the blog title, blogger profile, archive links, and so on. The post region contains the post itself, its title, publication date, and, often, a signature. The comment region contains the comments; we model its content as `((signature comment)|(comment signature))*`, i.e., a list of comments separated by signatures, where signatures can appear at the beginning of a comment or at its end. The signature itself typically contains a date and the name of the comment author, sometimes with a link to her blog. An example of the regions in a permalink page is shown in Figure 7.2. Note that visually, the header and footer are not necessarily located at the top and the bottom of the page, but sometimes appear as sidebars; however, in the underlying HTML, they are almost always at the beginning and the end of the document.

The comment extraction task, then, boils down to two subtasks: identifying the "comment region" within a permalink page, and extracting the comments from it.

Comment region identification. To locate the comment region within an HTML page, we first convert the permalink content into an XHTML representation, then parse it into a DOM tree. As in [87], we utilize the relatively regular nature of different blog formats, and use a set of structural rules matched with the DOM tree to identify possible locations of comment content; rules include matching the end of the blog post itself, as extracted from the syndicated content,

[2] This was measured in late-2005; the number of blogs syndicating comments is expected to increase as blogging platforms develop and new standards allowing syndication of comments (e.g., RSS 2) are adopted.



ALL THINGS DISTRIBUTED

Werner Vogels' weblog on building scalable and robust distributed systems.

« Your Queues are Ready | Main | Graphs as Art »

July 14, 2006

CAN YOU CARRY THIS FOR ME?

Wow! 20.1 inch screen & 20.8 pounds; Introducing the Dell XPS M2010. I have been looking for a new laptop but for some reason I don't see myself walking around with this one... It will be an interesting challenge to open it up in your economy airplane seat. Base price $3800.

Posted by Werner Vogels on July 14, 2006 5:45 PM | Permalink

TRACKBACK

TrackBack URL for this entry: http://mt.vogels.net/mt-tb.cgi/44

COMMENTS

Back in the mid-80s I used to have to travel between Boston and San Francisco with my "portable" computer: a Compaq. Portable? **Luggable**, perhaps: I think they weighed about 28 pounds.

I remember once landing at SFO on a TWA L1011 (those were the days!) and when I reached the top of the jetway, I realized I'd forgotten my computer! I explained to the flight attendant (they were still called *"hostesses"*, I think), that I needed to go back to get something from the mid-cabin closet. *"Oh, you mean your sewing machine?"* she said. Close enough....

Posted by: geoff | July 14, 2006 8:30 PM

I picked up one of the 17-inch Dell Duo-Core 9400s in it prior to the longest flight I've been on (Atlanta to Shenzhen, China) and I couldn't believe how well it worked / fit on all legs of the trip on the airplane. The key for me was the keyboard is far back tight near the monitor so there is some room for the hands to stick out.
The monster 17-inch fit much better than my old 14-inch Toshiba. I'm not sure about the 20-inch, but if the body edge to lift-latch is the same as the 9400, and I suspect it is, I'll bet it fits perfect on airplanes! (Of course, your side-passenger may hate you ;)).

Posted by: Kyle Lussier | July 16, 2006 12:16 PM

Actually scrap that comment... that thing is COMPLETELY differt, holy cow!

Posted by: Kyle Lussier | July 16, 2006 12:26 PM

POST A COMMENT

You are not signed in. You need to be registered to comment on this site. Sign in

(If you haven't left a comment here before, you may need to be approved by the site owner before your comment will appear. Until then, it won't appear on the entry. Thanks for waiting.)

CONTACT INFO

Werner Vogels
CTO - Amazon.com

605 5th Ave S.
Seattle, WA, 98104

werner@allthingsdistributed.com

SYNDICATION

Subscribe to this weblog's
atom feed or rss feed

CATEGORIES

- Amazon.com
- Conferences
- Distributed Systems
- Engineering
- Hardware

Figure 7.2: Regions on a permalink page, as modeled for the comment extraction process. Source: http://www.allthingsdistributed.com/2006/07/can_you_carry_this_for_me.html.

as well as searching for known textual comment indicators ("Add a comment," "Post a reply"). While this approach is simple, we will see later that it correctly identifies the comment region in 95% of the permalinks.

Identifying comments within the region. Given the part of the HTML most likely to contain comments, identification of the comments within it is similar to identification of blog posts within a blog page: the task is to identify lists of dates, which serve as separators between comments, and use them to segment the content into separate entities. Here, again we follow [87], using first a rule-based date extractor, then identifying lists of dates with properties matching those of comment signatures: dates falling within a reasonable period after the blog post itself (we used 60 days), monotonous increasing or decreasing time, and regularity in date format.

One difference between comment extraction and blog post extraction is that comments are less regular in the relative location of the signatures—while posts usually contain first the title and date, then the content [87], in the case of comments the date is equally likely to appear before or after the comment itself. In practice, we found that good accuracy in determining the "parity" of the comment signatures—whether they appear before or after the content—is achieved via a simple method: our extractor examines the content between the first identified date and the start of the comment region. If this area contains non-formatting content—text viewable to a user—it is assumed to be a comment, and the dates are deduced to appear at the end of comments; otherwise, a date is assumed to appear before the comment.

Once the comment parity is known and a list of dates extracted, each date is expanded to a full comment by adding to it all the DOM elements following it or preceding it—according to the parity—until the next date in the list, or the boundary of the entire comment region, is reached.

Coverage. To test the coverage of our extraction module, we manually evaluated its output on a set of 500 randomly-selected permalinks of blog posts; in this set, 146 posts (29%) contained comments. Coverage was tested by comparing manually-extracted comments with the comments found by the wrapper, measuring the percentage of posts for which extraction was correct, as well as the percentage of posts with no comments which were correctly identified as such. The results of this evaluation are given in Table 7.1.

Set	**Correct**	**Incorrect**	**Total**
Posts with no comments	342 (97%)	12 (3%)	354
Posts with comments	95 (65%)	51 (35%)	146
All posts	437 (87%)	63 (13%)	500

Table 7.1: Comment extraction accuracy.

Sources of errors. In all cases of false positives—posts with no comments, but where our method identified some text as comment data—the blogs containing the posts were highly-irregular ones, with customized structure. In all these cases, our wrapper failed to correctly identify the comment region—resulting in interpretation of archive links, blogrolls, or the blog posts themselves as comments. This type of error is typical of model-based extraction mechanisms, which are tuned to the formats used by the majority of the data; however, their low level—3% in this case—prevents substantial skew in the resulting comment information.

As for false negatives—posts containing comments, that were not identified as such by our wrapper—here, our success rate is substantially lower: about a third of comment data is not fully extracted. Sources of these errors are multiple: in 17 out of the 51 cases (33%), the comment region was not identified—resulting in later failure to extract comment data. Of the rest of the false negatives, in 30 cases (59%) dates were not recognized—mostly due to non-English text, which is not handled by the date extractor we used. Since we use dates as markers of comments, failure to identify them leads to extraction failure later. The last 4 false negatives were due to assorted reasons, mostly technical (e.g., HTML parsing errors).

Note that for 11 out of the 51 comment extraction failures (21% of failures), the number of comments and their dates were correctly extracted, but the content was not. This means that for analyses which do not take content into account (such as determining the average number of comments per post), the wrapper's accuracy is over 70% on commented blogs, and 90% overall. Also, as we noted, in many cases—23 out of 51, or almost half—failures occurred on non-English permalinks; our coverage on commented English blogs only is close to 80%, and more than 90% overall.

7.2.2 A Comment Corpus

Using the extraction mechanism described in the previous Section, we collected a set of approximately 645,000 comments posted to blogs between July 11th and July 30th, 2005. The set was obtained using the following steps:

1. Collect all blog posts in the Blogpulse [89] index from the given period containing a permalink.
2. Remove "inactive" blogs; a blog was marked as inactive if, during the three months preceding the analyzed period, it contained less than 5 posts.
3. Fetch the HTML of the remaining permalinks, and run the extraction process on them.

In each of these steps, some content is missed: in the first stage, posts with no permalinks—about 8% of the total amount of posts—are ignored. The next stage filters a large amount of single-post-only blogs (which account for a significant percentage of total blogs [239]), as well as a lot of spam blogs. The main reason to include this stage is that we are interested in the interactions between "real" bloggers—not in blogs which are a one-time experiment, and certainly not in spam blogs (where comments, if any, are likely to be spam too). Overall, 65% of blogs were filtered in this stage. In the final stage, sources of missing content are multiple: broken links, hosts which restrict crawling the post HTML (e.g., LiveJournal), and the wrapper's incomplete coverage, as detailed in the previous Section.

Overall, we extracted comments from close to 10% of all blog posts published during the 20-day period studied. Based on estimating the amount of content missed in every stage, and since many of the blogs skipped are not likely to contain comments, we believe that this comment collection includes at least one quarter of all comments posted to blogs, in the entire blogspace, during this period: this is a sufficiently large sample to be able to make observations about the entire collection of comments in the blogspace. Table 7.2 contains some descriptive statistics about the collection.

Blog posts	685,976
Commented blog posts	101,769 (15%)
Blogs	36,044
Commented blogs	10,132 (28%)
Extracted comments	645,042
Mean comments per post	0.9
Mean number of days in which comments were posted, per post	2.1
Comments per post, excluding uncommented posts	
Mean	6.3
StdDev	20.5
Median	2
Comment Length (words)	
Mean	63
StdDev	93
Median	31
Total corpus size	
Words	40.6M
Text	225MB

Table 7.2: Corpus Statistics.

As expected, the number of comments per post follows a power-law distribution, with a small number of posts containing a high number of comments, and a long tail of posts with few comments; a plot of the number of blogs and posts having a given number of comments is shown on a log-log scale in Figure 7.3, with the best-fit power-law exponents—1.2 and 2.2, respectively. The distribution of the lengths of comments is similar—a small number of long comments, and many shorter ones.[3]

[3] Once again, as in other power-law observations such as those given in Section 2.2.2, we observe a distinct curve in the distribution, suggesting that a better fit is obtained with a log-normal distribution; however, the bottom line is similar: very few of one type of comments, and a long tail of other types.

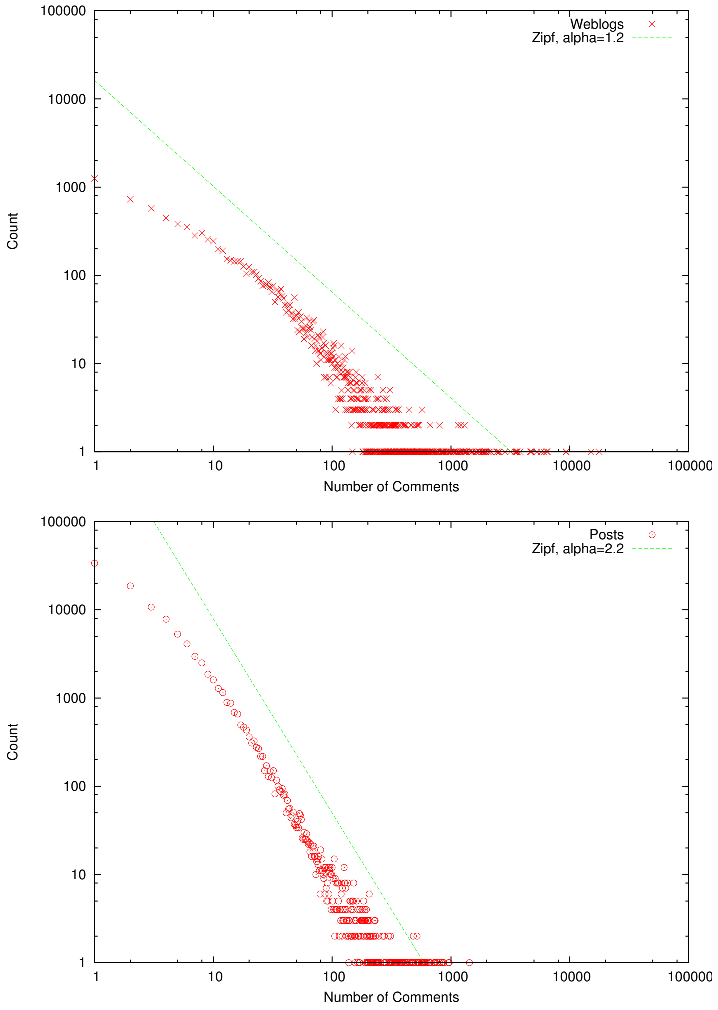


Figure 7.3: Distribution of the amount of comments per blog (top) and per post (bottom), compared to power-law distributions. Power laws are shifted to the right for visibility.

Total comment volume. Note that, due to crawling policies, our corpus does not contain blogs from some platforms. Some of these blogs—in particular those hosted on social-networking-oriented sites such as LiveJournal—contain more comments than other blogs, due to their community-based structure. To understand the commenting patterns in such domains, we revisited the corpus used in Section 3.5.2: a collection of all blog posts published in a large LiveJournal-like platform during 6 weeks, with all the comments posted to them (this collection was supplied by the blogging host, and therefore includes a complete view of all posts and comments). Of the 367,000 blog posts in this collection, more than 183,000 (50%) were commented; and of 36,000 different blogs, more than 24,000 (68%) had a comment posted to at least one of their posts during this 6 week period. The overall number of comments was 1.4 million—more than three times the amount of posts.

Based on our corpus and the estimates regarding the coverage of the comment extraction process and the amount of missing content as demonstrated by this different collection, we estimate that the number of blog comments in the entire blogspace is comparable to the number of posts in active, non-spam blogs: this means that the total number of comments is somewhere between 15% and 30% of the size of the blogspace. At the time of writing (late 2006), according to blog search engines such as Blogpulse, blog posts are added at a rate of over 800,000 a day: assuming our estimates are correct, this means a daily comment volume at an order of 200,000 comments.

On average, comments are shorter than blog posts (in terms of text length); comparing the average length of a comment to the average length of a post in the corpus we described, we estimate that the textual size of the "commentsphere," or "commentspace," is 10% to 20% of the size of the blogspace. Note, however, that influential blogs tend to have more comments than non-influential ones (see Section 7.4); in some cases of top-ranked blogs, the volume of comments far exceeds the volume of the posts themselves. By overlooking comments, much of the conversation around many influential blogs is being missed.

Comment prevalence. An additional issue to consider when studying blog comments is that some blogs do not allow commenting at all. While the vast majority of blogging platforms support comments, bloggers themselves sometimes choose to disable this option, to prevent flaming, spam, and other unwanted effects; other bloggers permit comments, but moderate them by manually reviewing submitted comments before publishing them, or allowing comments from trusted sources only. This, naturally, reduces the overall potential volume of the commentsphere.

Reports on the amount of blogs permitting comments are mixed; a low figure of 43% appears in the random sample examined in [112], while the community-related sample studied in [308] shows that more than 90% of the blogs enabled comments (both studies do not report on the actual number of commented blogs out of those allowing comments). A recent survey [165], mentioned earlier, placed the amount of blogs allowing comments at 87%. An analysis of our collection agrees with the latter figures: a random sample of 500 blogs shows that over 80% of blogs allow users to add comments to the posts, but only 28% of blogs actually had comments posted; as demonstrated earlier, both of these figures are likely to increase if including social-networking-oriented blogs such as LiveJournal, which are often commented. The increase in comment prevalence, compared to [112], can be attributed to the development of blogging software in the 2.5-year period between the two studies: more and more blogging platforms adopted features such as commenting as the standard.

7.2.3 Links in Comments

Overall, our comment corpus contained slightly more than 1 million HTTP links, an average of 1.6 links per comment. This number includes “signature links”—links that the comment author leaves as identification, in many cases linking back to her blog. For the same time period, blog posts themselves contained close to 20 million links. A examination of the top-linked-to domains in comments, in compared with the top-linked-to domains in posts, shows similar results: the top domains are blog communities such as `blogger.com` and `xanga.com`, data sharing websites such as `flickr.com` and `groups.yahoo.com`, news sites, and large retailers such as `amazon.com`. We found no substantial differences between the linking patterns in comments and in posts, and do not expect comments to contribute significantly to algorithms involving link analysis of blogs.

Having said that, in some blog domains (e.g., MySpace, LiveJournal) there is very little direct linking from post to post, and social behavior is centered instead around commenting. Thus, following the commenting behavior in these domains is crucial for understanding the social network and identify communities. In such domains, commenting can be mapped to linking—after which link-analysis methods used in link-rich blogs can be applied.

7.3 Comments as Missing Content

Following our initial analysis of the amount and volume of comments, we turn to evaluate to what degree the absence of blog comments from blog corpora affects real-life blog access. One task which is a good test case for this is blog search—retrieving blog contents in response to a specific request from a user. The general topic of blog search will be addressed in Part III of this thesis; in this Section, we investigate one aspect of it only—the contribution of comment content in this context.

Methodology. To understand the impact of including comment data in the retrieval process, we used two separate collections: the first is the comment corpus we have just described. The second collection is a subset of the Blogpulse index, a comprehensive index of the blogspace, containing all blog posts from the same period as the one for which comments were collected: a period of 20 days in July 2005. While our comment index contains 645,000 comments, the blog post index contained over 8 million posts (this number includes spam blogs and blogs with an infrequent, low level of posting—making it higher than the total number of posts shown in table 7.2).

We then collected a set of 40 queries submitted to the blog search engine at Blogpulse.com during the same 20-day period as those included in the collections. For each of these 20 days, we randomly selected two queries from the most popular

5 queries submitted by Blogpulse users during that day.[4] Example queries from this set are "space shuttle" (July 14th), "Clay Aiken" (July 16th), and "Mumbai floods" (July 29th).

Finally, we use the query set to retrieve a set of matching posts from each of the collections. For the blog post collection, each post was indexed separately and is the unit of retrieval. For the comment collection, comments are the unit of retrieval, not posts. In order to compare search results, we transform the list of returned comments into a list of posts: each retrieved comment is replaced with its parent blog post; multiple comments from the same post retrieved in response to a query contribute, then, a single result to the final ranked list (and are ranked according to the most highly-ranked comment belonging to them). The retrieval model used in both cases is the default Lucene model, a simple tf·idf approach (see [106] for a full description). As we will demonstrate in Chapter 9, this basic ranking model can be substantially improved for blog retrieval; this Section, however, focuses on the relative contribution of comments to existing approaches, rather than developing new ones.

We focus on comparative results—testing the difference in performance with and without the comment data—and regard the absolute numbers we obtain in the experiments as secondary only.[5]

Evaluation. We evaluate the contribution of comment content to the retrieval process on two aspects: coverage and precision.

Coverage is simply the number of returned results; note that this is different from the more traditional metric of recall—the number of *relevant* returned results. Estimation of the latter requires large-scale assessment efforts, and will be examined more closely in Chapter 9. Coverage, however, is an important measure for user-oriented comparisons of search engines, particularly web-related ones (e.g., [24]). Coverage is particularly important for blog search engines evaluations, since blog searchers tend to view results sorted first by recency, then by relevance—in other words, they may be more interested in complete coverage over the recent hours or days than in web-style relevance estimations (cf. [187]). Comparing coverage is straightforward: let P be the set of posts retrieved using their content, and C the set of posts retrieved through their comments. To measure the impact of comment on coverage, we examine $|P|$, $|C|$, and the sizes of their intersection and union—which tell us, respectively, how many posts are

[4] By "most popular," we mean queries which were submitted by the largest amount of different users, rather than queries which recurred most; this was done to address automated (subscription) queries, that may appear frequently in the search log although submitted by few users. More on this type of queries, and the difference between query popularity and frequency, follows in Chapter 8.

[5] An alternative, preferred methodology is to compare not a separate collection of posts and comments, but a collection of posts without their comments and a collection of the same posts, including their comments; this was not done due to technical reasons.

retrieved using their contents; how many posts are retrieved using the contents of their comments; the number of posts retrieved by both methods; and the number of posts retrieved by either approach. Clearly, comparing P to $P \cup C$ gives us a simple indication of the contribution of comments to coverage.

In the case of precision—the fraction of relevant results out of the returned ones—as in many evaluations of retrieval results in web settings, we concentrate on early precision; specifically, we measure precision at 10 (P@10): the fraction of relevant results out of the top-ranked 10 posts. However, evaluating the difference in precision between P and C is more complicated than comparing coverage. Since we test early precision only, we are not facing the union of sets scenario as we did in the case of coverage. We now look at P and C as ranked lists rather than unordered sets, and focus on how C can be used to improve the ranking order in P, thus increasing precision.

7.3.1 Coverage

As described earlier, to measure the impact of comment data on coverage, we compared the list of posts retrieved by searching the post index itself and the list of posts retrieved from the comment index (as noted earlier, multiple comments from the same post permalink were considered as a single hit for that permalink). For each query, we analyzed the overlap between the lists, as well as the contribution of each source separately. For example, for the query "space shuttle," a total of 7,646 permalinks were retrieved from both indices; of these, 7,482 (96.9%) were retrieved from the post index only, 164 (2.2%) were retrieved from the comment index only, and 74 (0.9%) were retrieved from both.

	Posts Only	**Comments Only**	**Both**
Mean	93.1%	6.4%	0.5%
StdDev	9.1%	8.7%	0.7%
Median	96.9%	2.6%	0.2%
Minimum	64.3%	0%	0 %
Maximum	100%	33.3%	2.4%

Table 7.3: Contribution of comments to coverage.

Table 7.3 shows the aggregated results over all 40 queries, using the same percentage view as used in the example. Keeping in mind that our corpus is estimated to contain around a quarter of all comments posted during the period (whereas our post corpus is more or less complete), we see a notable contribution of content in comments to the overall coverage. Extrapolating our observations to account for the comments which are not in our corpus as a result of the extraction process, we estimate an addition of 10%–20% "hits" for a query on average, given a complete index of all blog comments; the median addition would be lower at 5%–15%, due to a small number of queries with very high contributions from comment contents (in our experiments, these included both queries with many hits such as “rss” and queries with few ones, such as “Tighe”). In particular, it is interesting to note the relatively small overlap between the results of the comment search and the post search—suggesting that comments often add new terms to the contents of the post, terms which assist in retrieving it given a query.[6] Also worth noting is the high standard deviation of the contribution of comments to coverage, indicating that, for some queries, comment content is vital for comprehensive search performance in the blogspace.

7.3.2 Precision

Most commercial blog search engines present their results sorted by date, assuming that recent results are of higher importance to the searcher; typically, results from the same date are sorted according to some static ranking of blogs, based on an estimation of the blog’s popularity. We will revisit this ranking scheme in Part III, examining how well suited it is to the needs of blog searchers, and show that it can be improved. However, in this Section we focus on the contribution of comments to current retrieval settings in the blogspace.

Examining the top results obtained with this default ranking scheme on the blog post collection in our test set, we experienced an average P@10 of 0.55: the majority of top-ranked posts were indeed relevant for the queries. The accuracy of the same ranking scheme used on the comment index was lower, with average P@10 of 0.28. Following successful work on combining ranked lists created by different retrieval approaches discussed in [273], we experimented with various combination operators, but overall improvement was minor: the best performance, an average P@10 of 0.58, was measured with the CombSUM method (a simple addition of the two normalized retrieval scores), but was not statistically significant.

However, comments may contribute to precision in a different way: blog searchers are, sometimes, interested in more than the topical relevance usually used to evaluate retrieval. Analyzing a community blog, Krishnamurthy [154] observes that “the number of comments per post is perhaps the truest and most diagnostic metric of the nature of communication on a blog. The posts that are most insightful or controversial get the most comments. Those that are pedestrian do not get many comments.” This led us to believe that while topical precision itself is not greatly modified by using comments, they do provide access to a different perspective of blog posts, namely, the impact on their readers.

Evaluation of this new precision angle is complicated, and will be addressed more thoroughly in Part III of this thesis. For now, we support our claim anec-

[6] In several cases, we observed almost-empty posts, containing just a link to an article or another web page with a short remark such as “unbelievable;” the comments to the post contained actual content and keywords, supplying the context to the post and enabling its retrieval.

dotically, showing that usage of comments can indeed lead to a different way of addressing relevance. To do this, we experimented with a method for reranking the top 100 results produced by the "standard" ranking method according to the number of the comments associated with the blog posts: the normalized number of comments per post was used as a retrieval result, and combined with the retrieval score of the post—again, using the methods described in [273].

An examination of the top-10 ranked results using this combination showed similar early precision; average P@10 was 0.53. However, we found that this method, while preserving the same early precision levels as the "standard" ranking method, produces top-ranked results which are more discussion-oriented, attracting more feedback from users, suggesting that in scenarios where users seek such discussions (such as the discussion search task user scenario at the TREC Enterprise track [57]), it may be beneficial. We will return to this method and evaluate it more thoroughly in Chapter 9.

7.4 Comments and Popularity

Cursory examination of blogs, as well as intuition, suggests that the number of comments is indicative of the influence level a blog or post has—the degree to which it is read, cited, or linked to. In this section we attempt to substantiate this observation empirically, and understand the cases where it does not hold.

To measure blog popularity we use two indicators: the number of incoming links as reported by the Blogpulse index, and the number of page views for blogs that use a public visit counter such as Sitemeter[7]—their "readership." In total, there were 8,824 blogs for which we had both readership and inlink information [277]; of these, we found comments in 724 blogs.

First, we measured the pairwise Pearson r-correlation between the three measures: comment amount, readership, and incoming link degree. While the correlation between readership and indegree is high, the amount of comments correlated poorly with both. However, when examining only blogs for which we found comments, the correlation was substantially higher, leading us to believe that the results for all blogs are biased because of our partial comment extraction mechanism. The pairwise correlation values for both cases are summarized in Table 7.4.

The correlation between readership or indegree in commented blogs is relatively high, but still lower than the correlation between readership and indegree themselves. One possible reason is that the relation between comments and popularity in a blog is not a linear one: the influence of a blog does not correlate directly to the number of comments posted to it, but, on average, highly-commented blogs are more influential than less commented ones. To investigate this claim, we again measured the relation between the three figures—indegree, readership, and

[7] http://www.sitemeter.com

		Correlation	
Variable 1	**Variable 2**	**All posts**	**Commented posts**
Indegree	Readership	0.79	0.77
Indegree	Number of comments	0.18	0.58
Readership	Number of comments	0.17	0.60

Table 7.4: Pearson-r correlation between comments, readership, and indegree, for all data and for blogs for which comments were found.

amount of comments—this time bucketing the amount of comments into higher-level groups. Tables 7.5 and 7.6 compare the number of incoming links and page views for blogs with no comments and blogs with varying levels of comments.

Number of comments	**Count**	**Average page views**		**Average incoming links**	
0	8,104	453.7		66.7	
> 0	724	812.9	(+79%)	267.1	(+300%)
Breakdown:					
1–10	186	423.2	(−7%)	130.4	(+95%)
11–50	260	485.3	(+7%)	158.5	(+137%)
51–100	115	650.8	(+43%)	261.2	(+291%)
101+	163	1,894.6	(+317%)	600.3	(+800%)

Table 7.5: Blog popularity as relating to the number of comments; all percentages are in comparison with non-commented blogs.

Average comment length (words)	**Count**	**Average page views**		**Average incoming links**	
0	8,104	453.7		66.7	
> 0	724	812.9	(+79%)	267.1	(+300%)
Breakdown:					
1–10	46	782.4	(+72%)	327.7	(+391%)
11–50	291	388.3	(−14%)	156.6	(+136%)
51–100	260	978.5	(+116%)	309.1	(+363%)
101+	127	1,457.8	(+221%)	412.2	(+518%)

Table 7.6: Blog popularity as relating to the average length of comments; all percentages are in comparison with non-commented blogs.

Clearly, commented blogs are substantially more read and linked to than those having no comments. However, this is a chicken-and-egg situation: assuming a fixed percentage of blog readers post comments, blogs which have more incoming links and more readers are more likely to have higher amounts of comments. Nevertheless, the existence of many comments in a blog post is a clear indication of the popularity of the post, and unlike other measures (such as indegree count) does not require analysis of the entire blogpace.

7.4.1 Outliers

We witnessed an overall good correlation between the level of comments and the blog popularity on average; but we also encountered various exceptions: highly-ranked blogs with no or little comments, low-ranking blogs with many comments, and so on. We now discuss some of these cases.

"Too few" comments in high-ranked blogs. Many blogs, particularly highly-ranked ones, impose some moderation on reader comments, or disable them altogether; this is typically done to prevent spam and other forms of abuse. Of the top-10 ranked blogs with no or few comments we checked, all employed some sort of comment moderation, leading us to believe that these outliers are mostly due to this technical limitation.

"Too many" comments in low-ranked blogs. Most blogs that appeared to have substantially more comments than expected given their viewership and incoming link information turned out to be blogs of the personal-journal flavor, where a relatively small group of the blogger's friends used the comment mechanism as a forum to converse and interact. Many of these comments did not relate directly to the post, and resembled a chat session more than other comment threads in our collection.

An additional class of blogs which have a high number of comments, given their link indegree, consisted of blogs that are popular with the non-technical crowd, such as fashion or celebrity blogs—presumably, readers of these blogs tend to use links less than the more technologically-oriented readers (or, alternatively, do not blog at all).

Highly-commented posts in a given blog. Some posts in our corpus have a very large number of comments, compared with the median of that blog. In general, it seems such posts are either related to highly-controversial topics (usually, politics), or posts which were cited in mainstream media or in other sources (such as influential blogs), directing a high level of traffic towards them.

7.5 Discussions in Comments

Blog comments provide a rare opportunity to explore how users respond to online content. Excluding blogs and wikis, feedback on web sites is typically submitted through forms and email, and is not publicly available. A small number of personalized websites have guestbooks—a leftover from earlier internet days—but even those are used to provide feedback about the entire site, rather than about a particular topic or section. In contrast, blogs which allow commenting allow direct, personal, mostly unmoderated discussion of any post in the blog.

`mainstreambaptist.blogspot.com/2005/07/neo-con-plan-to-help-military.html`

Post:
The Neo-Con Plan to Help the Military
The New York Times published an article yesterday, "All Quiet on the Home Front, and Some Soldiers are Asking Why," that has a paragraph revealing the neo-conservative's plan to assist the military fulfill its mission in Iraq. Here it is . . . It will be interesting to see how the bankers and lawyers and doctors and engineers in Oklahoma respond to this call for support. If they can't sell their program here, they can't sell it anywhere.
Comments:
1. It's about time all those that voted for the warmonger in charge to put up or shut up.
2. Bruce, this is exactly what my son, Spc. ccsykes, was talking about when he made the following comment on your blogpost - "Iraq Imploding" - "Lack of support from the people of the United States, low morale in the Military and our policies have already lost this war."
3. Marty, you are right and so is ccsykes.
4. One of the more shocking moments, I thought, was when Bush counseled us to go out and shop in response to the ramping up of terrorism. Though I want us out of Iraq as soon as possible, I think we owe Iraq the . . .
5. ditto

Table 7.7: A non-disputed blog post, according to the comments.

Examining our comment collection, we identified various types of comments—among them personal-oriented ones (posted by friends), comments thanking the author for raising an interesting issue or pointing to additional related content, and so on. One class of comments we found particularly interesting was the set of *disputative* comments: comments which disagree with the blogger (or with other commenters), forming an online debate. We hypothesized that these comments can be used to identify controversial topics, authors, newspaper articles, and so on. An example of two comment threads from the same blog appear in Tables 7.7 and 7.8; the first contains no dispute, while the second demonstrates a disputative discussion.

In this section, we attempt to identify this type of comments computationally; we address this as a text classification task, and focus on features which are useful in this setting. We view this as a demonstration of the type of information which can be mined from comments with text analysis methods, and the usefulness of comments as adding new knowledge in the blogspace.

`mainstreambaptist.blogspot.com/2005/07/reclaiming-americas-real-religious.html`

Post:
Reclaiming America's Real Religious History
Kudos to Marci Hamilton at AlterNet for her outstanding article on "The Wages of Intolerance." She does an outstanding job of reclaiming America's real religious history from the revisionists who want to make . . .
Comments:
1. I think that the author's candor about American history actually undermines her argument. First, she . . . ...
2. Anon, it is obvious to me that you don't know Bruce personally. He speaks the truth. Perhaps one day the scales will fall off your eyes as well . . .
3. Perhaps Bruce could be more persuasive in proving his admittedly controversial (and I would say wild) assertions.
4. I've given a little thought to something I wrote to Bruce earlier: "You yourself seem to be guilty of wanting to impose . . . " It would be absolutely futile for me to attempt to engage in a reasonable discussion there; it is just as futile to do the same here.
5. I've watched with great interest as this blog has evolved from a discussion of Mainstream Baptist concerns and demoninational issues into a hyper-political, left-wing campaign against . . .
6. you can always tell that someone does good work, because someone is going to get angry about it. all this man is doing is standing up to what he believes are historic baptist principles.
7. mt1, I suggest that you read the description of the blog that is the top of each page. I also sign my full, real name to everything I write.
8. Anonymous, commenting on your comments has been very theraputic for me. God bless you and good luck on your new . . .

Table 7.8: A comment thread including disagreement.

7.5.1 Detecting Disputes in Comments

First, we describe the components of this text classification task.

We manually annotated 500 comment threads—randomly selected from the set of threads containing at least 5 comments (the examples in Tables 7.7 and 7.8 are taken from our manually annotated data). A thread was labeled "disputative" if the discussion contained within its comments was in the form of a debate, or if tension between the blogger and the commentators (or among themselves) was apparent in the form of strong language, attempts to discredit or refute others. In total, 79 (16%) threads in the set were marked as disputative. We then trained a decision tree boosted with AdaBoost with this data.[8] We follow with a description of the features used for the classification process.

[8] We experimented with other types of classifiers, with similar but slightly worse results.

Feature set.

- **Frequency counts.** The basic and most popular feature set used in text classification tasks, which we have also used for other classification tasks in this thesis, e.g., in Chapter 4. We used counts of words and word bigrams in the comments, as well as counts of a manually constructed small list of longer phrases typicallly used in debates ("I don't think that," "you are wrong," and so on).

- **Level of Opinion.** With a large amount of training data, the frequency counts would have captured most important words and phrases distinguishing controversy from other discussions. However, given our limited training data, we chose to measure the degree to which opinion is expressed in the comments separately from the frequency counts. The level of opinion is the extent to which a personal statement is made in the text; we will return to this notion in much more detail in Chapter 9, which addresses retrieval of blog posts containing opinions. For simplicity, though, one aspect setting opinionated content apart from objective content is language use, with phrases such as "I believe that" and "In my opinion" appearing in opinionated comments more frequenctly than in other comments.

  To capture this type of language, we again utilized comparisons of two language models as in Chapters 3 and 6, this time seeking to computationally identify the terms and phrases that are typical of subjective content. Our training set, however, is too small to effectively compare language models and extract meaningful terms; a substantially larger corpus of general-domain content, annotated as subjective and or objective, is required. The collection we use for this task is the English part of the online encyclopedia Wikipedia. Every entry in this encyclopedia has, in addition to the basic encyclopedic content, a separate "user discussion" page, where users are encouraged to discuss and debate the contents of the article. This is, indeed, a large-scale corpus: at the time of performing our experiments (late 2005), the entries themselves consisted of 2GB of text, and the discussions of an additional 500MB.

  We constructed unigram, bigram and trigram language models for both parts of this collection—the entries and the discussion pages—and compared them, as we did in previous chapters, using the log-likelihood measure. The top phrases found using this comparison include "I don't," "you have to" and similar opinionated terms and phrases; we aggregated this into a lexicon of opinion-bearing expressions. This list was then used, as in lexicon-based approaches to sentiment analysis (e.g., [64]), to derive an opinion level for the entire comment thread; we take a naive approach and simply sum the log-likelihood values of the opinionated phrases occurring in it, based on the two corpora we described.

- **Length Features.** Observing that disputative comments tend to be longer and appear in longer threads, we added features for the average sentence length, the average comment length in the thread, and the number of comments in the thread.

- **Punctuation.** We used both frequency counts of the various punctuation symbols in the text, and special features indicating usage of excessive punctuation (this has been shown to be effective for certain text classification tasks, e.g., [260]). Noting that some disputative comments begin with questions, we added separate features for the punctuation symbols used in the first sentence of the comment only.

- **Polarity.** The sentiment analysis method described in [222], which we also used in Chapter 6, was used to identify the orientation of the text of the comments. The intuition here is that disputes are more likely to have a negative tone than other types of discussion.

- **Referral.** While studying our corpus, we noticed that comments which disagree with the blog author (or with another commenter) contain, in some cases, references to previous content or authors. Typical such references are a quote (from the blog post or from another comment), or referral to previous authors by name.

  To capture this, we used a rule-based mechanism to detect referrals, implemented through regular expressions. Rules were crafted to identify repetitions of sentences from the post in a comment, usage of quoted text within a comment, and references to names appearing in the signature of a blog or a comment. For example, one rule checked whether text appearing in a comment within a `blockquote` tag—a tag often used to quote text from an external source—appeared earlier in the comment thread, or in the post itself. If any of these rules identified a referral, the entire comment thread was marked as containing a referral; the feature used for the learning process was, then, a binary one ("has referral" or "does not have a referral").

7.5.2 Evaluation

Using a 10-fold cross validation on our manually annotated corpus for evaluating the classifier, we obtained an accuracy of 0.88, as shown in Table 7.9. As this is an unbalanced distribution, comparison to a baseline is difficult (see, e.g., [210])—a maximum-likelihood classifier would have achieved an overall F-score of 0.84 by classifying all threads as non-disputative, but would have little meaning as a baseline as it would have yielded an F-score of 0 on the disputative comments only.

The following are the most important features utilized by the classifier, in decreasing order of importance:

	Precision	**Recall**	**F-Score**
Non-disputative comments	0.92	0.96	0.94
Disputative comments	0.72	0.58	0.65
Overall	0.88	0.89	0.88

Table 7.9: Accuracy of dispute detection in comments.

- Existence of a referral in the comments
- Usage of question marks in the first sentence
- Counts of phrases from the manually-built disagreement lexicon
- Number of comments in the thread
- Level of opinion

Among the words which were relatively important features are pronouns and negating words such as “not” and “but.”

Using the classifier on the entire comment corpus resulted in close to 21% of the comment threads being tagged as disputative, suggesting that comments are indeed used, in many cases, for argumentative discussions. Anecdotal examination of the disputed comment threads, in particular those assigned a high confidence by the classifier, revealed that these threads do contain a fair amount of controversial discussions. Table 7.10 contains the top terms appearing in disputed comment threads (excluding stopwords), showing what is fueling arguments in the blogspace; clearly, politics prevail as the central topic of debate.

Iraq	Government	Money	Country
America	Political	Bush	Women
Power	White House	Church	Media
President	School	United States	Children
Muslims	The Media	Supreme Court	The Constitution

Table 7.10: Disputed topics, according to comment threads.

7.6 Conclusions

This Chapter investigated a domain often neglected in computational studies of blogs—the comments posted in response to blog posts. We focused on three aspects: a study of the “commentspace” in terms of volume and characteristics; the contribution of comments to existing blog access tasks; and new types of knowledge which can be identified through comments.

In terms of characterizing comments and their properties, we found that they constitute a substantial part of the blogspace, accounting for up to 30% of the volume of blog posts themselves. We discuss comments as an indicator of the popularity of blog posts and blogs themselves, and find—as expected—empirical evidence that a wealth of comments in a blog is a good indication for the influence of the blog.

In terms of the contribution of comment content to blog access tasks, we focus on the search task and show that usage of comments improves coverage, sometimes significantly, and is also beneficial for rankings which are based not on topical relevance only.

Finally, demonstrating the type of knowledge that can be mined from comments, we describe a text classification approach to determining the level of debate following a blog post by analyzing its comments. Applying the resulting classifier to the entire blogspace, we can identify those topics that bloggers find most controversial.

Conclusions for Part II

In this part of the thesis, we moved from analytics of single blogs to methods aimed at analyzing multiple blogs. We started with a demonstration of the power of aggregate knowledge, showing that sentiment analysis of multiple blogs improves marketing prediction tasks. We continued to explore sentiment in the blogspace, revisiting mood classification—a task addressed in the previous part at the level of single blogs, with moderate success only. We show that with the redundancy of information found in large collections of blogs, this becomes a tangible task, with substantially better results. We followed by using the mass of moods in the blogspace to develop and explore new tasks—identifying regularities and irregularities in sentiment on the blogspace, and using the language of bloggers to explain changes in global emotions. Finally, we analyzed a secondary corpus, hidden inside the blogspace: the collection of all blog comments, the "commentspace." We demonstrated that this collection shares some properties with the blogspace itself, but can also be used for new analytical tasks, as well as to improve existing analysis of blogs.

As in the previous part, we utilized statistical language modeling in a several scenarios—and, in particular, extraction of terms based on language models. Other techniques used on top of this included regression analysis, model-based information extraction, and information retrieval approaches.

Our main observation in this part is that the blogspace is more than the sum of its parts. Tasks which are difficult to address at the single blog level (e.g., mood classification) become feasible with access to large-scale collections; additionally, new tasks altogether arise (e.g., product success prediction). But as the amount of available data grows, it also becomes more difficult to filter out irrelevant information. Although we did not discuss this explicitly, some of our methods, both for analyzing sentiment and for studying comments, have been substantially affected by the amount of noise in the data.

Separating relevant and irrelevant content in large-scale collections is the goal of information retrieval—an area we have utilized both in this part and the previous one, but not worked directly on. In the next and final part of this thesis, we focus on this domain, recognizing that the scale and character of the blogspace requires separate investment in understanding information retrieval in blogs.

Part III

Searching Blogs

The exponential growth in the amount of information available online led many users to rely on search technology, rather than navigation aides, for accessing the web (see, e.g., [246]). With similar growth in the number of blogs, this pattern is migrating to the blogspace as well. In the early days of the blogspace, blog directories—some claiming to list all blogs—were common; at a stage where the blogspace contains millions of blogs, search is becoming more and more important for blog readers. Consequently, a broad range of search and discovery tools for blogs has emerged in recent years. Pioneers in this area were focused exclusively on blog access (e.g., Technorati and BlogPulse); later, large scale web search engines such as Google or Yahoo developed their own specialized blog search services.

This final part of the thesis investigates blog search, focusing on elements which make it different from other types of web search. The part consists of two chapters: first, Chapter 8 studies the search behavior in the blogspace, showing the similarities and differences between blog searches and web searches. Based on this, Chapter 9 addresses a search task which is relatively unique to blog search: identifying and ranking blog posts expressing an opinion about the terms in the query, rather than just information about them.

Chapter 8

Search Behavior in the Blogspace

When we started exploring search in the blogspace, little information, if any, was publicly available about the type of information needs raised by blog searchers; blog search was viewed as one particular form of web search. The purpose of this Chapter is to understand how blog search differs from web search in terms of search behavior. More specifically, we address the following questions:

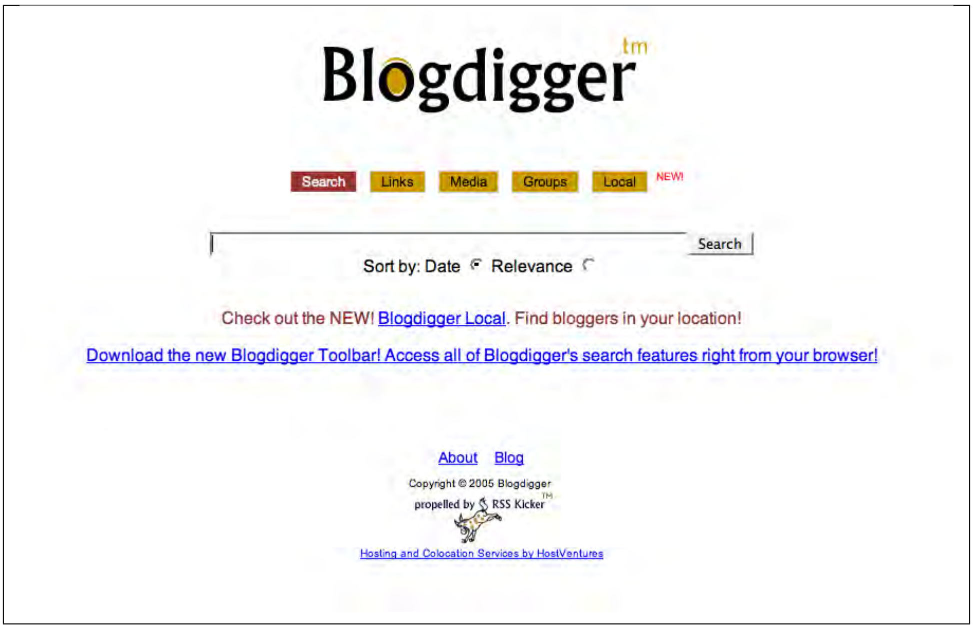
1. Which information needs lead a user to submit a query to a blog search engine, rather than to a general web seach engine? Which topics are users interested in?
2. What is the behavior profile of the blog searcher, in terms of popular queries, number of result pages viewed per query, frequency of searches, and so on?

Our analysis follows the large body of work in the area of search engine log analysis: a recent survey paper refers to numerous studies in this area published during the last 10 years [77]. In particular, we are guided by Broder's work on classifying search requests of web users using the (then popular) AltaVista search engine [40], as well as the follow-up work by Rose and Levinson with Yahoo data [255]. In terms of statistical analysis, we follow one of the first large-scale studies of search logs available to the public, carried out by Silverstein et al. [278], as well as the numerous analyses published by Jansen, Spink et al., which targeted various angles of search engine usage (e.g., [130, 129, 40]).

We proceed as follows. First, we describe the query log we used, its source, and the distribution of queries in it. Sections 8.2 and 8.3 follow with a comparison of the queries found in our data to queries found in web search engine logs: the first section identifies and compares the query types, and the second examines the popular queries issued to blog search engines, compared to those used in web search engines. Next, Section 8.4 categorizes the queries by topic, presenting a novel categorization scheme. Finally, Section 8.5 focuses on search behavior in terms of sessions: the number of queries submitted in a single visit to the search engine, the number of results examined, and so on. Section 8.6 concludes this Chapter.

8.1 Data

The data we use to analyze search patterns in the blogspace consists of the full search log of Blogdigger for the month of May 2005. Blogdigger is a search engine for blogs and other syndicated content feeds that has been active since 2003, being one of the first fully-operational blog search engines. As major web search engines introduced their capabilities for blog search in late 2005, it gradually became a second-tier engine. At the time the data we have was collected, however, Blogdigger was widely used, and provided some unique services in the blog search world. Some of these services, including location-based search (search blogs in a given geographic area) and media search (locate multimedia files such as images and videos in blogs), are shown in a screen capture of Blogdigger, taken at the time the searches in the log were done (Figure 8.1). The data we study here contains the queries sent to all these services.



Sort by: Date ● Relevance ○

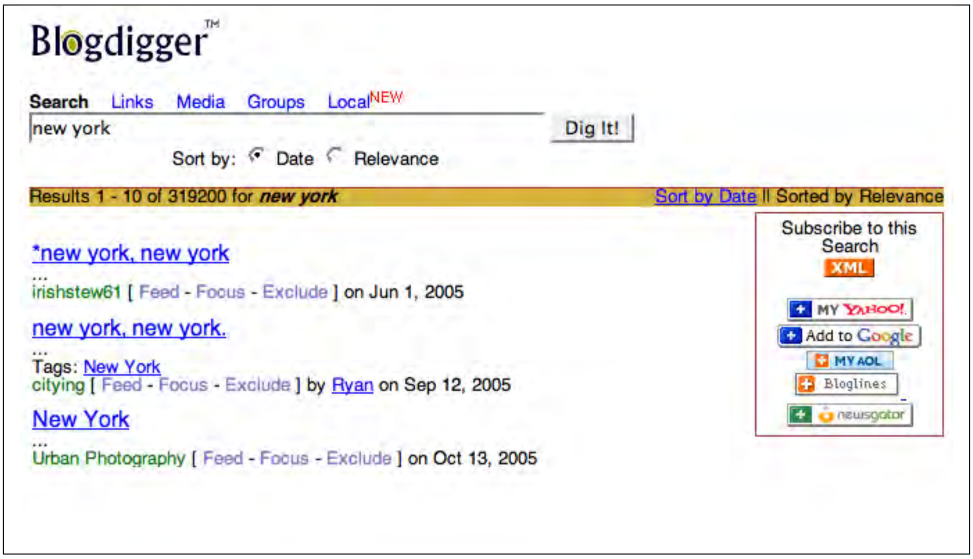
Check out the NEW! Blogdigger Local. Find bloggers in your location!

Download the new Blogdigger Toolbar! Access all of Blogdigger's search features right from your browser!

Copyright © 2005 Blogdigger
propelled by RSS Kicker™
Hosting and Colocation Services by HostVentures

Figure 8.1: Blogdigger homepage at the time our data was collected.

Like other commercial blog search engines, Blogdigger serves both ad-hoc queries and subscription queries. Ad-hoc queries originate from visitors to the search engine's web site, typing in search terms and viewing the result pages, in a similar manner to the typical access to web search engines. A user who is interested in continuous updates about the results of a specific query can subscribe to its results; Figure 8.2 shows a typical result page, including subscription options displayed to the right of the results. In practice, a subscription means that the user intends to add a request for a machine-readable version of the query results to a syndicated content aggregator (e.g., an RSS reader) she is running. The query results will then be periodically polled; each of these polls is registered as a subscription query in the search log. We will refer to these subscription queries using their more common IR name—filtering queries.



Blogdigger™

Search Links Media Groups Local NEW

new york

Sort by: ● Date ○ Relevance

Results 1 - 10 of 319200 for *new york* Sort by Date || Sorted by Relevance

*new york, new york
...
irishstew61 [Feed - Focus - Exclude] on Jun 1, 2005

new york, new york.
...
Tags: New York
citying [Feed - Focus - Exclude] by Ryan on Sep 12, 2005

New York
...
Urban Photography [Feed - Focus - Exclude] on Oct 13, 2005

Subscribe to this Search

Figure 8.2: Subscribing to Blogdigger results.

Descriptive statistics of the search log are shown in Table 8.1. Due to the large percentage of duplicates typical of query logs, statistics are listed separately for all queries and for the set of unique queries in the log (i.e., exact repetitions removed). While filtering queries make up the bulk of all queries, they constitute a relatively small amount of unique terms, and the majority of unique queries originate from ad-hoc sessions. The mean terms per query number for (all) ad-hoc queries, 2.44, is comparable to the mean terms per query numbers reported in the literature for general web search (2.35 [278], 2.21 [128], 2.4–2.6 [285], and 2.4 [129]); while the mean terms per query number for filtering queries appears to be somewhat smaller (1.96), a closer examination reveals that this difference is caused to a large extent by two specific clients; excluding these outliers, the mean terms per query for filtering queries is 2.5, similar to that of ad-hoc ones.[1]

8.2 Types of Information Needs

We now address the first of the questions presented in the beginning of this chapter, namely, identifying the types of information needs expressed by users in the blogspace.

Queries submitted to web search engines are usually grouped into three classes: *informational* (find information about a topic), *navigational* (find a specific web

[1] The two clients issued large amounts of queries in fixed, short intervals; the queries appear to have been taken from a dictionary in alphabetical order and are all single words, pushing down the mean number.

	All queries		Unique queries	
Number of queries	1,245,903		116,299	
Filtering queries	1,011,962	(81%)	34,411	(30%)
Ad-hoc queries	233,941	(19%)	81,888	(70%)
Text queries	1,016,697	(82%)	50,844	(44%)
Media queries	229,206	(18%)	65,455	(56%)
Link queries	2,967	(<1%)	562	(<1%)
Mean terms per filtering query	1.96		1.98	
Mean terms per ad-hoc query	2.44		2.71	

Table 8.1: Search log size and breakdown.

site), and *transactional* (perform some web-mediated activity) [40]. This is not necessarily the appropriate classification for queries submitted to blog search engines—clearly, transactional queries are not a natural category for blog search, and a user searching for a particular site, or even a particular blog (i.e., submitting a navigational query) would not necessarily use a blog search engine, but rather a general-purpose web engine. Our working hypothesis, then, is that the majority of blog queries are informational in nature, and a scan of the search log confirms this.

Given this assumption, is it possible to identify different types of informational queries submitted to a blog search service? Ideally, this would be done using a user survey—in a manner similar to the one performed by Broder [40]. This requires substantial resources (e.g., access to the front page of a search engine, to attract participants); instead, we rely on Broder's observation of a good correlation between the results of such a survey and manual classification of a subset of the queries. We therefore manually classify such a subset, and the analysis we present is based on this classification.

More concretely, we analyzed two subsets from the list of queries, as follows. First, we examined a random set of 1,000 queries, half of which were ad-hoc queries and half filtering ones, so as to discover likely query types. We observed that the majority of the queries—52% of the ad-hoc ones and 78% of the filtering ones—were named entities: names of people, products, companies, and so on. Of these, most (over 80%) belonged to two types: either very well-known names ("Bush," "Microsoft," "Jon Stewart"), or almost-unheard-of names, mostly names of individuals and companies.[2] An additional popular category of named entities was location names, mostly American cities. Of the 48% of queries which were not named entities, most queries—25% of the ad-hoc queries and 18% of the filtering ones—consisted of high-level concepts or topics, such as "stock trading," "linguists," "humor," "gay rights," "islam" and so on; the filtering queries of this type were mostly technology-related. The remainder of the queries consisted of adult-oriented queries (almost exclusively ad-hoc queries),

[2] The prevalence of the named entity was established using search engine hit counts: well-known names typically had millions of hits; unknown names had few if any.

URL queries, and an assortment of queries with no particular characteristics.

Next, we examined the 400 most common queries in the set of unique queries (again, half of them ad-hoc queries and the rest filtering ones), to find out whether the query types there differ from those found in the "long tail" of queries. While the types remained similar, we witnessed a different distribution: 45% of the ad-hoc queries and 66% of the filtering queries were named entities; concepts and technologies consisted of an additional 30% of top ad-hoc queries and 28% of filtering ones. Adult-oriented ad-hoc queries were substantially more common in top ad-hoc queries than in the random set.

Consequently, our hypothesis regarding the intents of blog searchers divides the searches into two broad categories:

- **Conversational Queries**: The purpose of these queries is to locate contexts in which a certain name appears in the blogspace: what bloggers say about it, and what discussions it attracts. Most of the named entity queries have this intent; the well-known names might be entities in which the searcher has an ongoing interest (such as politicians or celebrities), or products she is researching; lesser-known names are typically vanity searches, or searches for contexts of entities which constitute part of the searcher's closer environment (friends or colleagues, an organization of which the searcher is a member, and so on).

- **Concept Queries**: With these queries the searcher attempts to locate blogs or blog posts which focus on one of the searcher's interest areas, or with a geographic area that is of particular interest to the searcher (such as blogs authored by people from his home town). Typical queries of this type are the various high-level concepts mentioned earlier, as well as location names.[3]

Table 8.2 shows a breakdown of both the random set and the top-query set according to query type, for ad-hoc and filtering queries separately. For this breakdown, named-entity queries (except location names) were considered as conversational queries; high-level areas of interest and location names were considered concept queries.

As an aside, while examining the top queries, we observed an interesting phenomenon which we did not witness in the random set: many of the queries were related to events which were "in the news" at the time of the log. This supports the assumption that blogs are conceived as a source of information and commentry about current events [180]. To quantify the number of news-related queries, we used two independent methods. First, a human decided, for each query, whether it was news-related. This was done by studying the terms in the

[3] These queries are somewhat similar to distillation queries as defined by TREC (cf. [303]), with target results being blogs rather than websites.

	Top 1,000 queries		**Random 400 queries**	
Class	**Ad-hoc**	**Filtering**	**Ad-hoc**	**Filtering**
Conversational	39%	60%	47%	73%
Concept	36%	34%	30%	23%
Other	25%	6%	23%	4%

Table 8.2: Query classes: the top 400 queries vs. a random sample of 1,000 queries.

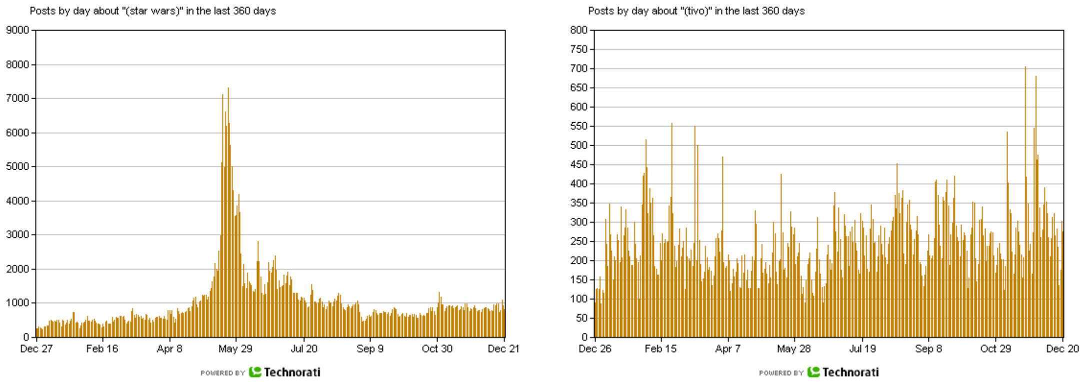


Figure 8.3: Sample daily frequency counts during 2005. Left: a news-related query ("Star Wars," a movie released during May 2005). Right: a non-news-related query ("Tivo"). Source: Technorati.

query, and attempting to locate events related to it that happened during May 2005, the period covered by the log. The second method was an automated one: we obtained daily word frequencies of the terms appearing in the query as reported by Technorati, for the entire year of 2005. Terms which had substantial peaks in the daily frequency counts during May 2005 were considered related to news; sample daily frequencies over the entire year of 2005 are shown in Figure 8.3. The agreement between our two methods was $\kappa = 0.72$.

In total, we found that 20% of the top ad-hoc queries and 15% of the top filtering ones are news-related; in the random set, news-related queries were substantially less frequent, amounting to 6–7% of both ad-hoc and filtering queries.

To summarize, we observe that blog searches have particular properties: an abundance of named entities and focus on recent developments; additionally, some queries consist of high-level concepts or domains. We interpret this as suggesting that the primary targets of blog searchers are tracking references to entities, and, to a lesser extent, identifying blogs or posts which focus on a certain concept.

8.3 Popular Queries and Query Categories

Next, we provide a brief overview of the most popular queries in our data.

Ad-hoc	Filtering	Web
filibuster	Lotus Notes	American Idol
Blagojevich	Daily Show	Google
sex	microcontent	Yahoo
porn	information architecture	eBay
blogdigger	MP3	Star Wars
Madagascar	Streaming	Mapquest
RSS	Google	Hotmail
adult	Wayne Madsen	Valentine's day
Google	Tom Feeney	NASCAR
nude	Clint Curtis	hybrid cars
MP3	digital camera	MP3 players
Los Angeles	DMOZ	NFL
test	desktop search	dictionary
China	manga	Paris Hilton
3G	RSS	Michael Jackson
Star Wars	Abramoff	Hillary Clinton
IBM	knowledge management	heartburn
blog	government	Lohan
music	restaurant	flowers
Bush	information management	Xbox 360

Table 8.3: Top 20 queries. (Left): Ad-hoc blog queries. (Center): Filtering blog queries. (Right): Web queries.

Simply counting the number of times a query appears in our log may yield misleading results regarding the most popular queries, particularly for filtering queries. As described earlier, these are automated searches which are repeated at regular intervals, and agents issuing these queries with high refresh rates will create a bias in the query counts. Consequently, we measure the popularity of a query not according to the number of its occurrences, but according to the number of different users issuing it. As a key identifying a user we use a combination of the IP address and the user agent string (more details on user identification are given in Section 8.5).

The most popular queries in the log according to this approach are shown in Table 8.3, separately for ad-hoc and filtering queries (leftmost column and center column, respectively).

Comparison with Web Queries. To compare the popular queries submitted to blog search engines with those sent to general web search engines, we obtained a set of 3.5M queries submitted to Dogpile/Metacrawler, a second-tier general web search engine,[4] during May 2005—the same timespan as our blog search log. The top 20 queries from this source (which did not include adult-oriented queries) are also listed in Table 8.3 (right column), alongside the top blog searches.

[4] Dogpile/Metacrawler is a metasearch engine, which submits queries issued to it to a number of other engines such as Google or Yahoo, and aggregates their results.

Some differences between the query lists are clear: the web queries contain many popular web sites (Yahoo, eBay, Hotmail, and so on), perhaps because some users use the search bar as a shortcut for reaching various web destinations. Additionally, the top blog queries seem to contain a somewhat higher percentage of political and technology-related queries; we will return to this observation in Section 8.4.

Other differences between blog queries and web queries require examining more than a small number of top queries. Comparing the most popular 400 queries from both sources, we observed a substantially higher rate of named-entity queries within blog queries than in web queries. As mentioned earlier, 45% of ad-hoc blog queries and 66% of the filtering queries were named entities; in comparison, only 33% of the top 400 web queries were named entities, many of which were website names. This suggests that blog searchers—especially those registering filtering queries—are more interested in references to people, products, organizations or locations than web searchers.

We mentioned earlier that we found a relatively large amount of news-related queries among top blog queries; this type of queries proved to be fairly uncommon in general web search engines, accounting for less than 8% of the top 400 queries, and less than 2% of 400 random ones.

An additional difference between the query lists is the presence of very detailed information needs (such as factoid questions or long phrases) in the web query log: such queries were not found among the blog queries. Finally, as is the case with web searches, adult-oriented queries are a major area of interest for ad-hoc blog searchers; however, these are nearly non-existent in filtering queries.

8.4 Query Categories

The next aspect of queries in the blogspace we wish to explore concerns partitioning them into topical categories, and examining the distribution of queries between categories. This allows us to understand the profile of blog searchers: their areas of interest. We begin by introducing our approach to query categorization.

Current approaches to automatic categorization of queries from a search log are based on pre-defining a list of topically categorized terms, which are then matched against queries from the log; the construction of this list is done manually [29] or semi-automatically [244]. While this approach achieves high accuracy, it tends to provide very low coverage, e.g., 8% of unique queries for the semi-automatic method, and 13% for the manual one.

We take a different approach to query categorization, substantially increasing the coverage but (in our experience) sustaining high accuracy levels: our approach relies on external "categorizers" with access to large amounts of data.[5] We sub-

[5] Similar methods to ours have been developed independently in parallel, for the 2005 KDD

mit every unique query in our corpus as a search request to two category-based web search services: the Yahoo Directory (http://dir.yahoo.com) as well as Froogle (http://froogle.google.com). The former is a well-known manually-categorized collection of web pages, including a search service for these web pages; the latter is an online shopping search service. We use the category of the top page retrieved by the Yahoo Directory as the "Yahoo Category" for that query, and the top shopping category offered by Froogle as its "Froogle Category;" while the Yahoo Category is a topical category in the traditional sense, the Froogle Category is a consumer-related one, possibly answering the question "if there is potential commercial value in the query, what domain does it belong to?" In spirit, this is similar to the usage of the Open Directory Project to classify web pages by category (e.g., in [275]), except that we classify terms, not URLs.

The coverage achieved with this method is fairly high: in total, out of 43,601 unique queries sent to Yahoo and Froogle, 24,113 (55%) were categorized by Yahoo and 29,727 (68%) by Froogle. Some queries were not categorized due to excessive length, non-standard encodings, and other technical issues, so the coverage over common queries is even higher. Cursory examination of the resulting categories shows high accuracy, even for queries which are very hard to classify with traditional methods, using the query words only; but a full evaluation of this categorization scheme is out of the scope of this Chapter. Table 8.4 lists some examples of queries along with their corresponding categories; note that the categories are hierarchical, and the entire path from the root of the hierarchy to the category is shown.

Figure 8.4 (left) shows a breakdown of the top Yahoo categories for ad-hoc and filtering queries. Taking into account that "Regional" queries often refer to news-related events, we witness again that current events are a major source of interest for blog searchers. A similar breakdown of the top Froogle categories is given in Figure 8.4 (right), indicating that most queries which can be related to products deal with cultural artefacts, such as movies and books.

An added benefit of both the Yahoo and Froogle categories is their hierarchical nature: this enables us to not only examine the most frequent category, but also to evaluate the breakdown of subcategories within a given category. For example, Figure 8.4 shows the distribution of subcategories within the top-level "Business and Economy" Yahoo category in ad-hoc queries, and the distribution of subcategories within the top-level "Entertainment" Yahoo category in filtering queries. As is the case for general web searches, adult-oriented searches are the top category for commercial queries, followed by technology-related queries and financial interests. In the entertainment domain, music clearly dominates the scene. Overall, the categories we observe indicate that in terms of interest areas, blog search queries are somewhat more oriented towards technology and politics

Cup [140] which targeted query classification; those approaches which resemble ours have also obtained the highest scores within that evaluation framework.

Query	Yahoo Category	Froogle Category
24	Entertainment Television Shows Action and Adventure 24	Books, Music and Video Video Action and Adventure
Atkins	Business and Economy Shopping and Services Health Weight Loss Diets and Programs Low Carbohydrate Diets Atkins Nutritional Approach	Food and Gourmet Food Snack Foods
Evolution debate	Society and Culture Religion and Spirituality Science and Religion Creation vs. Evolution Intelligent Design	Books, Music and Video Books Social Sciences
Vioxx	Health Pharmacy Drugs and Medications Specific Drugs and Medications Vioxx, Rofecoxib	Health and Personal Care Over-the-Counter Medicine

Table 8.4: Example queries and categories.

than those sent to general web search engines (cf. [29]).

8.5 Session Analysis

Finally, we analyze the query *sessions* in the log, examining issues such as the amount of queries submitted in a session and the number of viewed results. We use the traditional networking definition of a session: the period of time that a unique user interacts with a server—in our case, the search engine—and the actions performed during this period.

Our log does not contain full session information: we do not know how long the user spent examining the results, and which result links she followed. However, since some identification of the user is given for each query in the log in the form of IP address and user agent string, it is possible to group the queries by sessions and to perform a basic analysis of these.

Before describing our approach to session recovery and discussing characteristics of the extracted sessions it is important to note the difference between sessions that contain ad-hoc searches and sessions that contain filtering searches. The former are similar to standard web search sessions, and consist of different queries

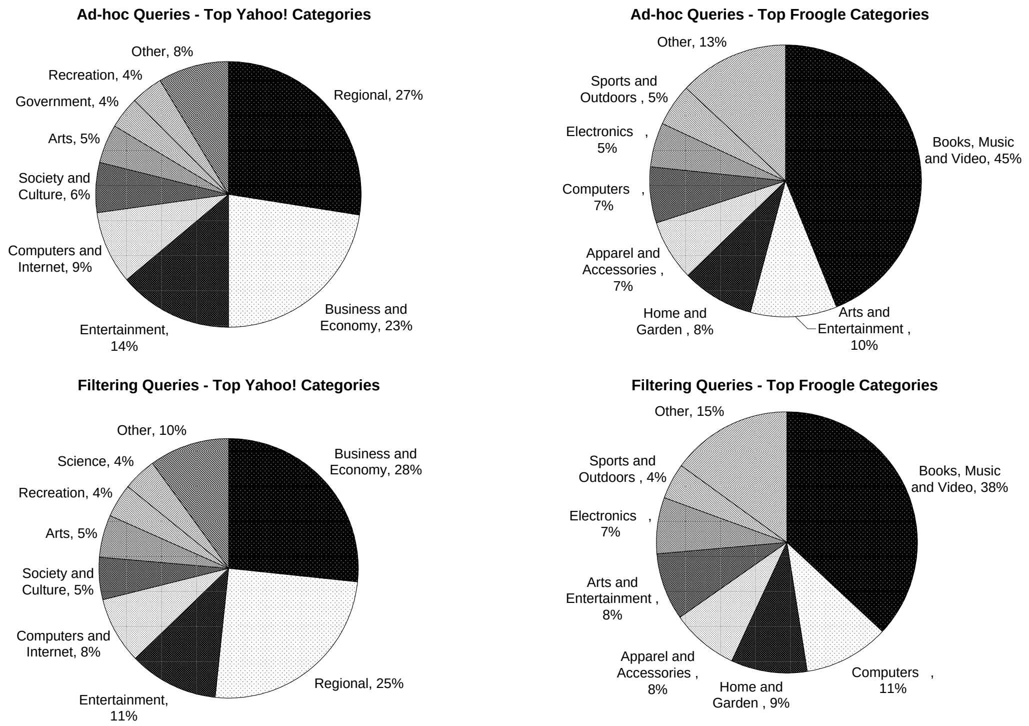


Figure 8.4: (Left): Top Yahoo categories; (Right): Top Froogle categories.

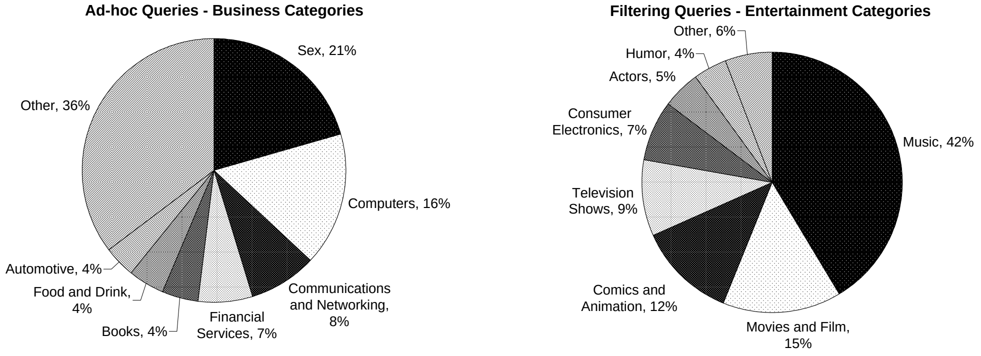


Figure 8.5: (Left): Breakdown into subcategories within the top-level "Business and Economy" Yahoo category in ad-hoc queries; (Right): breakdown into subcategories within the top-level "Entertainment" Yahoo category in filtering queries.

that a user submitted to the search engine during her visit. These different queries include, in many cases, reformulations of a query, or related terms which indicate the user is trying to collect more information regarding her interest. In contrast, "sessions" containing filtering searches are actually sets of queries registered by the same user: in practice, they are not queries submitted during a single visit to the search engine, but a list of queries the same user expressed ongoing interest in, possibly added over a long period of time.

8.5.1 Recovering Sessions and Subscription Sets

We assume two queries to belong to the same session if the following conditions hold: (1) The queries originate from the same IP address; (2) The user agent string of the two queries is identical; and (3) The elapsed time between the queries is less than k seconds, where k is a predefined parameter.

The main drawback of this method is its incompatibility with proxy servers: queries originating from the same IP address do not necessarily come from the same user—they may have been sent by different users using the same proxy server, a common scenario in certain environments, such as companies with a single internet gateway. While the usage of the user agent string reduces the chance of mistaking different users for the same one, it does not eliminate it completely. Having said that, anecdotal evidence suggests that the recovered sessions are in fact "real" sessions: the conceptual and lexical similarity between queries in the same session is high for the vast majority of sessions we examined. Additional evidence for the relative robustness of this method can be seen in the fact that, when used on the set of all queries, it produces less than 0.5% "mixed sessions"—sessions containing both ad-hoc and filtering queries, which are unlikely to be real sessions.

We performed our analyses independently for ad-hoc and filtering queries; to avoid confusion, we use the term "sessions" only for ad-hoc sessions—which are indeed sessions in the traditional sense; for filtering sessions, we use the term "subscription sets" (which denotes lists of filtering queries sent from the same user within a short timeframe).

8.5.2 Properties of Sessions

We experimented with various values of k; manual examination of the recovered sessions suggests that values between 10 and 30 seconds yield the most reliable sessions for ad-hoc queries. For filtering queries, the session time is much shorter, in-line with intuition (since the queries are automated): reliable sessions are found with k values of 2–5 seconds. The thresholds were set to 20 seconds for sessions and 5 seconds for subscription sets; this produces 148,361 sessions and 650,657 subscription sets.

Type	**Queries**
Session	autoantibodies autoantibodies histamine histamine
Session	firmware dwl 2000 ap+ dwl 2000 ap+ dwl-2000 ap+
Subscription set	"XML Tag Monitor Report" "XML Search Selector"
Subscription set	imap imap gmail Thunderbird IMAP imap labels rss email thunderbird label imap soap

Table 8.5: Example sessions and subscription sets.

		Blog queries		**Web queries**
		Sessions	**Subscriptions**	**Sessions** [278]
Size	Mean	1.45	1.53	2.02
	Stdev	0.94	2.26	123.4
	Variance	0.87	5.10	N/A
	Breakdown			
	Size 1	70.2%	75.8%	77.6%
	Size 2	20.9%	13.7%	13.5%
	Size ≥3	8.8%	10.4%	9.9%
Page views	Mean	1.09	N/A	1.39
	Stdev	0.55	N/A	2.74
	Variance	0.31	N/A	
	Breakdown			
	1 result page	94.9%	N/A	85.2%
	2 result pages	3.4%	N/A	7.5%
	3 or more pages	1.7%	N/A	7.3%

Table 8.6: (Top): Session and subscription set sizes (number of unique queries). (Bottom): Result page views for ad-hoc queries, per session.

Many sessions and subscription sets contain simple reformulations such as different uses of query operators; others are composed of related terms, and yet others consist of seemingly unrelated queries, matching different interests of the same user. Table 8.5 provides example sessions and subscription sets, and Table 8.6 (top) details statistics about the session size (the number of unique queries per session), comparing our findings to those for general web searches [278].

The short session size is similar to the one observed in web search engines, e.g., in [278]. While subscription sets also exhibit a short size on average, the actual sizes of the sets vary much more than those of sessions—as can be seen from the much higher variance. Users may subscribe to any amount of queries; in our data, some users registered as many as 20 queries.

For ad-hoc queries, an additional interesting aspect is the number of result pages the user chooses to view (each containing up to 10 matches). As with web searches, we find that the vast majority of users view only the first result page: see the detailed breakdown in Table 8.6 (bottom), again comparing our findings to those presented for general web searches in [278]. While there is a statistically significant difference between the two samples (blog sessions vs. web sessions), the bottom line is similar: most users do not look beyond the first set of results.[6]

In sum, while we found that query types in the blogspace differ from the types of queries submitted to general web search engines, we discovered a very similar user behavior regarding the number of queries issued during a single visit, and the number of result pages viewed.

8.6 Conclusions

This Chapter presented a study of a large blog search engine log, analyzing the type of queries issued by users, the user behavior in terms of amount of queries and page views, and the categories of the queries. The query log covers an entire month, and contains both ad-hoc and filtering queries.

Our main finding in terms of query types is that blog searches fall into two broad categories—conversational queries, attempting to track the references to various named entities within the blogspace, and concept queries, aimed at locating blogs and blog posts which focus on a given concept or topic. The distribution of these types differs between ad-hoc and filtering queries, with the filtering having a larger prevalence of conversational ones. In addition, we found that blog searches tend to focus on current events more than web searches. The observations regarding query types will guide us in the next Chapter, which addresses conversational queries—a query type somewhat unique to the blogspace.

Regarding user behavior, we observe similar behavior to that of general web search engines: users are typically interested only in the first few results returned, and usually issue a very small number of queries in every session. This motivates optimization of blog search for early precision, rather than mean precision.

Finally, using external resources to categorize the queries, we uncovered a blog searcher profile which is more concentrated on news (particularly politics), entertainment, and technology than the average web searcher.

[6] Also, the number of page views for web searches is constantly decreasing, as search engine technology is improving and more relevant documents appear in the first few results.

Chapter 9

Opinion Retrieval in Blogs

In the previous Chapter, we observed that many queries sent to blog search engines are named entities: names of people, locations, organizations, and so on. We hypothesized that these conversational queries stem from users' interest in the particular type of content blogs offer: thoughts and opinions of people about the entity, rather than informational, descriptive content. This calls for development of retrieval mechanisms for the blogspace which address not only the topical relevance aspect, but also take into account the degree to which a post expresses an opinion about the searched entity. This Chapter focuses on this task, *blog opinion retrieval*, systematically investigating methods to approach it.

The work in this Chapter revolves around the notion of relevance as defined for the task of retrieving opinions in blogs, and how it is addressed from different angles—recognizing opinion retrieval as a task spanning several domains. In particular, we seek answers to the following questions:

1. What are the factors indicating that a blog post is a relevant, opinionated post, given a query? What is the relative contribution of each factor?
2. How do traditional informational retrieval approaches perform on the opinion retrieval task?
3. What is the relation between the different components of opinion retrieval? Is the contribution of each factor independent from others?

The work presented in this chapter is largely based on the opinion retrieval task at TREC, which was already mentioned briefly in Section 2.3.1. We start by describing this task, introduced at TREC 2006, in more detail. Then, in Section 9.2, we describe our approach to opinion retrieval in blogs. In a nutshell, this approach combines different factors which potentially contribute to locating opinionated blog posts; we develop methods to address each of these contributing factors, and discuss their combination. Section 9.3 follows with an extensive evaluation of the actual contribution of each of these components to the success of retrieval of opinionated content, as well as the success of their combination. After describing other approaches to the opinion retrieval task at TREC and additional related work in Section 9.4, we conclude in Section 9.5.

9.1 Opinion Retrieval at TREC

TREC—the annual Text REtrieval Conference—has traditionally been the main forum for large-scale evaluation of text retrieval methodologies. It is organized around a set of separate tracks, each investigating a particular retrieval domain, and each including one or more tasks in this domain; example domains are web retrieval, enterprise retrieval, or question answering. Tasks are presented as retrieval challenges to participants, which, in turn, develop and test various retrieval approaches to address them.[1]

The 2006 edition of TREC included, for the first time, a track dedicated to blog retrieval: the TREC Blog Track [235]. In particular, the track included an opinion retrieval task, where participants were requested to locate opinionated blog posts in a large collection of posts. We now introduce this task: the collection used for it, the retrieval task, and the evaluation approach.

9.1.1 Collection

The collection used for the opinion retrieval task, the TREC Blog06 corpus [182], is a crawl of more than 100,000 syndicated feeds over a period of 11 weeks, amounting to more than 3.2 million blog posts. The collection contains, separately, the syndicated content of each post in XML format, the corresponding HTML contents, and timely snapshots of the blog home pages. The first format is useful for analyzing the blog contents as seen by feed aggregators (and most commercial blog search engines); the HTML contents include additional data such as comments and trackbacks; the blog home pages are useful for studying meta-data such as blogrolls, blogger profile links, and so on.

Descriptive statistics about the collection are shown in Table 9.1.

9.1.2 Task Description

The task presented to participants of the opinion retrieval task at TREC was "to locate blog posts that express an opinion about a given target . . . the task can be summarized as *What do people think about [the target]*" [235]. While the retrieved posts were not required to be focused on the target, an opinion about it had to be expressed in the contents of the post or in one of its comments. The underlying scenario behind this task was that of tracking online public sentiment towards entities and concepts.

[1] More information about TREC, the tracks included in it over the years, and approaches of participants to them can be found in [303] and at the TREC home page, http://trec.nist.gov.

Quantity	Value
Amount	
Unique feeds	100,649
Unique permalinks	3,215,171
Size	
Syndicated post content	38.6GB
HTML post content	88.8GB
Blog homepages	20.8GB
Total	148GB
Timespan	
First Feed Crawl	06/12/2005
Last Feed Crawl	21/02/2006

Table 9.1: Details of the TREC Blogs06 collection.

Overall, 50 topics—selected from queries sent to blog search engines during the time the collection was created—were used for the task. Topics were listed in a format similar to that used in other TREC tasks, where three fields—title, description, and narrative—describe the information need with increasing levels of detail. An example topic from the task is shown in Figure 9.1.

9.1.3 Assessment

Retrieved blog posts were assessed using a three-level scale: first, the content of the post (and any comments posted to it) was judged to be either topically relevant or not. For topically relevant posts, the post was further assessed to determine whether it contains an expression of opinion about the target, or non-opinionated content only. Finally, opinionated posts were marked by assessors as having a positive, negative, or mixed sentiment towards the topic.

More concretely, the following assessment scale was used:

0 *Not relevant.* The post and its comments do not contain any information about the topic, or refers to it only in passing.

1 *Relevant, non-opinionated.* The post or its comments contain information about the topic, but do not express an opinion towards it. To be assessed as “relevant,” the information given about the topic should be substantial enough to be included in a report compiled about this entity.

2 *Relevant, negative opinion.* The post or its comments contain an explicit expression about the topic, and the opinion expressed is explicitly negative about, or against, the topic.

3 *Relevant, mixed opinion.* Same as (2), but the expressed opinion contains both positive and negative opinions, or an ambiguous or unclear opinion.

4 *Relevant, positive opinion.* Same as (2), but the opinion expressed is explicitly positive about, or supporting, the topic.

```
<top>

  <num> Number: 886

  <title> "west wing"

  <desc> Description: Provide opinion concerning the television series
  West Wing.

  <narr> Narrative: Relevant documents should include opinion or
  reviews concerning history, actors or production information for the
  television series "West Wing".  Simple news updates or media reports
  are relevant only when they include editorial or quoted opinions.
  Articles or comments about the real-world White House west wing are
  not relevant.

</top>
```

Figure 9.1: Example opinion retrieval topic from the TREC 2006 blog track.

For the assessment used at TREC, an explicit expression was defined as a voicing of sentiment about the topic, showing a personal attitude of the writer. Examples of documents retrieved for the example query in Figure 9.1, along with their assessments according to this scale, are shown in Figure 9.2.

The total number of posts manually judged for the 50 topics was 63,103; of these, 11,530 were found to be relevant, opinionated posts—an average of 231 per topic. The per-topic distribution of relevant posts varied widely: some topics had less than 10 relevant documents, and others—more than 500; the standard deviation over all topics was 196. Each document was judged by one NIST assessor only, so no agreement information is available to estimate the stability of the judgments. Additional details about the topics, the distribution of relevant documents between them, and the assessment process, are given in the overview paper of this task [235].

The evaluation metrics used for the task were the standard ones used at TREC, i.e., mean average precision (MAP), R-precision, bpref, and precision at various cutoffs [303]. When calculating the evaluation metrics, no distinction was made between judgments (2), (3) and (4) described earlier: negative, positive and mixed-opinion posts were all judged as equally relevant for the opinion retrieval task.

9.2 A Multiple-component Strategy

The approach we take to the task of retrieving opinionated content in blogs is to identify different aspects which may indicate the presence of an expression of

Document ID: BLOG06-20051212-085-0009851294
Judgment: Topically non-relevant (0)
Excerpt:

> I seem to recall a Pegasus related story in the original series, but I can't remember how it panned out. Excellent show anyway, I can't recommend BSG more highly, even if West Wing is more your thing to sci-fi, you'll still like it.

Document ID: BLOG06-20060220-006-0000776890
Judgment: Topically relevant, no opinion expressed (1)
Excerpt:

> On the Feb. 7 flight to Los Angeles, Mr. Shelton sat next to actor Richard Schiff, who portrays Tobias Zachary on NBC's drama "West Wing." "Very nice guy . . .

Document ID: BLOG06-20060113-003-0022453465
Judgment: Topically relevant, negative opinion expressed (2)
Excerpt:

> Speaking of though, The West Wing really needs to end. This season has just NOT been that good. I've heard that this will be the last season and it will end with the announcement of the new president. So we'll see.

Document ID: BLOG06-20051211-063-0017325577
Judgment: Topically relevant, mixed opinion expressed (3)
Excerpt:

> I promised to post some thoughts on that week's episode of TWW? Well, I lied. I don't know exactly what it is about this season (because it's not really bad); but I just can't quite get excited by it - at least not enough to spend time writing any thoughts.

Document ID: BLOG06-20051212-011-0015811427
Judgment: Topically relevant, positive opinion expressed (4)
Excerpt:

> Yesterday I bought West Wing Season 6 on DVD . . . It may not be Aaron Sorkin's West Wing, but it's still one of the best dramas I've seen in a long time, some good writing and excellent storylines. I'm satisfied.

Figure 9.2: Examples of relevance judgments in the TREC 2006 opinion retrieval task. Top to bottom: non-relevant post; relevant, non-opinionated post; relevant post containing a negative opinion; relevant post containing a mixed negative-positive opinion; relevant post containing a positive opinion.

opinion in a blog post and rank the posts according to each of these separately; we then combine these partial relevance scores to a final one. This allows us to break down the opinion retrieval task to a number of simpler subproblems, which we treat as independent.

We proceed by describing the components of opinionated relevance we identify, and how posts are scored by each one of these components.

Opinion Retrieval Aspects

We group the different indicators of opinionated relevance into three high-level aspects: *topical relevance*, *opinion expression*, and *post quality.* The first aspect, topical relevance, is the degree to which the post deals with the given topic; this is similar to relevance as defined for ad-hoc retrieval tasks, such as many of the traditional TREC tasks. The second aspect, opinion expression, involves identifying whether a post contains an opinion about a topic: the degree to which it contains subjective information about it. Finally, the post quality is an estimation of the (query-independent) quality of a blog post, under the assumption that higher-quality posts are more likely to contain meaningful opinions and are preferred by users. In this last category of quality we also include detection of spam in blogs, assuming that a spam blog post is a very low-quality one.

These three high-level aspects can be modeled in various ways; indeed, we use multiple methods to estimate the relevance according to each aspect, obtaining a large number of rankings per post. We proceed by describing these multiple methods, grouped by their relevance domain; for each, we detail the retrieval status value (RSV) assigned by the method to a post. Then, we discuss how the separate retrieval status values are combined to a final, single ranked list of blog posts. A detailed evaluation of the contribution of each aspect and the success of the methods used for it follows in the next Section.

9.2.1 Topical Relevance

As stated earlier, in the opinion retrieval framework a relevant blog post does not necessarily have high topical relevance: a document is relevant if it contains an opinion about the target, even if the target is not the main topic of the document and the opinion is expressed only in passing. However, good correlation is reported between the performance of a system when measured on topical relevance only and its performance on the opinion finding task [235]. An examination of posts containing opinions about various targets shows that, indeed, in the majority of the cases, the target is also a main topic of the post, suggesting that topical relevance is an important component of opinion retrieval.

To estimate the topical relevance of a blog post given a target, we start by using standard information retrieval ranking models, focusing on those which have been reported to perform well in similar retrieval settings. In particular, we evaluated a number of probabilistic models, including the widely-used Okapi BM25 ranking scheme [254] and the language modeling approach to information retrieval as described in [115], which has also been used in earlier parts in this thesis. The retrieval status value of a blog post p given a query q according to the BM25 ranking scheme[2] is

$$\text{RSV}_{\text{topical,BM25}}(p, q) = \sum_{t \in q} w \cdot \frac{(k_1 + 1) \cdot tf_{t,p}}{K + tf_{t,p}} \cdot \frac{(k_3 + 1) \cdot tf_{t,q}}{k_3 + tf_{t,q}} \tag{9.1}$$

where $tf_{t,x}$ is the frequency of t in x, $K = k_1((1-b) + b \cdot \frac{\text{length}(p)}{\text{avelength}(P)})$; b, k_1, and k_3 are parameters of the collection; avelength(P) is the average length of the blog posts in the collection, and

$$w = \log \frac{|P| - df_t + 0.5}{df_t + 0.5}$$

where df_t is the number of documents containing the term t.

The retrieval status value according to the language modeling ranking scheme is

$$\text{RSV}_{\text{topical,LM}}(p, q) = \prod_{t \in q} (\lambda \cdot \frac{tf_{t,p}}{\text{length}(p)} + (1 - \lambda) \cdot \frac{df_t}{DF}) \ , \tag{9.2}$$

where $DF = \sum_{t' \in P} df_{t'}$, and λ is a smoothing (interpolation) parameter.

In addition to the base ranking scheme, we experiment with a number of techniques which have been reported as effective in web retrieval scenarios. First, we propagate anchor text to the content of linked blog posts; usage of anchor text in this manner has been shown to be very effective for some web retrieval tasks [58, 59]. Second, we test the effect of query expansion—the addition of new terms to a query to bridge possible vocabulary gaps between the query and relevant documents, a technique known to increase recall at the expense of early precision [21, 319]. Finally, we examine the effect of methods for using proximity information between the query terms, which have been reported as useful for web retrieval [200, 191]. More details about how these techniques were used appear in the next section, when their results are described.

Recency scoring. Until now, what we have described is a simple application of known state-of-the-art retrieval methods to blog retrieval. An examination of the performance of known methods in this new domain is important, but we would also like to explore how the properties of blogs can be used to improve over the state-of-the-art. We therefore experimented with an additional technique for improving topical relevance—one which is tailored to blogs, utilizing their temporal properties.

[2] This version of BM25 assumes no information is available regarding the number of relevant documents—as is the case in our settings.

As shown in the previous chapter, a substantial number of blog search queries are *recency queries*—queries which are related to ongoing events.[3] The distribution of dates in relevant documents for these queries is not uniform, but rather concentrated around a short period during which the event happened. For example, Figure 9.3 shows the distribution of dates in relevant documents for the query "state of the union," which seeks opinions about the presidential State of the Union address, delivered on the evening of January 31st, 2006: clearly, relevant documents are found mostly in the few days following the event. Consequently, it seems useful to assign higher relevance to blog posts which were "recent" at the time the query was issued.

Recency information has been used in the retrieval process before: Li and Croft propose to incorporate temporal information into language modeling retrieval, and use a decay function to modify retrieval scores according to their temporal distance from the event [169]. We adopt a simpler, parameter-free method which is somewhat more intuitive and does not require separate training of a decay function.

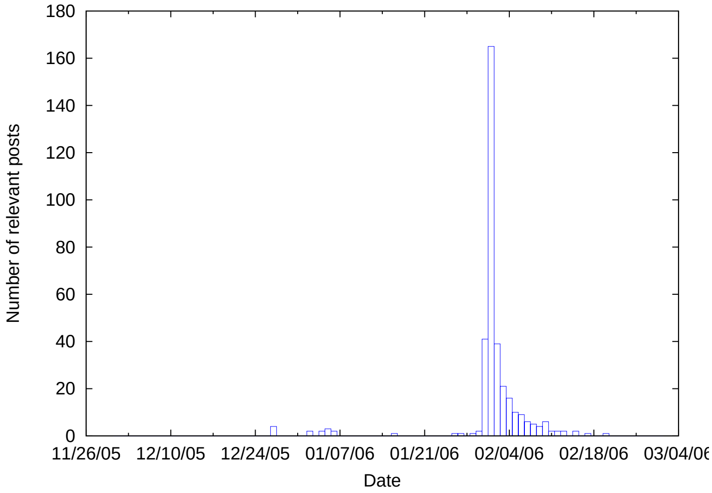


Figure 9.3: Distribution of dates of relevant posts for the query "state of the union" over time.

Our approach is simple: as with blind relevance feedback methods, we assume that highly-ranked documents (according to topical similarity) are more likely to be relevant than other documents. The distribution of dates in these highly-ranked documents serves as an estimation to the distribution of dates in relevant documents, and is treated as evidence of relevance. For example, the distribution of dates in the top-500 retrieved results for the same query as used in Figure 9.3, "state of the union," is shown in Figure 9.4. Clearly, this distribution is more

[3] The name "recency queries" is used since, assuming these queries are mostly issued near the time of the related event, they favor recently-published documents.

noisy than the distribution of the dates in relevant documents; but the peak around the time of the event is preserved. We assume, then, that blog posts published near the time of an event are more relevant to it, regardless of their topical similarity. As with content-based blind relevance feedback, this allows us to identify relevance also for documents which do not contain the query words.

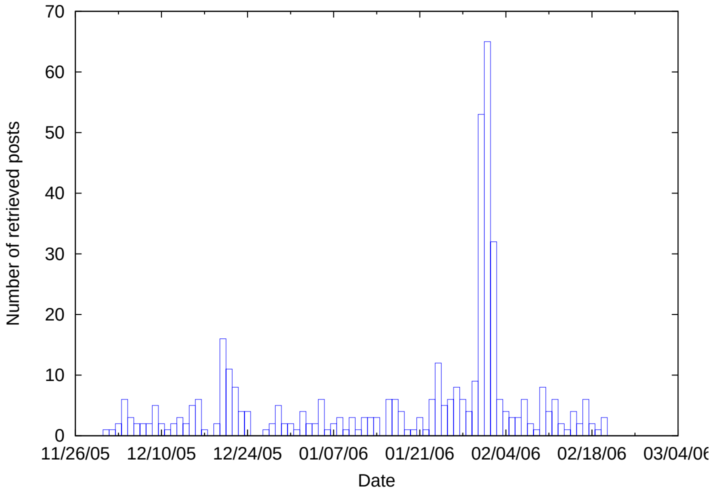


Figure 9.4: Distribution of the dates of the top-500 retrieved posts for the query "state of the union" over time.

More formally, the retrieval status value of a blog post p given a query q according to this approach is

$$\text{RSV}_{\text{recency}}(p, q) = \frac{1}{k} \cdot \text{count}(\text{date}(p), k), \tag{9.3}$$

where k is the number of top results used for the temporal feedback process, $\text{date}(p)$ is a function returning the publication date of a post p, and $\text{count}(d, k)$ is the number of posts p for which $\text{date}(p) = d$ within the top k results retrieved according to a given ranking model (in our case—a topical one, such as those appearing in formulas (9.1) or (9.2)).

Details about the success of this recency scoring approach—as well as the other techniques we used for topical relevance estimation in the opinion retrieval task—appear in Section 9.3.

9.2.2 Opinion Expression

We now turn to the second aspect of opinion retrieval we explore, namely, identification of opinionated content within retrieved posts. Clearly, of the three aspects we discuss, this is the one most strongly related to the opinion retrieval task, and the one setting it apart from other retrieval settings.

We treat the task of locating opinions in text as similar to the task of sentiment classification—identification of positive and negative opinions towards a topic. Broadly speaking, there are two main approaches to sentiment classification: lexicon-based methods, and machine learning approaches. Lexical methods first construct a dictionary of terms indicating sentiment (often, lists of "positive" and "negative" words); sometimes, a weight is associated with each word, rather than a binary indication. The sentiment of a given text is then derived by the occurrence of words from this dictionary in the text, e.g., by summing their weights or combining them otherwise. In some cases, additional heuristics are used, such as handling negation in a sentence by reversing the weight of words appearing in it [222], or taking into account the proximity between words and the topic for which the sentiment is classified [292]. As for constructing the sentiment lexicon, various approaches have been described, including manually [287], by using surface patterns [253], via WordNet [144], and with co-occurrence statistics [298].

The second approach to sentiment classification, the machine learning one, views it as a text classification task. In this approach, a classifier is trained on a set of texts annotated for sentiment; typical features used for the learning process are word n-grams and text length [238, 65]. Often, some linguistic information is embedded in the learning process by using, among others, part-of-speech tags, stemming, semantic orientation values, and additional language-oriented features [212, 65].

We employed both a lexicon-based approach and a text classification one to identify opinionated content in retrieved blog posts; additionally, we experimented with techniques tailored to blog data. We follow by describing the methods we used.

Lexicon-based. Much of the lexicon-based work on sentiment analysis is centered on creating the sentiment-tagged dictionary. In our settings, however, the focus is on measuring the effectiveness of applying a sentiment lexicon, rather than evaluating the success of different approaches to creating such a dictionary. For this reason, we choose to use a manual list of words rather than test various automatically-generated ones. The lexicon we use is the General Inquirer [287], a large-scale, manually-constructed lexicon which is frequently used for sentiment classification tasks (e.g., [323, 141]). The General Inquirer assigns a wide range of categories to more than 10,000 English words; among the categories assigned are Osgood's dimensions of the semantic space (Positive/Negative; Active/Passive; Strong/Weak [234]) and a number of emotional categories. Of this set of categories, we use a subset which we view as related to expression of opinion to construct a sentiment lexicon. Table 9.2 lists the General Inquirer categories we use as indicators for the presence of opinionated content in the text, providing a description and examples for each.[4]

[4] A full description of the categories of the General Inquirer is given in [287].

Category	**Examples**	**Description**
Valence categories		Based on Osgood's semantic axes
Positive	decent, funny, hope	
Negative	awkward, fussy, wrong	
Emotion categories		
Arousal	hate, sympathize	Words indicating excitement
Emotion	shy, wonder	Emotion-bearing words
Feel	cranky, gratitude	Words describing feeling
Pain	concern, scared	Words indicating suffering
Pleasure	comical, love	Words of joy and confidence
Virtue	handy, popular	Words indicating approval
Pronoun categories		
Self	I, myself	Standard personal pronouns
Our	our, us	
You	you, your	
Adjective categories		
Independent	amazing, offensive	Standard adjectives
Relational	blunt, rude	Relations between people
Misc. categories		
Respect	criticism, famous	Gaining or losing respect

Table 9.2: General Inquirer categories used as opinion indicators.

Given the wide range of categories involved, there are many ways to combine the counts of words belonging to the various categories in a given text. However, such combinations, or usage of weighted lists rather than a binary indication of whether a word belongs to a category, have shown little or no improvement over using raw General Inquirer data [215]. Consequently, we follow a straightforward approach: we simply combine the words belonging to any one of these categories to a single "opinionated term" list, ignoring whether words were originally marked as positive, negative, or any other specific category. To this end, we assume that any of these words indicate the existence of an opinion expression to some extent. The total number of opinion-bearing words in our list is 3,917.

After creating this list of opinionated terms, we calculate two sentiment-related values for each retrieved post: a "post opinion level" and a "feed opinion level." In both cases, the opinion level is the total number of occurrences of words from our list in the text, normalized by the total number of words in the text; the difference between the two values is the text used for counting the occurrences. As the name indicates, the post opinion level is measured from the text of the blog post itself; more concretely, we experimented with analyzing two types of post texts. The first is the entire text of the post, and the second is all "topical sentences" from the post. Topical sentences, in our approach, are all sentences containing the topic verbatim, as well as the sentences immediately surrounding them: this focuses the search for opinion-bearing words to parts of the post which are likely to refer directly to the topic, rather than the post in its entirety. Similar approaches to limiting sentiment analysis to relevant sentences have shown to improve results [237].

For the second value, the feed opinion level, we use the text of the entire feed to which the post belongs (i.e., the text appearing in the blog during the entire 11-week period); this is a static, topic-independent score per feed, estimating the degree to which it contains opinions (about any topic). The intuition here is that feeds containing a fair amount of opinions are more likely to express an opinion in any of their given posts. There is also a practical benefit to analysis of the entire feed: since the amount of text in a feed is typically substantially larger than that of a single post, and since lexical methods such as the one we use work better on longer texts (cf. [299]), the feed-based measurement is more robust.

We use the post opinion level and the feed opinion level as retrieval status values, and rank all posts according to them. Formally, the RSVs assigned by the lexical-based sentiment module to a blog post p are

$$\mathrm{RSV}_{\mathrm{opinion(entire\ post)}}(p,q) = \frac{\mathrm{opin\_count}(p)}{|p|} \tag{9.4}$$

and

$$\mathrm{RSV}_{\mathrm{opinion(topical\ sentences)}}(p,q) = \frac{\mathrm{opin\_count}(\mathrm{topical\_sentences}(p,q))}{|\mathrm{topical\_sentences}(p,q)|}\ , \tag{9.5}$$

where opin_count(s) returns the number of words in the string s which are included in the opinionated term list, and topical_sentences(p,q) returns the concatenated text of all sentences in p containing q as well as the sentences surrounding them, as described earlier. Similarly, the retrieval status value assigned to p based on its feed, P, is

$$\mathrm{RSV}_{\mathrm{opinion(feed)}}(p) = \frac{\mathrm{opin\_count}(P)}{|P|} \tag{9.6}$$

Text-classification based. As in earlier chapters, we utilize support vector machines for our text classification approach; SVMs have also been applied extensively to sentiment classification tasks in the past (e.g, [212, 65]).

We experimented with three types of training sets to train the classifier. The first is a set of 5,000 subjective and 5,000 objective sentences in the domain of movie review which was used in [237] for sentiment analysis in this domain. The second training set is a collection of posts from the Blog06 corpus consisting, as positive examples, of posts tagged with tags likely to indicate opinionated content (such as “rant” and “review”) and, as negative examples, of posts tagged with tags less likely to indicate opinions (such as “general”). The final training set utilized the TREC assessments: for each individual TREC topic, we used the posts judged as relevant, opinionated documents for all other 49 topics as positive examples, and the posts judged as relevant, unopinionated documents for the 49 topics as negative examples. A separate SVM was trained for each topic and used to predict opinionated content in posts retrieved for that topic.

The features we use for the classification are word unigrams only; additional features, such as longer n-grams or part-of-speech tags show limited or no improvements over a unigram approach in the setting of sentiment classification [238, 65], unless a very large amount of training data is available [61]. To reduce noise in the corpus, we used only the 20,000 most frequent terms in the collection as features.

For any of the training sets, the retrieval status value assigned by this component to a post p is simply the classifier's estimate of the probability of p belonging to the class of opinionated posts:

$$\mathrm{RSV}_{\mathrm{opinion(classifier)}}(p) = P_{\mathrm{classifier}}(\mathrm{class}(p) = \mathrm{opinionated}) \tag{9.7}$$

Opinion level through post structure. Aside from approaching the task of locating opinionated content as a sentiment classification task, we experimented with an additional method, utilizing the structure of a blog post rather than its content. In Chapter 7, when discussing blog comments, we showed that highly-commented posts tend to contain disputes and discussions. We hypothesize that the amount of comments is also related to the degree to which an opinion is expressed in the text, assuming that opinionated posts are more likely to attract comments than non-opinionated ones. This is somewhat similar to assumptions made by some of the approaches used at the discussion search task at the Enterprise track at TREC, where the length of the discussion was used as an indication of the presence of subjective content in it [57].

In Chapter 7 we described a complex model-driven approach to extracting comment data. For the purpose of retrieval, however, we are interested only in the amount of comments, or the amount of information added by comments to the original posts. To identify this, we opt for a simpler approach.

Recall that the Blog06 collection contains, separately, both the HTML and the syndicated content of each post. The difference between the length of the content in HTML format and the length of the syndicated content in XML format is the total amount of overhead found in the HTML version. This overhead contains layout information, as well as content which is not directly related to the post: archive links, profile information, and so on. One of the types of content present in HTML but not in the syndication is the comments and trackbacks of the post. Generally, the amount of non-comment overhead involved is fixed over multiple posts in the same blog: the same archive links, blogger profile and so on appear in every permalink, and remain constant; the only component that changes substantially is the comment data. To exploit this, we first compute the average HTML to XML ratio for a feed, over all posts that are part of it. Given a post, a comparison of its specific HTML to XML ratio to the average one gives us an indication of the amount of post-specific overhead associated with this post—which we view as reflecting the relative volume of comments of the post. This ratio, which we refer to as the "discussion level" of a post, is about 1 for posts which have the same amount of comments as other posts in the same blog, lower than 1 for posts which attract less discussion, and higher than 1 for posts with an above-average volume of responses. Note that in this way, the discussion level is normalized within a feed.

More formaly, let p_{XML} be the syndicated content of a post p from the feed P, and p_{HTML} the corresponding HTML content. Then the discussion level for a given post p, which also serves as the retrieval status value assigned by this component, is

$$\mathrm{RSV}_{\mathrm{opinion(structure)}}(p) = \frac{\frac{|p_{\mathrm{HTML}}|}{|p_{\mathrm{XML}}|}}{\frac{1}{|P|}\sum_{p' \in P} \frac{|p'_{\mathrm{HTML}}|}{|p'_{\mathrm{XML}}|}} \tag{9.8}$$

Computation of this value requires only knowledge of the length of the HTML and XML of p, and not full extraction of comments.

Table 9.3 compares the discussion level calculated this way with the manually extracted number of comments of a few posts in two different blogs, showing the correlation between the two—and, also, demonstrating the difference between simply extracting the number of comments, and estimating the comment volume through the HTML overhead: a post from a feed which has a low average number of comments may be assigned a higher level of discussion even when it has fewer comments than a post from a blog which is regularly commented.

Post ID	Feed ID	Disc. level	Comments
BLOG06-20051221-003-0015014854	feed-006792	0.6	0
BLOG06-20051207-022-0022756510	feed-006792	1.5	1
BLOG06-20051207-022-0023018509	feed-006792	2.1	2
BLOG06-20051207-029-0008800181	feed-007168	0.8	3
BLOG06-20060125-004-0015081881	feed-007168	1.0	7
BLOG06-20060215-005-0024698640	feed-007168	2.1	16

Table 9.3: Examples of discussion level values.

9.2.3 Post Quality

Finally, the third set of rankings we use to determine relevance for the opinion retrieval task concerns the "quality" of a blog post. In particular, we are interested in filtering spam blogs, and in incorporating the degree of authority assigned to a post (and its blog) into the final retrieval score. The benefit of spam filtering is clear, as a spam post is substantially less likely to constitute a relevant document. The benefit of using authority information remains to be proved: while posts containing opinions do not necessarily have high authority, we hypothesize that, given two posts with similar opinions, a searcher will prefer the one with higher authority.

To measure the quality of a blog post, then, we estimate separate spam and authority scores, both of which are query-independent, and incorporate them into the ranking. We follow with details about the approaches used for both estimations.

Link-based authority. Estimating the authority of documents in a hyperlinked environment using analysis of the link structure (e.g., PageRank, HITS) is known to be an effective approach for certain web retrieval tasks [49]. We follow Upstill et al. [300], who show that the inbound link degree is a good approximation of more complex approaches such as PageRank. To this end, we use both the inbound link degree of a post (discarding links from other posts which belong to the same blog) and the inbound link degree of the blog to which the post belongs (again, discarding intra-feed links) as a crude estimation of the link-based authority of a post; these are used as the retrieval status values relating to post authority, e.g.,

$$\mathrm{RSV}_{\mathrm{authority}}(p) = \log \mathit{indegree}(p) \tag{9.9}$$

Spam likelihood. Spam is a feature of complete blogs, rather than of individual posts; we therefore estimate a "spam likelihood" score for each feed, and propagate it to any post belonging to it. To estimate this score we build on existing work on identifying spam web pages [228], and, particularly, on identifying blog spam [149]. More concretely, we test two approaches which have been successful in these domains: a machine learning approach, and measures of compressibility.

For the machine-learning approach, we follow work which has been shown to be effective for spam detection in this domain [149]. We created a training set of spam and non-spam feeds from the Blog06 collection using two naive assumptions. The first is that any feed from the domain `blogspot.com`, and with a domain name exceeding 35 characters, is a spam blog. Sample feeds judged as spam using this rule are `weightloss7666resources.blogspot.com` or `casino-hotel-in-windsor-poker.blogspot.com`. The second naive assumption is that a feed from the domains `livejournal.com` or `typepad.com` is not spam. These assumptions are based on properties of blog spam which were true at the time the Blogs06 corpus was created: the popularity of Blogspot among spammers due to easy automation of posting, and a relatively low level of spam in TypePad (a platform requiring payment) and LiveJournal. While both assumptions are crude, we found that they achieve very high precision (at the expense of low recall): an examination of 50 random feeds which met the "spam assumption" and 50 random feeds which met the "non-spam" one revealed no incorrect classifications. Clearly, this training set is biased, but will still give us an indication as to the importance of spam detection for retrieval purposes. Our training set was created, then, by collecting 500 random feeds classified by the two assumptions as spam, and 500 feeds classified as non-spam. We then trained an SVM on this set; again, we use unigram features only, as additional features have been shown to contribute little to success for this task [149]. The predictions of the classifier for the entire feed collection are used as evidence for the likelihood of a given feed to be spam.

Our second spam detection method follows one of the techniques used by Ntoulas et al. [228] for spam classification on the web, namely, text-level compressibility. Many spam blogs use keyword stuffing—a high concentration of certain words, aimed at obtaining high relevance scores from search engines for these keywords. Keywords and phrases are often repeated dozens and hundreds of times in the same blog "post," and across posts in the spammy feed; this results in very high compression ratios for these feeds, much higher than those obtained on non-spam feeds. Using the same collection as used for training the SVM, we calculated the distribution of compression ratios of both spam and non-spam feeds; all feeds in the corpus were then assigned a likelihood of being drawn from each of these distributions. The text used for measuring compressibility was the syndicated content of the field, rather than the full HTML content; this was done to reduce the effect of repeated HTML text (such as templates repeating in every post of the same blog) on compression ratios.

In both cases, the retrieval status value assigned by the spam component to a post p is the classifier's estimate of the probability that p belongs to the non-spam class:

$$\text{RSV}_{\text{nonspam}}(p) = P_{\text{classifier}}(\text{class}(p) = \text{nonspam}) \tag{9.10}$$

9.2.4 Model Combination

We described three different aspects we consider as potentially useful for identifying relevant, opinionated blog posts; for each, we illustrated several methods for ranking the posts, creating the multiple rankings for each post shown in formulas (9.1) to (9.10). A user, however, is interested in a single ranked list; the different scores we estimate need to be combined into a single ranked result list.

Probabilistic ranking models such as the language modeling approach often use *mixture models* [163] to combine different distributions in the ranking process. In this setting, the combination is performed at the level of the ranking formula, assuming that the different components to combine can all be expressed in the atomic probabilities used in the ranking process, such as the likelihood of observing a word in a document. In our framework, however, what is combined is multiple evidence of relevance at the level of the entire post—final retrieval scores assigned by different approaches. Combining scores assigned by different retrieval methods is a common task in IR (cf. [164, 60]), particularly in web domains which are rich in different ways of ranking the documents (e.g., using the document text, authority scores, recency, and so on).

We adopt a standard method used in this scenario: computing the final retrieval score as a linear combination of the various partial scores [60]. In this approach, a weight is associated with each ranked list, and used to multiply the score assigned to a document by that list. The retrieval status value of a document—a blog post p, in our case—given a query q becomes then

$$\mathrm{RSV}(p,q) = \sum_i \lambda_i \cdot \mathrm{RSV}_i(p,q) \ , \tag{9.11}$$

where RSV_i are the different retrieval status values returned by the different components—the scores assigned to the blog post in different ranked lists, as shown in formulas (9.1) to (9.10)—and λ_i is the weight assigned to component i.

Optimizing the combination weights. The weight of each list, λ_i, was tuned independently of the other lists, by optimizing its combination with a baseline model assigned with a fixed $\lambda = 1$.[5] Optimization can be done separately for early and average precision; to obtain an optimal weight for early precision, we seek the maximum P@10 value of the combination of model i with the baseline:

$$\lambda_{i,\text{early\_precision}} = \underset{\lambda}{\operatorname{argmax}} \ \mathrm{P@10}(\lambda \cdot \mathrm{RSV}_i(p,q) + \mathrm{RSV}_{\text{baseline}}(p,q)).$$

Similarly, we optimize for average precision by seeking the maximal MAP value of the combination:

$$\lambda_{i,\text{average\_precision}} = \underset{\lambda}{\operatorname{argmax}} \ \mathrm{MAP}(\lambda \cdot \mathrm{RSV}_i(p,q) + \mathrm{RSV}_{\text{baseline}}(p,q)).$$

Optimally, a separate training set would be used to optimize λ_i (whether for early or average precision). However, since relevance assessments are available for 50 topics only, using a separate training set is impractical; instead, we use a leave-one-out approach, as we did when training the opinion classifier. In this approach, weights are optimized for each query separately, by using the remaining 49 queries as a training set.[6]

9.3 Evaluation

Returning to the research questions introduced earlier in this Chapter, we set out to identify what factors affect the performance of opinion retrieval, what

[5] Note that we are not combining probabilities, so the sum of λ_i is not required to be 1. The weight of the baseline was set to 1 so that the different λ_i will be comparable when combined together.

[6] For a given model i, these query-dependent weights were equal up to two digits after the decimal point; since we round weights before calculating the linear combination, in practice the same weight was used for all queries.

the contribution of each is, and whether they are mutually dependent. In the previous Section, we described a wide range of components, grouped into three high-level aspects, which we viewed as potentially contributing to the accuracy of identifying opinionated blogs posts. In this Section, we test the performance of these components, answering the research questions raised in the beginning of this Chapter. In particular, we are interested in identifying the factors which have the highest impact, and those which do not seem to improve performance at all. In several cases, we also evaluate the performance of the components on different subsets of the topics.

We proceed by examining the usefulness of the techniques proposed in the previous section, individually testing their effect on retrieval results. In addition, we examine the effect of varying different parameters of the retrieval environment, some of which are blog-specific (e.g., usage of the syndicated text only, compared to usage of the entire HTML of the post), and some of which are not particular to blogs; this addresses another one of the questions we raised, about the success of traditional retrieval approaches in this domain. Finally, we wish to examine whether combining multiple factors results in improvements beyond those obtained by the separate components, testing their independence.

All evaluations are based on the 50 topics of the TREC 2006 opinion retrieval task and the assessments provided for them at TREC.

Retrieval Settings

The first set of parameters we examine constitutes the basic building blocks of the retrieval environment: the ranking scheme used, the types of content indexed, and the type of queries used.

9.3.1 Base Ranking Model

To select the retrieval model used in the rest of the experiments, we tested the performance of a number of widely used ranking formulas. For this evaluation we used an index of the HTML contents of the posts (rather than the syndicated content only) and the "title" field of the topics; additional experiments with these parameters are described in the next sections, after fixing the base retrieval model used. Standard tokenization and English Porter stemming were used in all cases. Table 9.4 compares the evaluation of the top-1,000 ranked results obtained using the following ranking schemes:

- **Lucene.** The default ranking scheme of Lucene, a popular open-source retrieval system, which uses a somewhat outdated tf·idf variant; details about Lucene and its ranking scheme are found in [106].
- **Okapi BM25.** A popular probabilistic ranking formula, described in [254]; it is often used as a baseline for retrieval performance evaluation.

- **Divergence from Randomness.** Also a probabilistic approach, divergence from randomness has shown to improve over BM25 in some cases [9]; we use the $I(n_e)L2$ variant, which has been reported to have the highest performance of this family of ranking schemes—particularly for early precision, but also for other measures [9].
- **Language Modeling.** Another popular probabilistic retrieval approach, which we have already used in a number of places throughout this thesis (e.g., in Section 3.2). In particular, we use the language modeling variant proposed by Hiemstra in [115], which has shown to achieve same-or-better results as top ranking formulas.

Retrieval Model	MAP	R-Prec	bpref	P@10
Lucene (tf·idf)	0.1089	0.1830	0.1901	0.1920
Okapi BM25	0.1724	0.2465	0.2474	0.3500
Divergence from Randomness	0.1748	**0.2508**	0.2500	0.3500
Language Modeling	**0.1797**	0.2452	**0.2564**	**0.3560**

Table 9.4: Comparison of base ranking schemes.

Although performance of the three state-of-the-art ranking formulas is similar, the language modeling approach scores marginally better on most metrics; it was selected as the base ranking scheme for the rest of the experiments in this chapter. Worth noting is the relative poor performance of out-of-the-box Lucene, which is widely used in various search platforms (e.g., Wikipedia, CNET); one possible reason for this is the bias of the particular variant of tf·idf it uses towards shorter documents, which are not necessarily those blog posts containing opinionated content.

9.3.2 Full-content vs. Feed-only

As described in Chapter 2, most blogs provide a machine readable format of their posts via the syndication mechanism: an XML formatted version of the content of the post, along with meta-information about the post.

Indexing and searching the syndicated content only has some practical benefits for blog search engines. Syndicated RSS or Atom feeds are meant to be read by machines rather than humans, and are therefore standardized: post meta-data such as the author's signature, the timestamp, and even the URL of the blog the post is part of, is provided in known XML entities and easy to extract. On top of this, syndicated content is often "cleaner" than the matching HTML content, as it usually contains only the text related to the post (and not, for example, links to archives, blogger profile, advertisements, and so on); typically, removing this noise from web data is a laborious task. Finally, the lack of this extra content, as well as layout code related to rendering the HTML, results in feeds being substantially more compact than their counterpart HTML content: going back to the description of the Blog06 corpus in Table 9.1, we note that the size of the syndicated part of the corpus is less than half of that of the HTML part. This results in reduced bandwidth and storage requirements for search engines focusing on syndicated content only; indeed, some commercial blog search engines (e.g., Google Blog Search, Feedster) are syndication-based.

Usage of syndicated content only does have some drawbacks. The main disadvantage of relying only on syndication is that some data is not syndicated; we have already mentioned in Chapter 7 that the majority of blog comments are not distributed through RSS or Atom feeds, and the same holds for trackbacks. On top of this, some blogs syndicate only a summary of the entire post, leaving most of it in the HTML version of the blog (according to [87], the percentage of syndicated content which is partial is 11%). Additionally, some blogging platforms use non-standard syndication, or syndicate erroneous XML (an RSS/Atom-compliant parser failed to parse about 2% of the feeds in the Blog06 corpus), and other platforms (again, 11% according to [87]) do not syndicate at all. In total, 10%–20% of the full contents found in the HTML version are missing from the syndicated content.

Our next experiment is aimed at comparing the retrieval accuracy obtained when using the full HTML of the blog posts with the accuracy obtained using syndicated content only. We created two indexes: one using the HTML content of the permalinks, and one using the feeds only; again, we used title-only queries to evaluate retrieval from both indexes, using in both cases the language modeling framework. Table 9.5 compares the performance obtained with the two indexes.

Content Source	**MAP**	**R-Prec**	**P@10**
Syndicated RSS/Atom feeds	0.1465	0.2360	0.3480
Full HTML of permalinks	0.1797 (+23%)	0.2420 (+3%)	0.3560 (+2%)

Table 9.5: Retrieval from syndicated content compared with retrieval from HTML content.

Performance increases substantially when using the entire HTML contents; the increase is statistically significant. Note that the TREC Blog06 collection contains only syndicated blogs, as it was created using a seed set of feeds rather than HTML pages. Retrieval from an index of the entire blogspace—in which, as noted earlier, about 11% of blogs are non-syndicated—is likely to have yet larger differences in performance.

One interesting result of this comparison is that early precision scores are similar for both syndicated content retrieval and full HTML retrieval. This indicates that in an interactive search session, where a user is typically viewing only the first results, the differences in the percentage of relevant documents out of the top ranked results are small. A closer examination reveals that for queries with few relevant documents, early precision of syndicated content even exceeds that of HTML content. Table 9.6 shows the same comparison as that shown in Table 9.5, this time separately for the 10 topics with the largest amount of relevant posts and the 10 topics with the lowest amount of relevant posts (the average number in the first set is 38, and in the second 542; the average over all 50 topics is 231). One possible reason for the improved early precision is that syndicated content is clean of typical noise found in HTML content, and while an RSS-based index contains less information (decreasing recall and average precision), it is of high quality (increasing early precision).

Content Source	**MAP**	**R-Prec**	**P@10**
Top-10 topics, ranked by number of relevant documents			
Syndicated RSS/Atom feeds	0.2393	0.3697	0.6200
Full HTML of permalinks	0.3595 (+50%)	0.4367 (+18%)	0.6900 (+11%)
Bottom-10 topics, ranked by number of relevant documents			
Syndicated RSS/Atom feeds	0.0929	0.1394	0.2400
Full HTML of permalinks	0.0820 (−12%)	0.1158 (−17%)	0.1100 (−54%)

Table 9.6: Syndicated content compared with HTML content, different query types.

In scenarios where recall is important (such as tools for market analysis based on automated analysis of a large number of blog posts), the difference between using syndicated content and full HTML is much more apparent: for the entire 50-topic set, 6,628 relevant documents were retrieved from the HTML content (out of 11,530 documents judged relevant), compared to 5,649 retrieved when using syndicated content, a 17% change. Interestingly, the latter set of documents is not a subset of the former: 846 of the relevant documents were retrieved only from the syndicated content, and 2,557 only using HTML content. This, naturally, raises the possibility of merging the results of the two approaches; the performance obtained through this is shown in Table 9.7, showing that the improvements gained are substantial.

Content Source	**MAP**	**R-Prec**	**P@10**
HTML only	0.1797	0.2452	0.3560
Combined syndicated and HTML content			
Optimized for early precision	0.1905 (+6%)	0.2594 (+6%)	0.4020 (+13%)
Optimized for average precision	0.1978 (+11%)	0.2638 (+8%)	0.3820 (+7%)

Table 9.7: Combining syndicated content with HTML content.

As an aside, we also examine the amount of resources saved when using syndicated content only. As noted earlier, Table 9.1 shows that the raw size of the syndicated content is 43% of the size of the HTML content; this directly translates to a 57%

reduction in terms of bandwidth usage and required space for storing unprocessed content when using a syndicated content only system. Other affected resources are index size, index creation time, and retrieval time; the reduction of each appears in Table 9.8.[7] Note the large reduction in index creation time; this is both because extracting the content from an XML document is less computationally demanding than extracting content from HTML, and because indexing is an I/O-intensive process—decreasing the number of disk accesses results in substantial improvements.

Resource	**Reduction**
Bandwidth	57%
Storage of raw content	57%
Index size	67%
Index creation time	91%
Retrieval time	34%

Table 9.8: Resource requirement relaxation when using syndicated content only, compared to full HTML content.

All in all, usage of HTML content—despite the noise added by the HTML markup and templates—results in substantially better retrieval effectiveness than usage of syndicated content only. The main reasons for this are content found in comments and trackbacks, and the fact that many blogs syndicate only partial content. Some commercial blog search engines rely on the syndicated content for retrieval; our results suggest that while scanning the syndication for extraction of meta-data such as time-stamp and author name is useful, retrieval itself should utilize the full HTML content.

Anchor Text

As noted earlier, an additional source of content we experimented with was anchor text. In total, 2.3 million posts from the Blog06 collection (73%) were linked from other posts; the anchor text of these totaled slightly more than 1 GB of text. We experimented with adding the anchor text to the linked pages, but this resulted in marginal improvements to retrieval performance only (less than 0.5%); statistical significance for these was not established.

An examination of the anchors in the collection reveals why their usage fails to improve retrieval in this setting. In more than 93% of the 2.3 million posts which had propagated anchors, the anchors came only from intra-blog links—links between permalink pages from the same blog. The vast majority of these

[7] We report only percentages saved and not actual numbers, as these vary across retrieval systems and with the hardware used for the process; for these percentages, the same retrieval platform and the same hardware was used. Retrieval time was the aggregated time needed for all 50 TREC topics.

internal links are automatically generated by the blogging platform: “next” and “previous” post links, archive links, and so on. The anchors for these are created by the blogging platform, and are often repetitions of the title of the blog post, or its publication date. As this information is already present in the linked post, propagating it adds little in terms of retrieval. Examining those 167,000 blog posts which did have anchors originating from inter-blog links, we witnessed that the vast majority of anchor text consists of the name of the linked blog or its author—information useful for named-paged finding (cf. [58]), but not for opinion retrieval.

9.3.3 Query Source

Next, we explore the effect of increasing the context provided for queries on the retrieval performance. As in many other TREC tasks, each topic used at the opinion retrieval task contains three distinct fields: title, description, and narrative (see Figure 9.1). The fields provide an increasingly detailed description of the information need, where the title fields provides a summary of the need (and, in the case of the opinion retrieval task, is taken verbatim from a real user query found in a search log); the description is a one or two sentence long account of the required information; and the narrative provides yet more details about the requested documents. Table 9.9 compares the performance of using the different fields or combinations of them as queries.

Query Source	MAP	R-Prec	P@10
title	0.1797	0.2452	0.3560
description	0.1802 (+0%)	0.2578 (+5%)	**0.3900 (+10%)**
title, description	**0.2000 (+11%)**	**0.2713 (+11%)**	0.3880 (+9%)
title, description, narrative	0.1801 (+0%)	0.2585 (+5%)	0.3780 (+6%)

Table 9.9: Comparison of query sources.

Clearly, addition of some context to the query results in noticeable improvements, both for early and overall precision. Despite this, in the rest of the experiments described in this Chapter, we use title queries only. As shown in the previous Chapter, short queries, containing only an entity name, make the bulk of queries to blog search engines, and most effort should therefore go into improving the results obtained for them. Advanced searchers, such as marketing professionals interested in public opinion, would nonetheless benefit from issuing longer, more explicit queries, such as those created by combining the title and description fields of TREC topics.

Consequently, the title-only run shown in Table 9.9 serves as our baseline for the rest of the evaluations described, and as the component against which the weights of other components are optimized as detailed in the previous Section (the

$\text{RSV}_{\text{baseline}}$). This is a robust baseline: the median MAP score at TREC 2006 was 0.1371, and the top-performing system achieved a MAP of 0.2052 using many opinionated-content related heuristics; our baseline would have been ranked, as-is, 9th out of 57 participating runs at TREC 2006. To measure whether improvements we obtain over this baseline are statistically significanct, we use the t-test, which has been shown to produce the lowest error rates for significance testing of retrieval experiments [262]. From now on, all tables summarizing experimental results will contain a last column indicating the statistical significance of the change in performance from this baseline (note that with 50 topics, an improvement of less than 5% is marginal, even if statistical significance is established using the t-test).

Topical Relevance

After experimenting with the basics of the retrieval model and determining a baseline, we continue by evaluating the components described in the previous section, starting with the first aspect we identify as part of opinion retrieval, namely, improving the topical relevance of the retrieval. Three components are examined separately: query expansion techniques, term dependence models, and recency information.

9.3.4 Query Expansion

We evaluated two query expansion methods. First, we manually added terms to the query; terms were selected by researching the topic and determining words that seemed important or disambiguating in that context. Second, we used Ponte's language-modeling blind relevance feedback framework as proposed in [242]; essentially, the terms added by this method are the most discriminating terms found in the top retrieved results, when compared to the rest of the corpus. In this approach, the top retrieved results (we used the top 20 documents) are assumed to be relevant.

As noted in the previous section, query expansion tends to improve recall at the cost of reduced precision. We limited the number of manually added terms to 3 to prevent excessive topic drift; as for the number of terms added by blind relevance feedback, we experimented with various values between 3 and 40.

Examples of terms added to topics from the test set are shown in Table 9.10; the effect of the addition of terms is listed in Table 9.11. Examining the results, we see that with a small number of terms, manual expansion is favorable to blind relevance feedback; however, since blind relevance feedback allows for inexpensive expansion with many terms, it can outperform manual approaches, at least on average precision, when adding a fairly large number of terms. However, the overall performance gain from query expansion seems not quite as high as that achieved in other domains, (cf. [181]).

Topic	Blind Relevance Feedback	Manual
859. letting india into the club?	nuclear	Friedman
	times	NYT
	friedman	N.P.T
867. cheney hunting	vice	Whittington
	dick	accident
	accident	shooting
896. global warming	climate	temperature
	change	atmosphere
	greenhouse	greenhouse

Table 9.10: Examples of terms added with relevance feedback and manually.

Method	MAP	R-Prec	P@10	Sig.?
Baseline	0.1797	0.2452	0.3560	
Query Expansion				
Manual (3 terms)	0.1848 (+3%)	0.2528 (+3%)	0.3880 (+9%)	+
BRF (3 terms)	0.1808 (+1%)	0.2477 (+1%)	0.3720 (+4%)	−
BRF (5 terms)	0.1832 (+2%)	0.2454 (+0%)	0.3840 (+8%)	−
BRF (10 terms)	0.1845 (+3%)	0.2451 (−0%)	0.3800 (+7%)	−
BRF (20 terms)	0.1888 (+5%)	0.2478 (+1%)	0.3840 (+8%)	+
BRF (40 terms)	0.1811 (+1%)	0.2414 (−2%)	0.3780 (+6%)	−

Table 9.11: Effect of query expansion on opinion retrieval in blogs.

The topics with the highest increase in performance due to blind relevance feedback, as well as those with the highest drops in performance, are shown in Table 9.12. We examined a range of aspects, including the number of relevant documents per topic, the ratio of opinionated to unopinionated relevant documents, early precision, the degree to which relevant documents are concentrated in a short time-period and others, but we were not able to identify factors which indicate a topic is likely to benefit from blind relevance feedback. In comparison, Yom-Tov et al. [324] recently reported on a complex system predicting the difficulty of a query; one of its applications, selective query expansion, attempts to improve retrieval results by adding terms only to "easy" queries—queries with predicted high early precision. However, this method results in minor improvements, if any, over usage of query expansion in all cases, and we did not experiment with it further in the context of blog search. Our only conclusion, then, is that blind relevance feedback is indeed useful for opinion retrieval—although somewhat less so than for other domains.

9.3.5 Term Dependence

Next, we measure the usefulness of assigning higher scores to documents which contain the topic terms in close proximity, by reranking the base retrieval rank-

	Average Precision		
Topic	**No BRF**	**With BRF**	**Change**
Largest performance gains			
892. "Jim Moran"	0.4372	0.6323	+0.1951
865. basque	0.1082	0.2826	+0.1744
894. board chess	0.2439	0.4179	+0.1740
Largest performance drops			
858. "super bowl ads"	0.1568	0.0800	−0.0768
885. shimano	0.1584	0.0806	−0.0778
880. "natalie portman"	0.2621	0.1436	−0.1185

Table 9.12: Extreme performance changes when using blind relevance feedback.

ing according to the distance between topic terms in the document, as described in [200, 191]. Of the 50 TREC topics, 35 (70%) contain more than one word; of these, all but three are syntactic phrases, the vast majority of which are named entities—leading us to believe that using phrase-based methods will improve performance. We applied the optimal parameters of a method we developed for using proximity information in the context of web retrieval [200], assigning higher scores to documents based on the distance between the topic words in them; for web retrieval, this has been shown to substantially improve both early and overall precision. Results are shown in Table 9.13; the performance improvements—6% in mean average precision and 3% in precision at 10—are statistically significant, but closer to the average improvement reported for newswire text (5%) than to the improvements observed in web domains (13%) [191, 200]

Method	**MAP**	**R-Prec**	**P@10**	**Sig.?**
Baseline	0.1797	0.2452	0.3560	
Proximity-based scoring				
Optimized for early precision	0.1888 (+5%)	0.2627 (+7%)	0.3680 (+3%)	+
Optimized for average precision	0.1902 (+6%)	0.2610 (+6%)	0.3620 (+2%)	+

Table 9.13: Factors contributing to opinion retrieval in blogs: proximity scoring.

To understand the reason for the difference between the effectiveness of term dependence models for opinion retrieval and for other domains, we turn to a per-topic examination of the retrieval results.

Generally, models taking into account term dependence through word distance are particularly useful for queries whose terms constitute a non-compositional collocation, rather than a syntactic phrase. Topics containing phrases will benefit little from proximity scoring, since their terms will usually appear within the same distance in the document (e.g., as consecutive words); term dependence models will not differentiate between different documents containing them. On the other hand, queries containing words which appear separately more often (and, when appearing separately, do not refer to the same concept as when appearing together) will benefit more from taking into account the term proximity during ranking.

To quantify the "phraseness" of a topic, we return to pointwise mutual information (PMI), which was used in Section 3.4; this measure has been reported as useful for identifying phrases in other contexts [297]. The link between PMI and effectiveness of proximity methods is demonstrated in Table 9.14, which shows the topics gaining most from the term dependence approach, and the topics benefiting least—along with their PMI values.

	Average Precision			Norm.
Topic	**No proximity**	**With proximity**	**Change**	**PMI**
Largest performance gains				
881. "Fox News Report"	0.0292	0.1047	+258%	13.68
886. "west wing"	0.0693	0.2229	+222%	14.16
860. "arrested development"	0.0480	0.1526	+218%	13.61
Largest performance drops				
872. brokeback mountain	0.3270	0.2697	−17%	16.60
854. "Ann Coulter"	0.5289	0.4827	−9%	16.47
871. cindy sheehan	0.4511	0.4298	−5%	18.41

Table 9.14: Extreme performance changes when using proximity information, with normalized PMI-IR values.

Examining the PMI values of the multi-word topics in the opinion retrieval task, we suggest that the reduced effectiveness of term dependence models is caused by the types of queries typical of the task, rather than the corpus itself. As noted earlier, most topics in the task are named entities—syntactic phrases, rather than other forms of collocations. Indeed, when comparing the PMI values of the topics used in the opinion retrieval task with other TREC topic sets, we observe notable differences: whereas the average PMI for blog topics is 16.41, the average PMI for distillation topics at the 2003–2004 Web Tracks at TREC was 15.50, and the average for the topics used at the Terabyte Tracks during 2004–2006 was 14.84.

Observing the relation between the term dependence as measured with PMI and the performance gained through proximity information, we experimented with applying proximity information only for topics whose PMI exceeds a threshold. Although there was a positive correlation (0.45) between the PMI-IR values and the change in retrieval scores, we experienced only minor improvements (of about 1% to mean average precision) which were not statistically significant. However, a more involved method, such as weighing the proximity information according to the PMI, may lead to more noticeable gains.

9.3.6 Recency Scoring

Until now, the techniques we applied to improve topical ranking have not been specific to blog data. We now turn to examine a blogspace-related technique we proposed in the previous section: usage of the estimated query time. As described earlier, we combine two ranked lists: the first is our baseline retrieval run, and the second ranks documents according to the distribution of dates in the top-ranked k documents (in the experiments reported here, we used $k = 100$). Table 9.15 shows the effect on retrieval performance: gains are substantial, in particular for early precision.

Method	**MAP**	**R-Prec**	**P@10**	**Sig.?**
Baseline	0.1797	0.2452	0.3560	
Recency Scoring				
Optimized for early precision	0.1807 (+1%)	0.2533 (+3%)	0.4140 (+16%)	–
Optimized for average precision	0.1883 (+5%)	0.2587 (+6%)	0.3880 (+9%)	+

Table 9.15: Effect of recency scoring on opinion retrieval in blogs.

Not all topics used at TREC are indeed recency ones. However, in our approach, topics which are not related to a particular event will result in a relatively flat distribution of dates in the top-ranked results. This will, in turn, lead to more uniform prior values for each date, meaning that the recency scoring will not play an important role. In our experiments, we found that manually excluding these queries which do not seem as recency ones did not improve over handling all queries as recency ones.

Opinion expression

We now turn to the second aspect of opinion retrieval we identified, and the one most characteristic of this particular retrieval task: identifying, for a given blog post, whether it contains opinionated content about a topic.

9.3.7 Content-based Opinion Scores

Table 9.16 shows the results of applying the lexical-based sentiment reranking methods to different texts: the post, topical sentences in the post, and the entire feed. Clearly, improvements are substantial—particularly when examining the topical sentences only. Interestingly, the feed level reranking and the post level reranking are statistically different, meaning that the ranking of opinionated content by post or by feed differs substantially. However, combining both of them (last line in Table 9.16) leads to minor improvements only. One reason for this may be the conflicting values of both measures: some blog posts are highly opinionated, but appear in a mostly non-opinionated blog, and vice versa.

Method	MAP	R-Prec	P@10	Sig.?
Baseline	0.1797	0.2452	0.3560	
Entire feed text	0.1932 (+8%)	0.2591 (+6%)	0.3920 (+10%)	+
Entire post text	0.1964 (+9%)	0.2610 (+6%)	0.4000 (+12%)	+
Topical sentences only	0.2112 (+18%)	0.2878 (+17%)	0.4180 (+17%)	+
Post and feed, combined	0.2159 (+20%)	0.2855 (+16%)	0.4180 (+17%)	+

Table 9.16: Effect of lexical-based sentiment analysis on opinion retrieval in blogs.

As an aside, we also experimented with limiting the dictionary of sentiment words to pronouns only: posts were reranked simply based on the frequency of pronouns in the topical sentences. While the improvements achieved were not as high as using the entire lexicon, we still experienced increased accuracy: a MAP of 0.1933 (+8% over the baseline) and P@10 of 0.3660 (+3%)—substantial improvements, given the low effort involved.

Moving on to the other type of sentiment analysis we employed, Table 9.17 shows the performance gains using the text classification-based sentiment analysis method we used, separately for each of the training sets we experimented with.

The movie review data proved to be too domain-specific for building useful classifiers; an examination of the top-ranked features by the classifier showed that most important words were indeed in the domain of films, such as "watched."

Using posts tagged with terms indicative of sentiment did lead to some improvements, although marginal and not statistically significant. Examining the top-ranked features for this training set showed a mix of useful words with a large amount of noise; possibly, a much larger training set will improve the performance of such an approach.

Finally, using the actual opinionated posts, as judged by TREC assessors, in the leave-one-out manner we described, we experienced substantial performance gains over the baseline—slightly better than those obtained with the lexical approach when using the entire text of the post. We conclude that the two approaches—lexical and machine learning one—perform similarly, and did not experiment further with limiting the training and test sets to topical sentences only.

Method	MAP	R-Prec	P@10	Sig.?
Baseline	0.1797	0.2452	0.3560	
Trained on movie review data		*No improvements*		
Trained on tagged posts	0.1840 (+2%)	0.2474 (+1%)	0.3560 (+0%)	–
Trained on TREC data, leave-one-out	0.1984 (+10%)	0.2667 (+9%)	0.4020 (+13%)	+

Table 9.17: Effect of text classification-based sentiment analysis on opinion retrieval in blogs.

9.3.8 Structure-based Opinion Scores

Evaluating our final method for opinionated content detection in blogs, Table 9.18 shows the effect of using comment information: the contribution is minor, possibly due to the partial success of our comment extraction mechanism.

Method	**MAP**	**R-Prec**	**P@10**	**Sig.?**
Baseline	0.1797	0.2452	0.3560	
Comment-based reranking				
Optimized for early precision	0.1807 (+1%)	0.2447 (−0%)	0.3920 (+4%)	–
Optimized for average precision	0.1828 (+2%)	0.2498 (+2%)	0.3660 (+3%)	+

Table 9.18: Factors contributing to opinion retrieval in blogs: comment-based sentiment analysis.

Overall, and in-line with intuition, content-based approaches to locating opinionated content in blogs prove highly beneficial for the opinion retrieval task. Two notable observations from the results we presented are the inapplicability of existing, domain-specific sentiment analysis training sets to the task, as well as the relatively high performance gains achieved with simple pronoun-based counts.

Post Quality

Finally, we examine the third aspect of opinion retrieval relevance we referred to earlier: the quality of the blog post. Recall that quality, in our approach, is composed of two distinct factors: spam blogs are considered low-quality ones, and are modeled through a spam likelihood score; in addition, the authority of a blog post is estimated using incoming links. We report on the success of these two factors separately.

9.3.9 Spam Filtering

Table 9.19 shows how the spam classification methods we used affect retrieval results. While compression ratio alone does not result in a significant improvement, the combination of compressibility with the text classifier we trained does provide a substantial increase in performance. In particular, when optimizing the combination of topical relevance and spam likelihood for early precision, precision at 10 increases by 25%.

Spam and commercial intent. The average performance gain when using the combined spam filtering approach is 9%; but a per-topic analysis of the contribution to each of the 50 topics shows varying levels of contribution to different queries. For 39 topics spam filtering increased precision, and for 10 topics precision was reduced (for the remaining topic no change occurred). Table 9.20 shows

Method	MAP	R-Prec	P@10	Sig.?
Baseline	0.1797	0.2452	0.3560	
Feed-level Spam Filtering				
Using compressibility				
Optimized for early precision	0.1678 (−7%)	0.2426 (−1%)	0.4000 (+12%)	−
Optimized for average precision	0.1857 (+4%)	0.2561 (+4%)	0.3640 (+2%)	+
Using text classification				
Optimized for early precision	0.1719 (−4%)	0.2474 (+1%)	0.4140 (+16%)	−
Optimized for average precision	0.1899 (+6%)	0.2585 (+5%)	0.4040 (+13%)	+
Combining compressibility and classification				
Optimized for early precision	0.1886 (+5%)	0.2594 (+6%)	0.4460 (+25%)	+
Optimized for average precision	0.1961 (+9%)	0.2633 (+7%)	0.4140 (+16%)	+

Table 9.19: Effect of spam filtering on opinion retrieval.

the topics for which the improvement was most substantial, and the topics where performance degraded the most. Other than topic 859, as expected, all topics where spam filtering is highly beneficial are commercially-oriented; topics where performance is degraded are assorted sports, politics, and miscellaneous ones.

	Average Precision		
Topic	Baseline	Spam filtered	Change
Largest performance gains			
883. heineken	0.1924	0.4055	0.2131 (+110%)
893. zyrtec	0.0413	0.0826	0.0413 (+100%)
885. Oprah	0.1241	0.2445	0.1204 (+97%)
877. sonic food industry	0.0234	0.0459	0.0225 (+96%)
859. “letting india into the club?”	0.0064	0.0115	0.0051 (+80%)
Largest performance drops			
882. seahawks	0.0428	0.0373	−0.0055 (−13%)
871. cindy sheehan	0.4511	0.4014	−0.0497 (−11%)
892. “Jim Moran”	0.6152	0.5499	−0.0653 (−11%)
880. “natalie portman”	0.2332	0.2106	−0.0226 (−10%)
887. World Trade Organization	0.0331	0.0303	−0.0028 (−8%)

Table 9.20: Extreme performance changes when using spam filtering.

Observing this, we hypothesized that if we can predict the degree to which a query is commercially-oriented, we may be able to improve average performance by increasing the importance of spam filtering for those highly commercial queries, and decreasing it for the non-commercial ones.

Sophisticated methods for deriving commercial intent of queries require manually annotated data as well as an analysis of the landing pages of queries (e.g., [62]); we adopt instead a much simpler approach which requires no training, and is somewhat similar to the PMI method we used earlier to identify syntactic phrases. To estimate the level of commercial intent behind a query, we simply measure its correlation with a term known to relate to commercial activities, again using pointwise mutual information. We refer to this as the query commercial intent value, and use it (after normalization) as a query-specific weight for spam reranking, instead of using a fixed, query-independent weight.

Table 9.21 lists the query commercial intent values of the most commercially-oriented queries according to this method, and the least commercially-oriented ones. In this case, the term used as highly correlated with commercial activity was "shop." Some correlation with the most and least benefiting queries from Table 9.20 is visible, and indeed using this commercial intent identification method to weigh the spam component of different queries separately improves retrieval performance. Table 9.22 compares the performance of query-independent weights (as shown earlier in Table 9.19) and query-specific weights, showing a relative improvement of 35% (and absolute of 3%) by using the commercial intent estimation.

Topic	**Query Commercial Intent**
898. Business Intelligence Resources	14.20
893. zyrtec	13.38
885. shimano	13.18
873. "bruce bartlett"	13.13
856. macbook pro	12.80
892. "Jim Moran"	10.86
869. muhammad cartoon	11.05
897. ariel sharon	11.18
870. "barry bonds"	11.21
878. jihad	11.24

Table 9.21: Query commercial intent values for independent query weighing in spam filtering.

While the total contribution of spam filtering in the TREC settings is substantial, we believe its importance in some real-life blog search scenarios is even higher, for a number of reasons. First, the prevalence of spam in the blogspace is higher than in the TREC collection: while 15% of the Blog06 documents are known spam [235], estimates about the percentages of spam in the blogspace are 20% (and higher for new blogs) [149, 125]. Second, most of the 50 topics in the 2006 task were not highly commercially-oriented, and their benefit from spam filtering was limited; for marketers searching the blogspace for references to products—clearly, commercially-oriented queries—the average improvement is likely to be higher.

9.3.10 Link-based Authority

Finally, we evaluate the success of incorporating link-based influence measures into the retrieval process. The total number of links we extracted from the Blog06

Method	MAP	R-Prec	P@10	Sig.?
Baseline	0.1797	0.2452	0.3560	
Query-independent weights				
Optimized for early precision	0.1886 (+5%)	0.2594 (+6%)	0.4460 (+25%)	+
Optimized for average precision	0.1961 (+9%)	0.2633 (+7%)	0.4140 (+16%)	+
Query-specific weights				
Optimized for early precision	0.1952 (+8%)	0.2630 (+7%)	0.4500 (+26%)	+
Optimized for average precision	0.2019 (+12%)	0.2703 (+10%)	0.4200 (+18%)	+

Table 9.22: Spam filtering: query-independent and query-specific weights.

corpus, where both the linking and linked page are found in the collection, is 25.2 million. However, a large number of these are intra-blog links: links between posts in the same feed, often automatically generated by the blogging platform (e.g., automated archives, “next post” and “previous post” links, and so on). In total, less than 5 million of the links (20%) connected two posts from different blogs. As the internal links are mostly automated and not indicative of authority in the manner usually associated with links, we disregard them in our analysis. The distribution of both in- and out-degree of the remaining links—those between different blogs—follows a power-law, as already shown in Section 2.2.2.

We attempted to rerank blog posts by their indegree, log indegree, the indegree of the feed they belong to and the log of the feed indegree; this type of reranking has shown to substantially improve retrieval for certain web retrieval tasks [300]. But for the opinion retrieval task, not only did none of these improve retrieval—in most cases, performance was substantially degraded. The assumption that posts with higher link-authority will be preferred by users turned out to be incorrect.

Surprised at this counter-intuitive result, we examined the link profile of documents judged at TREC as relevant, compared to the profile of documents judged as non-relevant. We found that the average (external) link indegree of relevant posts was *lower* than that of posts judged non relevant: 2.4 links compared to 4.6 links, respectively (although the the average over all posts in the corpus, judged and not judged, was even lower at 1.9 links). A per-topic analysis of the contribution of link usage to retrieval showed that for a small number of topics usage of indegree information improves results substantially, while most topics incur moderate degradation in accuracy; Table 9.23 lists the top-gaining topics, as well as those where performance degrades the most. However, we could not identify a property setting apart those topics that benefit from link degree usage, so that it would be applied only to them. Note that while some topics exhibit very high performance gains in terms of percentages of improvements, the absolute gain is low. The overall decrease in performance results from the fact that the average precision of topics where performance degrades is substantially higher than the average precision of the topics benefiting from indegree reranking.

	Average Precision		
Topic	**Baseline**	**Indegree-reranked**	**Change**
Largest performance gains			
876. “life on mars”	0.0001	0.0067	0.0066 (+6,593%)
898. Business Intelligence Resources	0.0019	0.0228	0.0209 (+1,100%)
866. “Whole Foods”	0.0354	0.1437	0.1083 (+306%)
860. “arrested development”	0.0480	0.1894	0.1414 (+295%)
886. “west wing”	0.0693	0.2676	0.1983 (+286%)
Largest performance drops			
899. cholesterol	0.1279	0.0135	−0.1144 (−89%)
859. “letting india into the club?”	0.0064	0.0017	−0.0047 (−73%)
865. basque	0.2666	0.0904	−0.1762 (−66%)
878. jihad	0.0966	0.0373	−0.0593 (−61%)
895. Oprah	0.2220	0.1403	−0.0817 (−37%)

Table 9.23: Extreme performance changes when using link-based authority.

9.3.11 Component Combination

Finally, we combine all factors we have just described: components related to topical retrieval, opinion extraction modules, and methods for controlling the quality of the blog post. The result is a linear combination of several ranked lists, where the weight of each of the lists is set to the best weight obtained when optimizing the method producing it separately, when combined with the baseline. Table 9.24 shows the total contribution of the methods we developed for each of the three aspects we identify as contributing to opinion retrieval, as well as the performance gained by combining all factors; all results in this Table are statistically significant. The Precision/Recall graphs of the baseline and the combination of all factors are displayed in Figure 9.5. For comparison, the best-performing system at TREC 2006 obtained a MAP score of 0.2052, although direct comparison is misleading since we used information not available to TREC participants, for example for training the opinion-detection text classifiers. Having said this, our performance is substantially higher than the reported best performing system, also without using this information.

Clearly, all aspects we evaluate contribute significantly to performance in this domain; in particular, and in-line with intuition, sentiment analysis seems to have a crucial role. An additional observation is that, to some extent, the different approaches we propose are orthogonal: their combination improves substantially over any single approach.

9.4 Related Work

The domain of sentiment analysis is closely related to opinion retrieval, as we have seen from the contribution of sentiment analysis methods used to rerank topical

Method	**MAP**	**R-Prec**	**P@10**
Baseline	0.1797	0.2452	0.3560
Combination of all topical factors			
Optimized for early precision	0.2019 (+12%)	0.2797 (+14%)	0.4080 (+15%)
Optimized for average precision	0.2056 (+14%)	0.2835 (+16%)	0.4020 (+13%)
Combination of all post quality factors			
Optimized for early precision	0.1952 (+8%)	0.2630 (+7%)	0.4500 (+26%)
Optimized for average precision	0.2019 (+12%)	0.2703 (+10%)	0.4200 (+18%)
Combination of all opinion-level factors			
Optimized for early precision	0.2222 (+24%)	0.2950 (+20%)	0.4840 (+36%)
Optimized for average precision	0.2271 (+26%)	0.3011 (+23%)	0.4400 (+24%)
Overall combination			
Optimized for early precision	**0.2306** (+28%)	**0.3050** (+24%)	**0.5100** (+43%)
Optimized for average precision	**0.2411** (+34%)	**0.3122** (+27%)	**0.4900** (+38%)

Table 9.24: Combination of all factors contributing to opinion retrieval; all results are statistically significant.

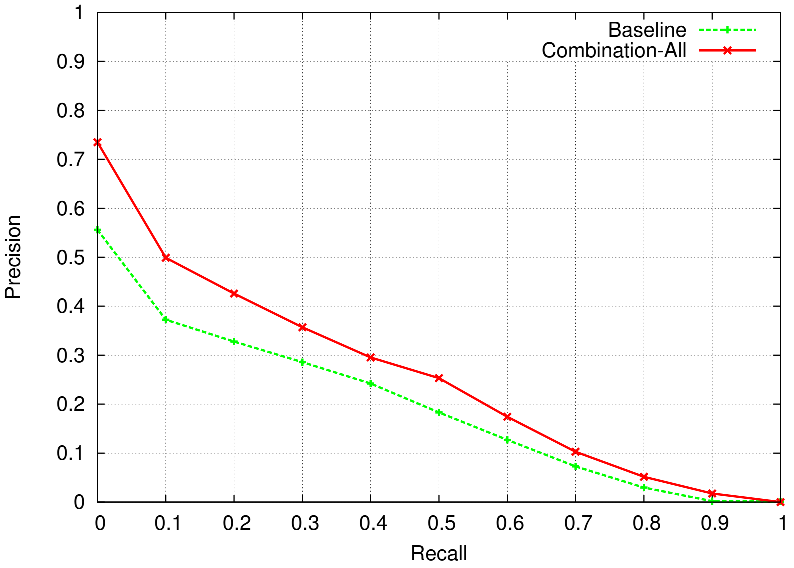


Figure 9.5: Precision/Recall graph for the baseline retrieval model (dashed, green line) and the combined model (solid, red line).

retrieval results. Sentiment analysis is a research area with a relatively long history (see, e.g., [270]); some of the main works in this field have been described in Section 2.3.2. While sentiment analysis deals, broadly, with extraction and characterization of emotions, opinions, and other subjective aspects of text, most of the work in the field concentrates on sentiment classification—identifying positive and negative opinions towards a given topic. Typically, sentiment classification tasks are used in settings where the analyzed text is known to contain opinion (e.g., product reviews [65, 83]), although applications to other domains have been proposed (e.g., news corpora [311]). Sentiment classification work which is directly related to our work involves applying it in large-scale retrieval settings; this type of research has only recently started to emerge, and we concentrate on it.

Opinion retrieval at the TREC Blog Track. In parallel to the work described here, a number of similar approaches to opinion retrieval in blogs have been developed by participants in the opinion retrieval task at TREC 2006 [235]. Most participating systems were based on a two-stage approach: in the first stage, documents are retrieved using standard ranking schemes (i.e., not tailored to opinion retrieval); sometimes, various retrieval heuristics such as relevance feedback are used. In the second stage, the documents are filtered, or their retrieval score is modified, based on identification of an opinion expressed towards the topic in them. For this second stage, most approaches used existing sentiment analysis techniques, either using publicly-available sentiment lexicons or by building text classifiers. The classifiers were trained, given the absence of training data at this first run of the Blog Track at TREC, on corpora perceived as opinionated (such as product reviews) and corpora perceived as non-opinionated (such as Wikipedia). As with our experience, this type of training data was too domain-specific to create useful models—except where specific methods were implemented to adapt it and extract more general knowledge [229]. The overall success of the opinionated retrieval component of these approaches was moderate; on average, methods for reranking topical retrieval by opinion level achieved 5%–10% improvement over their topical-retrieval component, with the top-performing systems (including one based on methods described in this chapter) showing an overall improvement of about 20%. A robust topical retrieval ranking formula was reported to perform almost as well as top-performing systems [235].

A different, single-stage approach to opinion retrieval which was tested at the TREC Blog Track was proposed by Attardi and Simi [15]. Here, a sentiment dictionary is merged into the retrieval index by storing explicit information about the polarity value of words. Given a retrieval model which can handle proximity searches, this enables queries such as "⟨word⟩ NEAR ⟨POLAR-WORD⟩, where ⟨word⟩ is taken from the original query, and POLAR-WORD is the meta-information stored in the index. In this approach, opinion retrieval is instantiated as a query rewrite process, resulting in high efficiency, without a substantial loss in effectiveness; in terms of performance at TREC, the approach exhibited early precision scores which were on-par with top performers, although average precision was lower. A similar approach is used in a commercial setting, focusing on early precision rather than recall [192].

Opinions in other retrieval contexts. A task related to opinion retrieval, *discussion search*, was carried out as part of the Enterprise Track at TREC, a track exploring information seeking behavior in intranets [57]. The aim of the discussion search task was to locate emails containing a discussion of a given topic, including comments in favor or against it, in a large collection of email messages; participants in this task used mostly non-content features such as email thread length or usage of quotes to detect the presence of opinionated content. Usage of sentiment analysis through a text classification approach has resulted in relatively minor improvements over robust topical-relevance approaches [315, 326].

Another TREC task related to opinionated content was part of the TREC Novelty Track, a track aimed at developing methods for identifying relevant and novel information about a topic—information which is not found in previous relevant results found by a system [280, 279]. Part of the topics used in the Novelty Track were "opinion topics," and the task was to identify sentences in the form of an opinion about these topics. Participants in the track approached these topics with a range of sentiment analysis methods, but none seem to have significantly improved non-sentiment oriented approaches. One lesson learned from the task was that opinion topics were harder than other topics: identifying opinionated content proved a difficult task.

In [76], Eguchi and Lavrenko develop a generative language modeling approach for the task of sentiment retrieval—retrieval of opinionated content of a given polarity (positive or negative). In this framework, language models are estimated for sentiment-bearing sentences from an annotated corpus; these models are then combined with topical language models such as those typically used in language modeling-based retrieval. Although initial experiments with this approach on a small (500 document) corpus were promising, it did not improve retrieval accuracy when applied to the opinion retrieval task at TREC. A similar approach is described in [116], also on a small-scale corpus (in Japanese) and also focusing on retrieval at the sentence level. Here, annotated content is used to train a text classifier to distinguish between opinionated and non-opinionated sentences; the output of the classifier is used to filter results of a standard retrieval approach.

Finally, retrieval of blogs is related to web retrieval. Prominent approaches in the web retrieval domain include usage of HTML structure, the web link graph, anchor text, and user behavior, among others; good overviews of the main aspects of web retrieval are found in [49, 21, 22]. As blogs are a specialized form of web pages, some of the techniques we applied to the opinion retrieval task—e.g., usage of links or term proximity—were based on ideas that have been successful in the web retrieval context.

9.5 Conclusions

This Chapter addressed the task of opinion retrieval in blogs: identifying blog posts that express an opinion about a topic. Our research questions in this Chapter were aimed at identifying the different components of this task, their contribution to retrieval performance, and their mutual dependence, as well as the performance of more traditional retrieval approaches in this domain.

Answering the first of these questions, regarding different factors of this task, we identified three aspects which potentially contribute to success of opinion retrieval: robust topical retrieval, identification of opinionated content, and control of blog post quality. Each of these three aspects was instantiated both by applying known methods (e.g., proximity-based retrieval reranking to improve the topical component of the retrieval), and by testing techniques which we developed specifically for blog data. Among the specialized techniques tested are a recency scoring method which benefits from the temporal nature of blogs; approaches to sentiment analysis which make use of tags and comments; and a query-dependent weighting scheme for blog spam filtering which substantially improves over non-query-specific methods.

To measure the contribution of each of these factors to performance in the opinion retrieval task, we evaluated them separately, comparing them with a robust baseline. The results indicate that while all components are important, sentiment analysis methods are critical both to early precision, which is important to the incidental searcher who is examining the top-ranked results, and to overall performance, which is important to marketers and professionals aggregating results from multiple posts. We have also observed that relatively simple, inexpensive methods of sentiment analysis provide satisfactory results. Another aspect we identify as contributing significantly to accuracy of opinion retrieval is spam filtering—which is particularly crucial for early precision; we expect this contribution to be even higher when retrieving from the entire blogspace. Finally, an important observation is the lack of improvements gained by using link authority methods which are reported to work well for other web domain, indicating that, for opinion retrieval, users do not necessarily prefer highly-linked-to posts. This demonstrates that blog posts are not just a form of web pages, and blog retrieval is not a subtask of web retrieval: there are substantial differences between the two.

Addressing the second of our research questions, regarding the performance of traditional information retrieval techniques in this domain, we experimented with varying the parameters of the retrieval model, testing aspects such as the type of content used for retrieval (HTML compared to syndicated content) and the effectiveness of different ranking schemes. These lead to an observation about the performance of current (commercial and other) blog search engines. Many of these base their retrieval model exactly on those aspects which were found in this chapter to be sub-optimal, such as usage of RSS feeds, application of basic, off-the-shelf retrieval models, and ranking of results by number of incoming links. We demonstrate that, at least for opinion retrieval, which is a prominent type of search in the blogspace, this can be significantly improved.[8]

Finally, answering the last of our research questions, regarding the mutual dependence between the different factors, we show that—to some extent—the three aspects we explore are orthogonal: a combination of all methods results in substantial improvements over using any of them separately. Overall, this combination shows both significant gains over a baseline and over existing state-of-the-art.

Open issues. One factor limiting our success was the availability of training data for some of the approaches: for example, for developing a spam classifier we used naive assumptions which resulted in a biased training set. We believe that as research in this domain continues, with datasets and results being made available to the community, the success of our approach will further increase, without requiring substantial modifications.

Additionally, while we have not benefited from link-based authority measures, we nevertheless believe that links are useful in the context of retrieval and can be used to improve results. With much larger collections, it may be possible to develop link-based methods which are useful for this task; e.g., the "linking profile" of a blogger—whether she often links to pages known to contain opinions such as newspaper op-ed columns—can be used in the same manner as we used the sentiment analysis-based "feed level opinion level." On top of this, links may be used in the opinion retrieval context for tasks other than authority estimation: for example, to propagate opinions from linking to linked page.

Finally, the lack of benefit from links raised a different question altogether, about the nature of the opinion retrieval task. The user scenario in the task is a market-research one: users are assumed to be tracking the response of people, through their blogs, to events, products, and brands. This is not the only user profile associated with this type of task: individuals searching for opinions in blogs may prefer to invest their time reading opinions expressed by leading commentators of a field, rather than by unknown, uncredited bloggers. Consequently, systems addressing the needs of these users may need to focus on additional areas: identifying domain expertise, or credibility of opinions; here, link-based authority may have benefits.

[8] Recall that throughout our experiments, we compared our performance with that of a baseline which already performs better than the sub-optimal solution; comparing our combined model with an RSS-based, basic ranking model approach would have resulted in substantially higher improvements.

Conclusions for Part III

Viewing the task of search as a prominent way in which people currently access information, this final part of the thesis set out to investigate search behavior in the blogspace. We started with an analysis of a blog search query log, seeking to understand the differences between search patterns in this domain and in the web. We discovered that while some behavioral properties are shared among web and blog search—short queries, short sessions, focus on first few results—the type of queries issued differs more notably. Of the three types usually used to classify web queries—navigational, transactional, and informational—blog searches are mostly of the latter type. Further, those informational queries tend to be named entities, and in many cases are related to ongoing events at the time of the query.

The type of queries sent to blog search engines indicates that the relevance of blog posts is sometimes measured not by the amount of information they supply for a given query, but by the degree to which they express a person's thoughts or comments about a topic. Consequently, the second Chapter in this part explored the task of opinion retrieval: locating posts which contain an opinion about the query. We investigated a wide range of techniques for improving retrieval, grouped into three aspects: techniques aimed at improving topical relevance alone, techniques focusing on detecting sentiment in a blog post, and techniques for incorporating the quality of a post into the ranking process. Overall, we observe that most of the methods we developed improve retrieval effectiveness—sometimes, significantly so. Combining insights from both chapters in this part, we observe that user needs, particularly regarding the type of content they search for, are different in the blogspace; applying techniques which are known to be effective for web retrieval does not necessarily answer those needs, which are best addressed with other, specialized methods.

Chapter 10

Conclusions

10.1 Answers to Research Questions

In the beginning of this thesis, we formulated several questions we intended to answer. We now revisit these questions, examining the results we found along the way.

1. What types of information can be mined from the personal, informal content found in blogs? How can text analysis be used to extract this knowledge, both at the individual blog level and at the aggregate level?

Throughout this work, we used the blogspace as a source of information for two types of knowledge. The first is *people-centric* information: an individual's interests, preferences, and surroundings. As unmoderated accounts of people's daily lives, we view blogs as a unique window to such knowledge. The second type of information we identify in blogs is *non-factual* content: emotions, opinions, and moods. Other sentiment-related collections of data exist, and have been studied extensively; however, they are usually domain-specific, commercially-oriented ones (e.g., product reviews). The sentiment expressed in blogs is not restricted to one domain, and differs from these collections in two additional, important aspects: the amount of data, and the subtlety of the sentiment. In terms of size, the volume of the blogspace as a collection of opinions far exceeds other collections of subjective content; as for the depth of opinions expressed, no other corpus gives direct accounts of such personal information as the mood of the author.

Can these types of knowledge be mined computationally, from the text used by the bloggers? We have shown the answer to this question to be affirmative, utilizing a range of methods to extract both personal information and opinionated content. We have also seen, in Part II, that text analytics, when applied to large collection of blogs, yield results which are more than simply the aggregation of many single blogs: new knowledge emerges, such as global emotional patterns or the type of topics raising online controversy.

2. How do existing text analytics methods perform when applied to blog content? How can known approaches benefit from properties of blogs?

In several scenarios, we found best-practice text mining approaches to be suboptimal when applied to blogs. For example, state-of-the-art retrieval methods did not necessarily address the relevance required by search tasks common in the blogspace; a standard keyword extraction method performed poorly on the task of identifying useful tags for a blog post; best-practice contextual advertising approaches did not provide an optimal solution for blog content. A particular case is that of sentiment analysis, a task which is considered moderately difficult in terms of text mining, also out of the blogspace. As we have shown in Chapter 4, the more subtle forms of sentiment found in blogs—emotions and moods—make it an even harder task in this domain, particularly when coupled with the noisy, irregular language used in blogs. We have demonstrated that this task is not trivial for humans either.

However, we also found that—once the shortcomings of standard approaches are identified—features of the blogspace can be used to improve performance. In all examples we have just cited—blog retrieval, tag recommendation, and contextual advertising—we improved existing approaches through usage of blog properties. Supplementing traditional retrieval with modules based on blogspace feature such as timestamps or existence of comments substantially increase effectiveness; tag recommendation was made possible by utilizing the mass of existing, manually selected tags in the blogspace; contextual advertising was improved by combining representations of different components of a blog. For sentiment analysis tasks, one feature of the blogspace that greatly improves performance is the volume of data: at the aggregate level, we witness surprisingly accurate results, even with relatively simple methods.

In Chapter 1 we introduced three aspects which, we believe, set apart the blogspace from other data collections used for text mining: the personal nature of the content, its dynamic structure, and the temporal information available in the corpus; we attempted to incorporate these—particularly content-related aspects—in our approaches. A more general point emerging out of our experience with text analytics for blogs is that the key to success of these types of methods is to identify, for each task, how the unique properties of blogs can be combined into the process. Effective analysis of blog content addresses it not simply as text, but as a reflection of an individual: her preferences, opinions, and interactions with others.

3. How does blog search differ from web search? What are the differences and similarities in the types of information needs and the user behavior? What factors contribute to the effectiveness of blog retrieval, and, in particular, to the performance on those search tasks which are characteristic of the blogspace?

The last part of this thesis, Part III, was dedicated to search tasks in the blogspace. An important conclusion from the first Chapter in this part is that while search behavior is similar in blog search and web search—short sessions, few results examined—the user needs are different: queries sent to blog search engines tend to be named entities, and are often related to ongoing events. We concluded that a prominent type of information need in the blogspace is search for opinions and commentary—people's thoughts about a topic or recent development, as reflected in their blogs.

As a result, the second half of this Part introduced the *opinion retrieval* task, aiming at locating blog posts expressing an opinion about a topic. In Chapter 9, we developed a multiple-ranking approach to this task, where blog posts are ranked separately on different aspects contributing to relevance in this domain. We identified three such aspects: topical retrieval, opinion identification, and quality control, and developed blog-specific methods for ranking each; these were then evaluated within the TREC framework.

In terms of factors contributing to effectiveness of opinion blog retrieval, we found all three aspects we identified to be important to success in this task; in particular, accurate ways of classifying sentiment in blogs, coupled with robust spam filtering, are crucial to success. We also found that the aspects are orthogonal, meaning that a divide-and-conquer approach as we have used is both practical and effective: methods addressing one area can be optimized independently from others.

Themes

One set of approaches we found useful throughout different tasks in this dissertation is those involving statistical language modeling. We used it to extract keywords from individual blogs (for profiling) and from multiple blogs (to identify a set of predictors for moods and to annotate fluctuations in temporal mood patterns); but also to compare the language used in a blog with the language used in comments for spam filtering, as well as in the retrieval settings. Although blog language is indeed informal, unfocused, and irregular, statistical methods prove robust even in this environment, and are important building blocks in text analysis applications.

An additional theme recurring in multiple topics we presented is viewing blogs as semi-structured data, partitioning the text to be analyzed by its structure. The difference segments within a blogs were then combined (e.g., for post-level profiling, aimed at contextual advertising) or contrasted (e.g., for comment spam filtering); in other scenarios, this partition was used to identify knowledge about the blog itself (e.g., identifying disputed topics through comment threads, or combining post-level and blog-level opinion levels for retrieval).

10.2 Main Contributions

We group the contributions of this work into three different areas: basic analysis and understanding of the blogspace, contributions related to text analytics applications, and contributions specific to the domain of blog search.

Domain Analysis and Understanding

As a relatively young domain, much of the research involving blogs is of an exploratory nature, identifying characteristics of the blogspace and ways in which it differs from other web domains. This dissertation adds a number of contributions to the basic study of blogs; these contributions were, at the time they were made public, the first of their kind, and include:

- A study of search behavior in the blogspace based on a large-scale query log, identifying differences and similarities between search of the blogspace and search of other web domains.
- Empirical results showing, for the first time, that the aggregate sentiment found in references to products in blogs is useful for business intelligence, and improves on using the discussion volume only.
- A description of the "commentsphere"—the collection of all blog comments—providing statistics about this often-overlooked domain as well as an analysis of the type of content it contains and how it can be used in text mining tasks.

Text Analytics Applications for the Blogspace

The focus of this thesis is on text analytics: applying and developing a range of methods based on text classification, information extraction and computational linguistics for mining knowledge from blogs. Within this framework, we introduced a number of novel techniques for existing information access tasks, as well as introduced new tasks for the domain. Our contributions include:

- A spam filtering approach based on measuring the divergence between the language used in two texts, useful for blog comments and for similar texts containing content and responses to it.
- A method for language-based blogger profiling and its application to product and advertisement matching, showing improvements over state-of-the-art approaches.
- Novel text classification tasks for blogs—mood classification and automated tagging—and a detailed solution for each.
- A platform linking global mood levels with blog content, including a method for fast, online estimation of global moods using the text of blog posts, and

a method for identifying and generating explanations for irregularities in the temporal patterns of these moods.

Blog Search

Finally, the last part of this thesis studied search in the blogspace, recognizing the prominence of information retrieval as an access method to large collections of data. In particular, we focused on the opinion retrieval task—a task which is relatively unique to blogs—offering the following contributions:

- Several novel techniques to improve opinion retrieval in blogs, based on properties which are characteristic of the blogspace; these include recency-based prior likelihood scores, usage of comments to identify the level of sentiment in a blog post, and query-dependent spam filtering for blogs.
- A state-of-the-art opinion retrieval system for blogs, incorporating these techniques as well as a wide range of other components, and achieving same-or-better accuracy levels as other systems designed for this task.
- Detailed evaluation of the effect of each of the components of opinion retrieval on retrieval results, as well as the effect of more traditional retrieval heuristics.

10.3 Future Directions

Research of blogs is in a relatively early phase, and there are many additional paths to explore. For example, as this thesis focuses on textual content rather than the structure of the blogspace, we have not invested much in exploring links in this domain: this type of analysis shows interesting results also for text analysis tasks (e.g., [2]). Of the possible additional work in knowledge mining from blogs we list here, we focus on those directions which are direct extensions to work developed in this thesis, or benefit from this work significantly; we address separately work related to the first two parts of the thesis—text analysis applications—and work related to the third part, blog search.

Text Analytics

Sentiment analysis in the blogspace. There are two distinct research areas involving sentiment analysis and classification in blogs. The first involves development of methods for performing the classification in this domain: how existing approaches perform on blog content, and how they can be adapted to it. The second research area concerns applications of the sentiment found in blogs: in what tasks it can be used and how.

In terms of improving sentiment analysis methods for blog-like content, the main challenge remains the informal, irregular language used in blogs, as well as the short length of many blog posts. Compared to traditional collections used to mine sentiment such as product reviews, blogs tend to be less focused; sometimes, sentiment towards an entity is expressed only briefly.

As in other domains, the availability of large datasets for training will gradually improve accuracy of sentiment analysis in this domain; manual annotation, however, is expensive. One possibility here is to use the bloggers' own annotations—mood tags or category tags (such as "rant")—in a bootstrapping process, as a seed of training data. We have seen in Chapter 9 that usage of tags as an indication of subjective content is useful, to some extent, in the context of opinion retrieval (as a way of collecting training data); in a pure sentiment classification task, it may be more beneficial.

Regarding the second research area, which focuses on applications of mining subjective content in blogs, a natural direction to follow is to connect sentiment analysis to blogger profiling. This combination of the two types of knowledge we extracted from blogs using text analysis—personal information and subjective content—has substantial benefits for business intelligence applications: not only will blogs provid general observations of people's opinions about products and brands, but the opinions will be classified by audience type. In this context, there are various useful types of blogger profiles: those concentrated on interest areas such as we used in our work, but also those based on demographics, characterizing the blogger in terms of gender, age group, or geographic location. A different type of knowledge about a blogger which can be used for deeper aggregate sentiment analysis is information about the blogger's influence: to what degree she is being read, cited, commented, and so on; combining this with sentiment analysis, marketers can follow opinion leaders and trend-setters.

Community-related analysis. As we have already mentioned, communities in the blogspace were not a main focus of this thesis, as they are strongly related to linking patterns rather than content. However, given a known community, we can extend some of our methods to mine useful information from the entire community, rather than from the separate blogs in it. For example, by comparing the language-based profiles of each of the members in the community, or constructing a profile of the entire community, we can identify whether this is a community centered around social contacts, or built on shared topical interests; in the latter case, the interests can be identified and used for the type of product and advertisement matching we demonstrated for individual bloggers.

Other methods developed here where community information may be useful are the automated tagging approach—where tags used by a blogger's community are used to enrich or filter the recommendations generated from the entire blogspace, and our opinion extraction modules—where a blogger's involvement in a community, and the responses of members from the community to her writings, can be used to estimate the level of opinion in a post or in an entire blog.

Temporal aspects. In Chapter 6, we witnessed the strength of following changes over time in the blogspace—utilizing both the affective content of blogs and their timelined nature to observe global sentiment patterns. Temporal analysis can also be applied to other methods we developed. For example, by following the profile of a blogger over time, we can identify changes in the interests, activities, and even behavior of the blogger. Other possible temporal extensions, for business intelligence from blogs, include not only aggregating the sentiment expressed in the blogspace towards a product as we have done, but also following its changes over time: this may be useful for tracking public response not about short-lived entities such as movies and books, but towards brand names or political issues.

Blog Search

In Chapter 8 we identified two main types of user needs in the blogspace: conversational and concept queries. Conversational queries are the prominent type of queries, and were discussed extensively in Chapter 9, as the main aim of the opinion retrieval task. Concept queries are those queries with which the user expresses an interest in high-level areas (e.g., "taxation," "enviroment"), rather than a particular named entity; often, the user is interested in results at the blog level, rather than the blog post level. The task related to these queries is *blog finding*: locating blogs which focus on a particular area, rather than *blog post retrieval*. This task is different from the task addressed in Chapter 9, which focused on single blog posts and stressed locating opinions in them—although, possibly, searchers at the blog level may also be more interested in blogs expressing opinions and commentary than blogs providing objective information only.

The blog finding task is somewhat similar to certain researched web retrieval tasks, e.g., topic distillation as explored at TREC [303]; some approaches can be directly borrowed. For example, in a blog finding scenario, link-based authority scores are likely to play a much more important role than in opinion retrieval; similarly, anchor text may be more useful. But the blogspace contains a wealth of additional information for estimating authority of blogs: readership (via direct access to subscription information, or through estimation); level of interaction (through comments, trackbacks, and quotes in other blogs); posting profile (post frequency, average length); and more. A multiple-ranking approach such as the one we used for opinion retrieval may be beneficial for blog finding too, utilizing some of this information about the blog.

Returning to the opinion retrieval task, the framework we proposed in this dissertation is a multiple-component one. As such, other modules can be added into it, and existing modules can be enhanced. For example, although we have used a state-of-the-art sentiment classification approach in this context, sentiment analysis is a rapidly changing field; additional heuristics to those we used may improve performance. In the experiments reported in Chapter 9, proximity between query terms and sentiment-bearing words was used only implicitly (when limiting the analyzed text to the sentences surrounding the topic). In some cases, more direct usage of the positional information of subjective terms was shown to improve accuracy [237], and may be also useful here. Similarly, other approaches such as domain-specific sentiment dictionaries [138] (rather than a global one as used by us) may also be incorporated into our framework. In terms of new modules that can be added, although we have shown that link analysis methods which are useful for web retrieval do not improve opinion retrieval in blogs, we believe that other, more subtle ways of using links have the potential to increase effectiveness. As an example, propagation of opinion information between blogs by following links for which the anchor text, or its surroundings, contain opinionated text, may prove to be a useful approach.

A different aspect of retrieval in blogs, that we did not address here, is the need to identify *new* relevant information—posts containing content that the searcher did not see in previous searches. Here, personalized retrieval techniques (to store the user's history) can be combined with content similarity approaches to remove redundant content in the search results.

Beyond Blogs

Blogs are one form of user-generated content; but other types of social media exist, including popular services for sharing photos, videos, bookmarks, and other types of information. In particular, social-networking websites such as MySpace or Facebook have proliferated in recent years. In addition to blogging facilities, social-networking sites typically provide means for publishing and sharing multimedia content, as well as community features such as groups and discussion facilities. These services are, in a way, a community-oriented extension of personal blogs: they focus on the networks created between people and the interactions within them, but still provide a separate space to each user and ways of expressing his identity through it.

Many of the text analysis techniques we developed apply also to other forms of social media, and, in particular, to social-networking ones. In some cases—e.g., comment spam filtering based on language model divergence—the methods can be applied as-is in these new domains. In other cases, extending our work to match new types of content is straightforward. Possible extensions include:

- **Profiles.** The methods we used to profile blogs and to match products and advertisements with bloggers will greatly benefit from the richer personal information found on social-networking sites. These sites may also have more detailed—and sincere—self-supplied profiles, which can be used to construct new classification tasks, such as classifying by profession.
- **Collaborative recommendations.** Tag recommendation through a collaborative process is possible not only for blog posts, but also for other content types: for example, given a method to compare URLs (e.g., through

a comparison the contents of their pages), we can use the same method to recommend tags for bookmarks in bookmark-sharing services.

- **Search.** Usage of some of the retrieval methods we developed in non-blog domains may also be beneficial: for example, reranking retrieval results by the recency methods we proposed, or by the number of comments in response to the document (which can be an audio or video clip).

Similarly, extensions can be designed for the sentiment analysis methods we used—either at the single user level, or at the aggregate level. But the complexity and richness of social-networking sites bring also new challenges to the area of text analytics, from technical ones relating to scale and robustness, to more conceptual ones, such as identifying user behavior and needs in these domains. We have shown in this dissertation that blogs are substantially different from other forms of web pages; we believe social-networking sites are different too, in additional ways.

Lastly, some of the work presented here applies not only to blogs and social media, but to broader domains. The spam-related work we developed can be used in other settings: indeed, our language-divergence based approach to spam filtering has been successfully adopted to filter web spam [30]. In the retrieval domain, both the query-specific spam weighing approach we developed for the opinion retrieval task and the temporal relevance feedback method we use for recency scoring seem applicable to other search tasks. Our approach to query categorization applies also to web queries; similar approaches, developed in parallel to ours, have been successfully used for web query categorization (e.g., at the 2005 KDD Cup [140]). Finally, the general framework we use to track global mood changes—detecting an irregularity by comparing expected and actual values, then automatically annotating the irregularity by identifying different language use in it, and relating it to external resources—can be used also to mine non-sentiment related irregularities in temporal data associated with text, such as financial information (e.g., stock prices with their associated news sources).

Is the blogspace—an interconnected collection of personal, informal writings—a viable domain for computational text analysis methods? Our answer is positive: computational text mining approaches expose useful, and sometimes surprising, information about blogs and about the people behind them. As the volume of user-generated content is expected to continue to grow in the foreseeable future, the amount of knowledge that can be identified in it through text analytics will increase too—as, we expect, will be its importance.

Appendix A

Crawling Blogs

Typically, collections of web pages used for research purposes are built using a crawler—an agent that traverses the web by starting from a set of seed URLs and following hyperlinks on every page it visits. To crawl a set of web pages whose contents are continuously updated—such as home pages of blogs—the crawling process is repeated at regular intervals, or at intervals learned from the update rate of a page.

In earlier days of computational access to the blogspace, this general web crawling method was used also for blog posts (see, e.g., [89, 217]). However, this approach is not optimal for collecting blog data for a number of reasons. First, the high update rate of blogs requires a low-latency crawling process, which is not guaranteed by traditional crawling architectures. More importantly, traditional crawling ignores two features of blogs which can improve the crawling process significantly. As described in Section 2.1.2, most blogs—unlike most regular web pages—issue a notification when their content has changed (a "ping"), and therefore do not need to be re-visited unless such a notification has been issued. Additionally, the contents of most blogs are syndicated in a format which is more conveniently processed by machines (i.e., RSS or Atom), preventing the need to crawl the entire blog page, which often includes multiple posts as well as additional, static data. These two features enable significant savings on bandwidth and computation time, by crawling only the relevant part of blog data—and only when needed. Collecting blog posts is done more effectively, then, by using a specialized agent which takes the properties of the blogspace into account; commercial blog search platforms have moved towards this direction. This appendix describes such a blog crawler which was built for collecting and storing some of the data used in this thesis.

The design of our blog crawler is based on following information about updates to blog pages, rather than actively traversing links in web pages as traditional crawlers do. Once an update is registered by the crawler, it fetches its contents—in syndicated form first, and, if needed, in HTML format too. A high-level diagram of the architecture of the crawler appears in Figure A.1; the upper part contains an overview of the entire crawler, and the lower part zooms in on one of the components which has a more complex internal structure. We follow with details about each of the components.

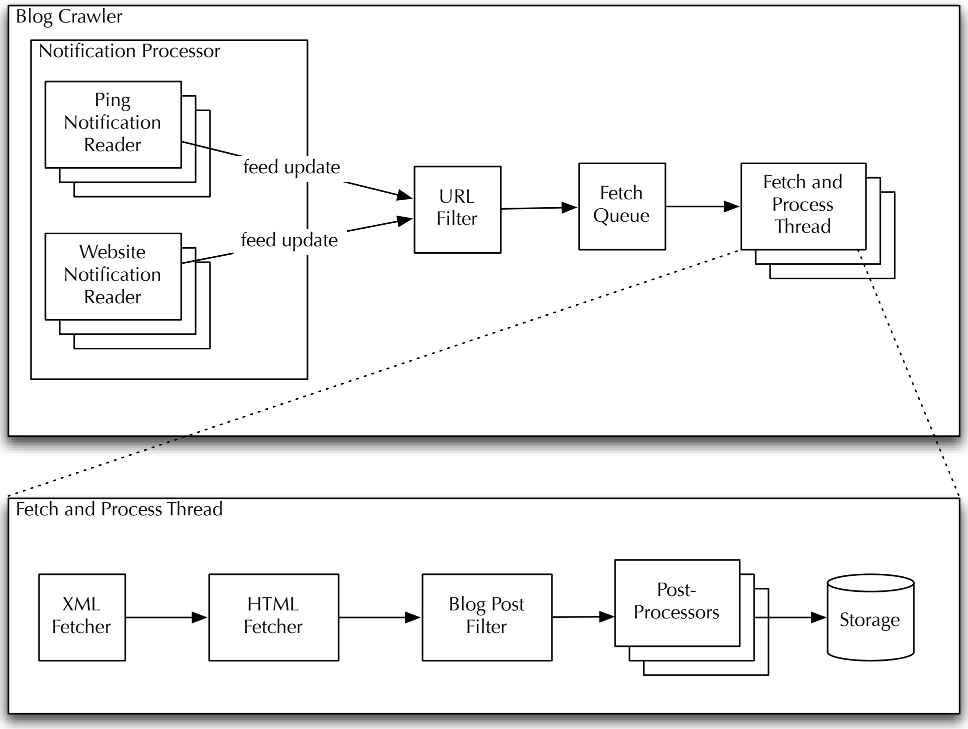


Figure A.1: Architecture of the blog crawler.

- **Notification Processor**. The purpose of this component is to track updates to blogs by following known sources of information about such changes. There are various indications that a blog has been updated. Some blog hosting platforms, for example, list recently updated blogs; examples are Blogger's "Recently Updated" list[1] or the "Recently Updated Weblogs" section in the Typepad homepage.[2] Another source is ping aggregators: services which are notified by various bloggers once their blog has been updated (usually, this is done by the blogging software automatically), then redistribute the combined updates publicly. Ping aggregators can either distribute the list of changes through a web page which is frequently updated (e.g., Weblogs.com[3]), or actively, through a streaming mechanism (e.g., Feedmesh[4] or blo.gs[5]). Our crawler polls "recently updated" lists and

[1] http://www.blogger.com/changes.g
[2] http://www.typepad.com
[3] http://rpc.weblogs.com/shortChanges.xml
[4] sandbox.pubsub.com:9999, offline as of early 2007
[5] blo.gs:29999

web page based ping aggregators, as well as accepts blog update notifications through various streaming services. A simple de-duping mechanism handles duplicate notifications for the same blog, which tend to appear in multiple data sources with short delays.

- **URL Filter**. Notifications about blog updates are passed though a set of blog URL filters, which determine if the update should be handled by the crawler or discarded. Available filters include regular-expression based ones (useful for naive filtering of non-English updates by ignoring blogs hosted by largely non-English domains such as .jp or .br), black-list or white-list filters (ignoring all blogs except a list of permitted ones, or accepting all blogs from a list of known ones), and random-sampling filters. The filters used for a given crawl depend on the specific task at hand and are user-configurable; notifications which pass all filters are inserted into a **Fetch Queue**, a FIFO queue of requests waiting to be handled (each request contains the URL of the updated blog as well as some meta-information such as the update date and time).

- **Fetch and Process Threads**. These threads perform the bulk of the crawler's workload: emptying the Fetch Queue and processing each request. A multiple-thread architecture is common in crawlers, allowing for parallel downloading and processing of multiple URLs; often, the network usage associated with each fetch request is the bottleneck in the process, and usage of more than one thread substantially increases the throughput of a crawler. Usage of threads allow also handling politeness requirements from crawlers: many web sites limit the maximal number of fetch requests in a given time period or the number of parallel requests, and using threads to fetch from multiple domains concurrently increases throughput. Each thread executes the following instruction pipeline, illustrated in the lower part of Figure A.1.

  - First, the next blog update notification is picked up and removed from the Fetch Queue.
  - Next, the syndicated contents of the updated blog (i.e., the RSS or Atom feed) are fetched. Typically, the syndicated version of a blog contains multiple posts—the recent n entries in the blog; at this stage, the latest post in the feed, the one with a timestamp matching the update notification, is extracted and stored separately.
  - As noted in Section 9.3.2, some blogs syndicate only part of the full contents of their posts; estimates place the percentages of such blogs at 11% [87]. In these cases, to obtain the missing content, the corresponding HTML permalink (which is part of the update notification) is fetched, and the post is extracted from it. When extracting the

post, the crawler makes use of the fact that it already has the partial contents of it, to identify the location of post inside the text of the HTML page. The decision whether a syndicated version of a post is partial and requires fetching the HTML contents is taken using a set of heuristics, including testing whether the syndicated content contains HTML markup (typically, a strong indicator of full syndication); checking the length of the syndicated post (long posts are unlikely to be partial); testing for the presence of ellipsis at the end of a post, and so on.

- Once the content of the post has been extracted, a set of content filters is applied to it. Implemented filters include language-based ones (for selecting only posts of a given language, if this is required by the parameters of the crawl, we use the language-guessing module described in [48]), and spam filters (we use a filter similar to the one described in Section 9.2, but trained on additional data).
- Finally, the posts which pass the previous filtering stage are forwarded to a set of processors for analyzing and storing them. Depending on the task, processors include linguistic tools such as a named-entity or part-of-speech tagger, as well as an inverted indexer for retrieval purposes. Like the set of filters, the set of processors used can be configured in the parameters of the crawl.

Once the post-processing is done, the thread starts over, picking the next fetch request from the queue.

Note that as the crawler fetches posts as soon as they are published (or, more accurately, as soon as a notification about their publication is made public), it typically does not collect the post comments; these are posted in varying delays after the publication of the post. To account for this, and especially since we have shown the usefulness of comments in several places in this thesis, our crawler contains a modification allowing comment access. Once a post has been extracted, the Fetch and Process Thread may—if configured to do so—add an additional HTML fetch request for the same URL to the Fetch Queue, but with a temporal delay: this will cause a re-fetch of the HTML after a given period (e.g., 24 or 72 hours). When a post is fetched due to such a request rather than the standard one, a special post-processor is used to separate the "new" content from that already stored; this new content is assumed to consist of comments and trackbacks.

The blog crawler is limited only by bandwidth, and was successfully used to retrieve posts at a daily rate of 1GB of text on a single desktop machine, including full processing (English named entity and part-of-speech tagging as well as indexing).

Appendix B

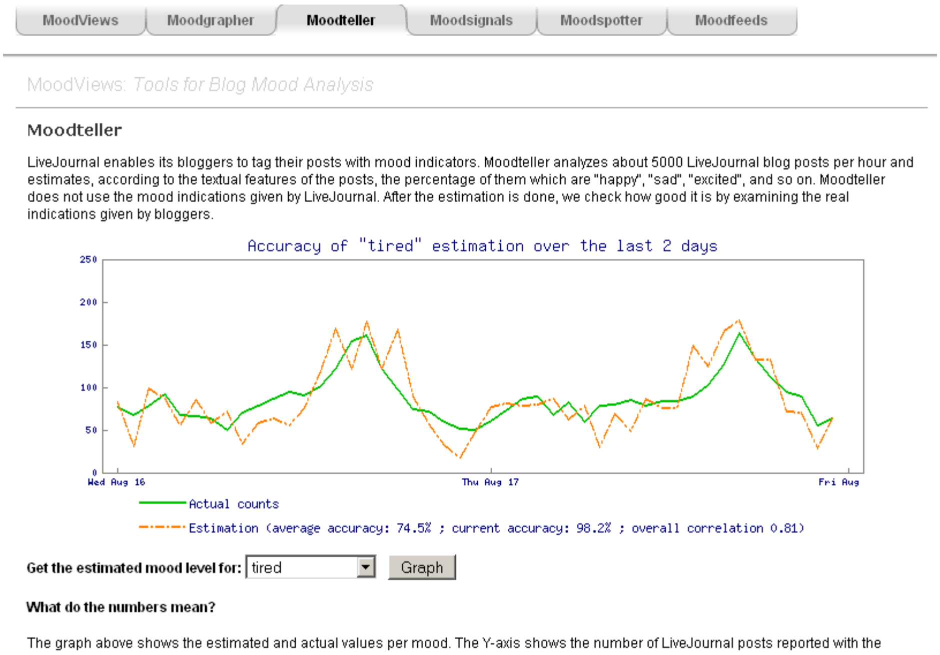
MoodViews: Tools for Blog Mood Analysis

MoodViews is a platform for collecting, analyzing, and displaying aggregate moods in the blogspace; it was used for some of the work described in Chapter 6. This appendix describes its internal workings.

As mentioned in Chapter 6, some blogging platforms (e.g., LiveJournal and MySpace) allow bloggers to assign a mood descriptor to their blog posts. The post is then published with an indicator of the "current mood" of the blogger, at the time of posting the blog. MoodViews continuously collects these mood indications, as well as the blog posts themselves, and provides a number of services based on these. More specifically, the data collected is all public LiveJournal posts; at the time MoodViews was created, it was tracking about 100,000 posts every day, but the amount has decreased since to around 50,000 daily posts. In total, in December 2006, MoodViews provides access to more than 35 million blog posts.

The services provided by MoodViews based on the mood indicators on the blog posts are as follows.

- *Moodgrapher* displays the aggregate counts of the different moods in the LiveJournal data, plotting them over time. This gives users an indication of current trends of moods among bloggers, and allows observations such as those made in Section 6.2 regarding the different cyclic nature of some moods. The sample plots in Figures 6.2 and 6.3 were generated by Moodgrapher.
- *Moodteller* is an online implementation of the mood level prediction method described in Section 6.3; it continuously estimates mood levels in the blogspace, shows plots of these estimates, and compares them to the actual, reported mood levels. An example plot generated by Moodteller is shown in Figure B.1.
- *Moodsignals* detects irregular patterns of mood levels, and attempts to pro-


MoodViews
Moodgrapher
Moodteller
Moodsignals
Moodspotter
Moodfeeds
MoodViews: *Tools for Blog Mood Analysis*
Accuracy of "tired" estimation over the last 2 days
250
200
150
100
50
0
Wed Aug 16
Thu Aug 17
Fri Aug
Actual counts
Estimation (average accuracy: 74.5% ; current accuracy: 98.2% ; overall correlation 0.81)
Graph

Moodteller

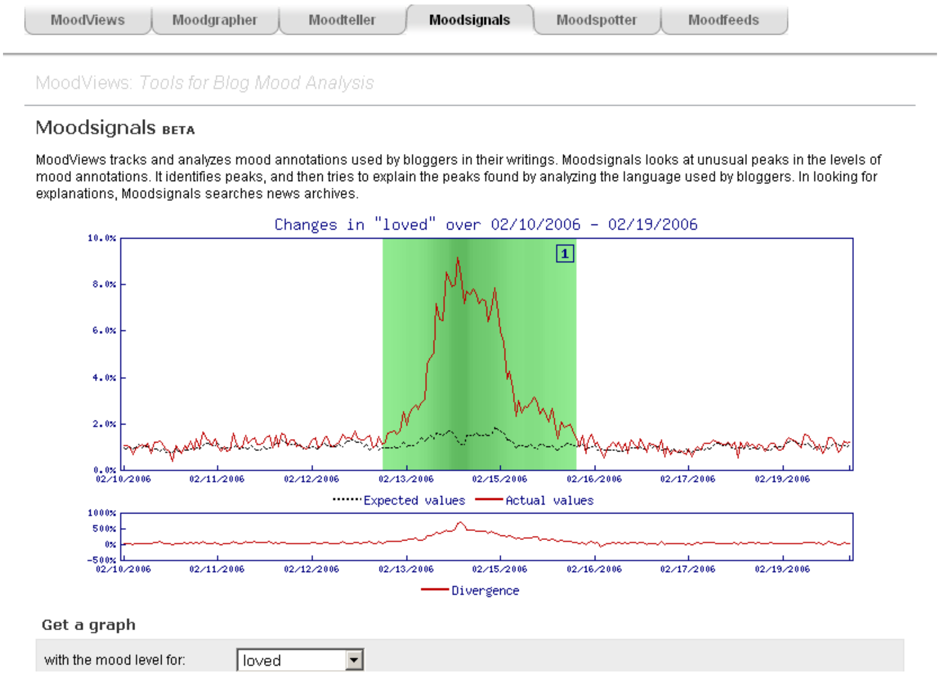
LiveJournal enables its bloggers to tag their posts with mood indicators. Moodteller analyzes about 5000 LiveJournal blog posts per hour and estimates, according to the textual features of the posts, the percentage of them which are "happy", "sad", "excited", and so on. Moodteller does not use the mood indications given by LiveJournal. After the estimation is done, we check how good it is by examining the real indications given by bloggers.

Get the estimated mood level for: tired

What do the numbers mean?

The graph above shows the estimated and actual values per mood. The Y-axis shows the number of LiveJournal posts reported with the

Figure B.1: MoodViews screen captures: Moodteller.



Moodsignals BETA

MoodViews tracks and analyzes mood annotations used by bloggers in their writings. Moodsignals looks at unusual peaks in the levels of mood annotations. It identifies peaks, and then tries to explain the peaks found by analyzing the language used by bloggers. In looking for explanations, Moodsignals searches news archives.

Get a graph

with the mood level for: loved

Figure B.2: MoodViews screen captures: MoodSignals.

vide a natural-language explanation for them; it is an online implementation of the method described in Section 6.4.

- *Moodspotter* associates between terms and moods, displaying the moods which are most correlated with terms selected by the user, again plotting these over time.

The architecture of MoodViews is shown in Figure B.3; it is composed of a separate backend and frontend. The MoodViews backend collects and stores the data, and performs the analysis and calculations needed for the various services; the frontend consists of web-based interfaces to the services.

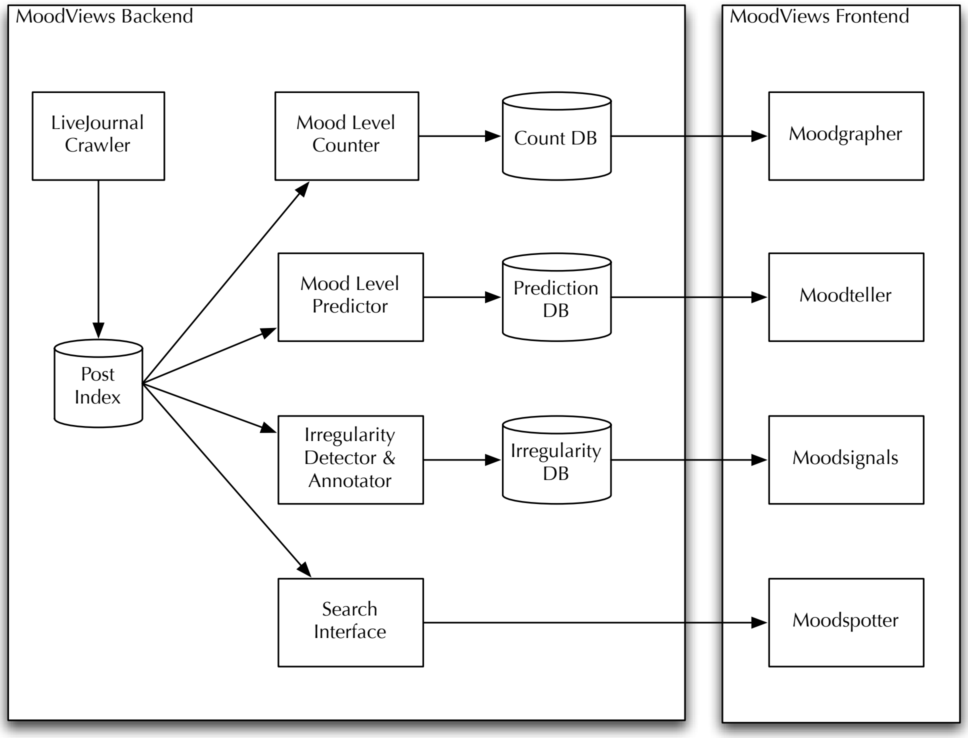


Figure B.3: Architecture of MoodViews.

Within the MoodViews backend, the LiveJournal Crawler periodically polls a stream of "recently published posts," parses the text, date, and mood indicators, and stores them.[1] A series of analysis modules then performs, in parallel, the computations required for each one of the services; these are updated every few minutes to keep data fresh, and stored in dedicated databases which are queries by the frontend components.

For Moodgrapher, a Mood Level Counter stores the total amount of posts tagged with each mood from a predefined set of 132 moods, their percentages out of the total number of mood-annotated posts, and the rate of change in the count of each mood. A Mood Level predictor performs, similarly, the calculations for

[1] The stream is found at `http://www.livejournal.com/stats/latest-rss.bml`

Moodteller: it stores the term counts for all terms in the set of indicative terms described in Section 6.3, trains linear models for predicting the mood levels from these based on data from the previous 24 hours, and uses the models to store an estimated level for each one of the moods. For the Moodsignals service, an Irregularity module implements Section 6.4 of this thesis, identifying recent time periods for which the actual mood counts diverge substantially from the estimated ones, extracting the indicative terms for these time periods, and querying a collection of events from Wikinews—which is fetched and indexed separately—to provide an explanation for the irregularity.

An indication to the usefulness users find in the services offered by MoodViews can be found in the amount of attention it attracts in the blogspace and beyond: within a period of a few months, it was used by more than 100,000 unique visitors and referenced in hundreds of blog posts, as well as in mainstream media sources.[2]

[2] Highlights of the coverage of MoodViews in the press are available at http://moodviews.com/Press.

Samenvatting

Het World Wide Web heeft invloed op vele facetten van ons leven: op ons werk en onze vrije tijd, en ook op onze manier van communiceren. Het web waaraan wij zo gewend geraakt zijn is sinds jaren echter aan een verandering onderhevig. De vooruitgang in de technologie en de eenvoudige toegang tot het internet, gecombineerd met een generatie die opgegroeid is met dit altijd aanwezige web, hebben een fenomeen voortgebracht dat bekend is onder de term *user-generated content*: door gebruikers gegenereerde inhoud. Deze inhoud is gemaakt door de gebruikers van websites en niet door professionele website beheerders: iedereen kan zijn of haar bijdrage leveren. In 2006 koos éénderde van de internet gebruikers ervoor om niet alleen internet inhoud te consumeren, maar ook te produceren. Een specifieke vorm van deze door gebruikers gegenereerde inhoud zijn de zogenaamde *blogs*. Dit zijn persoonlijke webpagina's die periodiek worden vernieuwd en vaak als dagboeken dienen. Het bijhouden van een dagboek is op zichzelf niets nieuws, maar het blogging fenomeen is uniek omdat de dagboeken publiekelijk beschikbaar zijn en zo de levens van miljoenen individuen over de hele wereld blootleggen. De *blogspace*—de verzameling van alle blogs—verschilt van andere grote datacollecties op meerdere niveaus. Het meest in het oog springend is de persoonlijke aard van de inhoud. De tekst in blogs bevat vaak beschrijvingen van het leven van een persoon en zijn of haar leefomgeving, en daarnaast gedachten, emoties, en commentaren op verschillende onderwerpen. Dergelijke inhoud is zeldzaam in andere publiekelijk beschikbare corpora.

De beschikbaarheid van dergelijke inhoud biedt nieuwe uitdagingen op het gebied van de tekstanalyse—een interdisciplinair onderzoeksgebied dat een reeks van methoden voor het ontdekken van kennis in ongestructureerde tekst omvat, en waar technieken gecombineerd worden uit disciplines zoals de computationele taalkunde, information retrieval en machineleren. Dit proefschrift opereert binnen dit raamwerk van de tekstanalyse met als doel om kennis te identificeren die kan worden gevonden in blogs. Hierbij maken wij zowel gebruik van bestaande methoden, als ook van nieuwe methoden die door ons zijn ontwikkeld. De voornaamste vraag die we proberen te beantwoorden is: *hoe kunnen de karakteristieke eigenschappen van blogs worden gebruikt om effectief kennis te vergaren uit de blogspace?* Meer specifiek beogen wij twee verschillende soorten informatie te identificeren. Ten eerste, de feitelijke informatie rondom een persoon. Hierbij gaat het om de interesses, culturele voorkeuren, en leefomgeving van een individu. Het tweede type informatie dat we zoeken is niet-feitelijke informatie: emoties, meningen, en stemmingen. Dit is de informatie die grotendeels uniek is voor de blogspace.

We laten zien dat deze twee soorten van kennis inderdaad effectief vergaard kunnen worden uit blogs. Hierbij maken we gebruik van een reeks methoden, veelal gebaseerd op statististische taalmodellen en tekstclassificatie. We laten ook zien dat deze vorm van informatievergaring uit een groot aantal blogs (gebruikmakend van tekstanalyse) informatie blootlegt die niet beschikbaar is in individuele blogs. Voorbeelden zijn bepaalde globale patronen en onderwerpen die leiden tot publieke debatten. In een ander deel van dit proefschrift onderzoeken wij het zoekgedrag in de blogspace: wat zoeken gebruikers eigenlijk in blogs, en hoe kan effectief op hun behoefte gereageerd worden? We ontdekken dat gebruikers andere informatie zoeken in blogs dan in andere web omgevingen. Hierdoor zal er ook anders op hun eisen gereageerd moeten worden. Tot slot: wij noemen het werk dat hier wordt beschreven *toegepaste tekstanalyse*, omdat elk voorgesteld algoritme getest wordt in een realistische omgeving, met een concrete taak, echte data, en (waar mogelijk) vergeleken wordt met de *state-of-the-art*.

We geloven dat het fenomeen van door gebruikers gegenereerde inhoud zal blijven bestaan, en dat dit fenomeen de komende jaren een blijvend en substantieel effect zal hebben op het web. Terwijl de vorm van blogs mogelijk zal veranderen, kunnen vele van de in deze dissertatie ontwikkelde methoden worden uitgebreid tot andere vormen van door gebruikers gegenereerde inhoud. Dit geldt specifiek voor toepassingen in sociale netwerken die sinds kort zeer populair zijn. Nu steeds meer persoonlijk georiënteerde informatie online beschikbaar komt, zal het belang van de hier besproken tekstanalyse methoden verder toenemen. Zowel de producent als de consument van webinhoud kan deze kennis gebruiken om verborgen informatie te organiseren en beschikbaar te maken.

Bibliography

[1] Lada A. Adamic. The small world web. In *ECDL '99: Proceedings of the Third European Conference on Research and Advanced Technology for Digital Libraries*, pages 443–452, London, UK, 1999. Springer-Verlag.

[2] Lada A. Adamic and Natalie Glance. The political blogosphere and the 2004 U.S. election: Divided they blog. In *WWW 2005 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[3] Eytan Adar and Lada A. Adamic. Tracking information epidemics in blogspace. In *2005 IEEE / WIC / ACM International Conference on Web Intelligence (WI 2005)*, pages 207–214, 2005.

[4] Eytan Adar, Li Zhang, Lada A. Adamic, and Rajan M. Lukose. Implicit structure and the dynamics of blogspace. In *WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '04: the 13th international conference on World Wide Web*, 2004.

[5] Navot Akiva and Jonathan Schler. TrendMine: Utilizing authorship profiling and tone analysis in context. In *Workshop on Stylistics for Text Retrieval in Practice at SIGIR '06: the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 2006.

[6] Reka Albert, Hawoong Jeong, and Albert-Laszlo Barabasi. The diameter of the world wide web. *Nature*, 401:130–131, 1999.

[7] Noor Ali-Hasan and Lada Adamic. Expressing social relationships on the blog through links and comments, 2006. Available online at http://www-personal.umich.edu/~ladamic/papers/oc/onlinecommunities.pdf, accessed August 2006.

[8] James Allan. *Topic Detection and Tracking: Event-Based Information Organization*. Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[9] Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.*, 20(4):357–389, 2002.

[10] Einat Amitay, David Carmel, Adam Darlow, Ronny Lempel, and Aya Soffer. The connectivity sonar: detecting site functionality by structural patterns. In *HYPERTEXT '03: Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 38–47. ACM Press, 2003.

[11] Chris Anderson. The Long Tail. *WIRED magazine*, October 2004.

[12] Ion Androutsopoulos, John Koutsias, Konstantinos V. Chandrinos, and Constantine D. Spyropoulos. An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages. In *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 160–167. ACM Press, 2000.

[13] Anjo Anjewierden and Lilia Efimova. Understanding weblog communities through digital traces: a framework, a tool and an example. In *International Workshop on Community Informatics (COMINF 2006), OTM Federated Conferences*, 2006.

[14] Shlomo Argamon, Sushant Dhawle, Moshe Koppel, and James Pennebaker. Lexical predictors of personality type. In *Joint Annual Meeting of the Interface and the Classification Society of North America*, 2005.

[15] Giuseppe Attardi and Maria Simi. Blog mining through opinionated words. In *TREC*, Gaithersburg, Maryland USA, 2006.

[16] Paolo Avesani, Marco Cova, Conor Hayes, and Paolo Massa. Learning contextualised weblog topics. In *WWW 2005 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[17] Leif Azzopardi, Mark Girolami, and Keith van Rijsbergen. Investigating the relationship between language model perplexity and ir precision-recall measures. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 369–370, New York, NY, USA, 2003. ACM Press.

[18] Nofollow tag cheers bloggers, but fails blogs, URL: http://www.platinax.co.uk/news/archives/2005/01/new_nofollow_ta.html, accessed June 2005.

[19] Nofollow No Good? URL: http://jeremy.zawodny.com/blog/archives/006800.html, accessed September 2006.

[20] Ricardo Baeza-Yates and Emilio Davis. Web page ranking using link attributes. In *WWW '04: Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 328–329. ACM Press, 2004.

[21] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

[22] Pierre Baldi, Paolo Frasconi, and Padhraic Smyth. *Modeling the Internet and the Web: Probabilistic Methods and Algorithms*. Wiley, 2003.

[23] Krisztian Balog, Gilad Mishne, and Maarten de Rijke. Why are they excited? identifying and explaining spikes in blog mood levels. In *Proceedings 11th Meeting of the European Chapter of the Association for Computational Linguistics (EACL 2006)*, 2006.

[24] Judit Bar-Ilan. On the overlap, the precision and estimated recall of search engines. a case study of the query "erdos". *Scientometrics*, 42(2):207–228, June 1998.

[25] Judit Bar-Ilan. An outsider's view on "topic-oriented blogging". In *WWW '04: Proceedings of the 13th international World Wide Web conference, Alternate track papers & posters*, pages 28–34, New York, NY, USA, 2004. ACM Press.

[26] Albert-Laszlo Barabasi and Reka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

[27] Jorn Barger. Weblog resources FAQ – what is a weblog?, 1999. http://www.robotwisdom.com/weblogs, accessed June 2006.

[28] Luca Becchetti, Carlos Castillo, Debora Donato, Stefano Leonardi, and Ricardo Baeza-Yates. Using rank propagation and probabilistic counting for link-based spam detection. Technical report, DELIS – Dynamically Evolving, Large-Scale Information Systems, 2006.

[29] Steven M. Beitzel, Eric C. Jensen, Abdur Chowdhury, David Grossman, and Ophir Frieder. Hourly analysis of a very large topically categorized web query log. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, 2004.

[30] András A. Benczúr, István Bíró, Károly Csalogány, and Máté Uher. Detecting nepotistic links by language model disagreement. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 939–940, 2006.

[31] Andras A. Benczúr, Károly Csalogány, Tamás Sarlós, and Máté Uher. SpamRank – fully automatic link spam detection. In *First International Workshop on Adversarial Information Retrieval on the Web, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[32] Bettina Berendt and Roberto Navigli. Finding your way through blogspace: Using semantics for cross-domain blog analysis. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[33] Hemant K. Bhargava and Juan Feng. Paid placement strategies for internet search engines. In *WWW '02: Proceedings of the 11th intern. conf. on World Wide Web*, pages 117–123. ACM Press, 2002.

[34] Pete Blackshaw and Mike Nazzaro. Consumer-generated media 101: Word-of-mouth in the age of the web-fortified consumer. Nielsen BuzzMetrics white paper, 2004.

[35] Anita Blanchard. Blogs as virtual communities: Identifying a sense of community in the Julie/Julia project. In L. Gurak, S. Antonijevic, L. Johnson, C. Ratliff, and J. Reyman, editors, *Into the Blogosphere; Rhetoric, Community and Culture of Weblogs*, 2004.

[36] Anita L. Blanchard and M. Lynne Markus. The experienced "sense" of a virtual community: characteristics and processes. *SIGMIS Database*, 35(1):64–79, 2004.

[37] Rebecca Blood. Weblogs: a history and perspective, 2000. http://www.rebeccablood.net/essays/weblog_history.html, accessed July 2006.

[38] Rebecca Blood. *The Weblog Handbook: Practical Advice on Creating and Maintaining Your Blog*. Perseus Publishing, 2002.

[39] Thorsten Brants. TnT – a statistical part-of-speech tagger. In *Proceedings of the Sixth Applied Natural Language Processing (ANLP-2000)*, Seattle, WA, 2000.

[40] Andrei Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, 2002.

[41] Christopher H. Brooks and Nancy Montanez. Improved annotation of the blogosphere via autotagging and hierarchical clustering. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 625–632. ACM Press, 2006.

[42] Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. Class-based n-gram models of natural language. *Comput. Linguist.*, 18(4):467–479, 1992.

[43] Peter F. Brown, Vincent J. Della Pietra, Robert L. Mercer, Stephen A. Della Pietra, and Jennifer C. Lai. An estimate of an upper bound for the entropy of English. *Comput. Linguist.*, 18(1):31–40, 1992.

[44] Ralf Brown and Robert Frederking. Applying statistical English language modeling to symbolic machine translation. In *Proceedings of the Sixth International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-95)*, pages 221–239, July 1995.

[45] John D. Burger and John C. Henderson. An exploration of observable features related to blogger age. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[46] Lou Burnard. *User Reference Guide for the British National Corpus*. Oxford University, 2000.

[47] *AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006.

[48] William B. Cavnar and John M. Trenkle. N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, 1994.

[49] Soumen Chakrabarti. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann, August 2002.

[50] Stanley F. Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318, Morristown, NJ, USA, 1996. Association for Computational Linguistics.

[51] Paula Chesley, Bruce Vincent, Li Xu, and Rohini Srihari. Using verbs and adjectives to automatically classify blog sentiment. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[52] Yun Chi, Junichi Tatemura, and Belle Tseng. Eigen-Trend: Trend analysis in the blogosphere based on singular value decompositions. In *CIKM '06: Proceedings of the fifteenth international conference on Information and knowledge management*, New York, NY, USA, 2006. ACM Press.

[53] Alvin Chin and Mark Chignell. A social hypertext model for finding community in blogs. In *HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia*, pages 11–22, New York, NY, USA, 2006. ACM Press.

[54] Charles Clarke, Nick Craswell, and Ian Soboroff. The TREC terabyte retrieval track. *SIGIR Forum*, 39(1):25–25, 2005.

[55] Tom Coates. On permalinks and paradigms, 2003. http://www.plasticbag.org/archives/2003/06/on_permalinks_and_paradigms, accessed September 2006.

[56] Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. Wiley-Interscience, New York, NY, USA, 1991.

[57] Nick Craswell, Arjen P. de Vries, and Ian Soboroff. Overview of the TREC-2005 Enterprise Track. In *TREC*, Gaithersburg, Maryland USA, 2005.

[58] Nick Craswell and David Hawking. Overview of the TREC-2004 Web Track. In *TREC*, Gaithersburg, Maryland USA, 2004.

[59] Nick Craswell, David Hawking, and Stephen Robertson. Effective site finding using link anchor information. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 250–257, New York, NY, USA, 2001. ACM Press.

[60] Nick Craswell, Stephen Robertson, Hugo Zaragoza, and Michael Taylor. Relevance weighting for query independent evidence. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 416–423, 2005.

[61] Hang Cui, Vibhu O. Mittal, and Mayur Datar. Comparative experiments on sentiment classification for online product reviews. In *The Twenty-First National Conference on Artificial Intelligence (AAAI-06)*, 2006.

[62] Honghua (Kathy) Dai, Lingzhi Zhao, Zaiqing Nie, Ji-Rong Wen, Lee Wang, and Ying Li. Detecting online commercial intention (OCI). In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 829–837, 2006.

[63] Daniel W. Drezner and Henry Farrell. The power and politics of blogs, 2004. Available online at http://www.danieldrezner.com/research/blogpaperfinal.pdf, accessed May 2006.

[64] Sanjiv Das and Mike Chen. Yahoo! for Amazon: Sentiment parsing from small talk on the web. In *Proceedings EFA 2001*, 2001.

[65] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 519–528, New York, NY, USA, 2003. ACM Press.

[66] David Sifry. Sifry's Alerts (blog), http://www.sifry.com/alerts/, accessed May 2006.

[67] David Sifry. State of the blogosphere, april 2006, part 2: On language and tagging. http://www.sifry.com/alerts/archives/000433.html, accessed September 2006.

[68] Brian Davison. Recognizing nepotistic links on the web. In *AAAI-2000 workshop on Artificial Intelligence for Web Search*, pages 23–28. AAAI Press, 2000.

[69] Aldo de Moor and Lilia Efimova. An argumentation analysis of weblog conversations. In *The 9th International Working Conference on the Language-Action Perspective on Communication Modelling (LAP 2004)*, 2004.

[70] The Berkeley/Stanford Web Term Document Frequency and Rank project, URL: http://elib.cs.berkeley.edu/docfreq/, accessed October 2005.

[71] Christine Doran, John D. Griffith, and John Henderson. Highlights from 12 months of collected blogs. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[72] Helen S. Du and Christian Wagner. Success in the blogosphere: Exploring the role of technology. In *PACIS 2005:The 9th Pacific-Asia Conference on Information Systems*, 2005.

[73] Ted Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[74] Lilia Efimova and Aldo de Moor. Beyond personal webpublishing: An exploratory study of conversational blogging practices. In *HICSS '05: Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 4*, page 107.1, Washington, DC, USA, 2005. IEEE Computer Society.

[75] Lilia Efimova, Stephanie Hendrick, and Anjo Anjewierden. Finding "the life between buildings": An approach for defining a weblog community. In *AOIR Internet Research 6.0: Internet Generations*, 2005.

[76] Koji Eguchi and Victor Lavrenko. Sentiment retrieval using generative models. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, pages 345–354, July 2006.

[77] Federico Michele Facca and Pier Luca Lanzi. Mining interesting knowledge from weblogs: a survey. *Data & Knowledge Engineering*, 53(3):225–241, 2005.

[78] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On power-law relationships of the internet topology. In *SIGCOMM '99: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, pages 251–262, New York, NY, USA, 1999. ACM Press.

[79] Dennis Fetterly, Mark Manasse, and Marc Najork. Spam, damn spam, and statistics: using statistical analysis to locate spam web pages. In *WebDB '04: Proceedings of the 7th International Workshop on the Web and Databases*, pages 1–6. ACM Press, 2004.

[80] Ko Fujimura, Takafumi Inoue, and Masayuki Sugisaki. The EigenRumor algorithm for ranking blogs. In *WWW 2005 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[81] Tomohiro Fukuhara, Toshihiro Murayama, and Toyoaki Nishida. Analyzing concerns of people using weblog articles and real world temporal data. In *WWW 2005 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[82] William A. Gale and Geoffrey Sampson. Good-Turing frequency estimation without tears. *Journal of Quantitative Linguistics*, 2(3):217–237, 1995.

[83] Michael Gamon, Anthony Aue, Simon Corston-Oliver, and Eric K. Ringger. Pulse: Mining customer opinions from free text. In *IDA*, pages 121–132, 2005.

[84] Michel Gènèreux and Roger Evans. Distinguishing affective states in weblogs. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[85] Kathy Gill. How can we measure the influence of the blogosphere? In *WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '04: the 13th international conference on World Wide Web*, 2004.

[86] Kathy E. Gill. Blogging, RSS and the information landscape: A look at online news. In *WWW 2005 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[87] Natalie Glance. Indexing the blogosphere one post at a time. In *Third International Workshop on Web Document Analysis (WDA2005)*, 2005.

[88] Natalie Glance, Matthew Hurst, Kamal Nigam, Matthew Siegler, Robert Stockton, and Takashi Tomokiyo. Deriving marketing intelligence from online discussion. In *KDD '05: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, 2005.

[89] Natalie Glance, Matthew Hurst, and Takashi Tomokiyo. BlogPulse: Automated trend discovery for weblogs. In *WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '04: the 13th international conference on World Wide Web*, 2004.

[90] Nathalie Glance, Damian Arregui, and Manfred Dardenne. Knowledge Pump: Supporting the flow and use of knowledge. In D.K. Holtshouse, Uwe M. Borghoff, and Remo Pareschi, editors, *Information Technology for Knowledge Management*. Springer Verlag, 1998.

[91] Scott Golder and Bernardo A. Huberman. The structure of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208, 2006.

[92] José María Gómez-Hidalgo. Evaluating cost-sensitive unsolicited bulk email categorization. In *Proceedings of SAC-02, 17th ACM Symposium on Applied Computing*, 2002.

[93] Eugene Gorny. Russian livejournal: The national specifics in the development of a virtual community. In *Internet Research 5.0*, 2004.

[94] Gregory Grefenstette, Yan Qu, James G. Shanahan, and David A. Evans. Coupling niche browsers and affect analysis. In *RIAO'2004*, 2004.

[95] Lev Grossman. Cover story: Person of the year. *Time Magazine*, 168(26), 2006.

[96] China Market Research Group. The blogging point, 2005.

[97] Daniel Gruhl, Ramanathan Guha, Ravi Kumar, Jasmine Novak, and Andrew Tomkins. The predictive power of online chatter. In *KDD '05: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 78–87. ACM Press, 2005.

[98] Daniel Gruhl, Ramanathan Guha, David Liben-Nowell, and Andrew Tomkins. Information diffusion through blogspace. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 491–501, 2004.

[99] Lei Gu, Tom Lento, Marc Smith, and Paul Johns. How do blog gardens grow? language community correlates with network diffusion and adoption of blogging systems. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[100] Michelle Gumbrecht. Blogs as "protected space". In *WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '04: the 13th international conference on World Wide Web*, 2004.

[101] Robert Gunning. *The technique of clear writing.* McGraw-Hill, 1952.

[102] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating web spam with TrustRank. In *Thirtieth International Conference on Very Large Data Bases (VLDB 2004)*, pages 576–587, 2004.

[103] Katie Hafner. For some, the blogging never stops. *The New York Times*, May 27 2004.

[104] Alex Halavais. Collaborative web publishing as a technology and a practice. In J. Nolan J. Weiss and P. Trifonas, editors, *International Handbook of Virtual Learning Environments.* Kluwer Academic Publishers, In press.

[105] Seungyeop Han, Yong yeol Ahn, Sue Moon, and Hawoong Jeong. Collaborative blog spam filtering using adaptive percolation search. In *WWW 2006 Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '06: the 15th international conference on World Wide Web*, 2006.

[106] Erik Hatcher and Otis Gospodnetic. *Lucene in Action (In Action series).* Manning Publications Co., Greenwich, CT, USA, 2004.

[107] Vasileios Hatzivassiloglou and Kathleen R. McKeown. Predicting the semantic orientation of adjectives. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 174–181, Morristown, NJ, USA, 1997. Association for Computational Linguistics.

[108] Monika R. Henzinger, Rajeev Motwani, and Craig Silverstein. Challenges in web search engines. *SIGIR Forum*, 36(2):11–22, 2002.

[109] Susan C. Herring, Inna Kouper, John C. Paolillo, Lois Ann Scheidt, Michael Tyworth, Peter Welsch, Elijah Wright, and Ning Yu. Conversations in the blogosphere: An analysis "from the bottom up". In *HICSS*, 2005.

[110] Susan C. Herring, Inna Kouper, Lois Ann Scheidt, and Elijah L. Wright. Women and children last: The discursive construction of weblogs. In L. Gurak, S. Antonijevic, L. Johnson, C. Ratliff, and J. Reyman, editors, *Into the Blogosphere; Rhetoric, Community and Culture of Weblogs*, 2004.

[111] Susan C. Herring and John C. Paolillo. Gender and genre variation in weblogs. *Journal of Sociolinguistics*, 10(4):460–480, Sept 2006.

[112] Susan C. Herring, Lois Ann Scheidt, Sabrina Bonus, and Elijah Wright. Bridging the gap: A genre analysis of weblogs. In *The 37th Annual Hawaii International Conference on System Sciences (HICSS'04)*, 2004.

[113] Susan C. Herring, Lois Ann Scheidt, Inna Kouper, and Elijah Wright. A longitudinal content analysis of weblogs: 2003-2004. In Mark Tremayne, editor, *Blogging, Citizenship and the Future of Media*. Routledge, 2006.

[114] Francis Heylighen and Jean-Marc Dewaele. Variation in the contextuality of language: An empirical measure. *Context in Context. Special issue Foundations of Science*, 7(3):293–340, 2002.

[115] Djoerd Hiemstra. *Using Language Models for Information Retrieval.* Phd thesis, Enschede, January 2001.

[116] Nobuaki Hiroshima, Setsuo Yamada, Osamu Furuse, and Ryoji Kataoka. Searching for sentences expressing opinions by using declaratively subjective clues. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, pages 39–46, Sydney, Australia, July 2006.

[117] Thomas Hofmann. Latent semantic models for collaborative filtering. *ACM Trans. Inf. Syst.*, 22(1):89–115, 2004.

[118] Vera Hollink, Jaap Kamps, Christof Monz, and Maarten de Rijke. Monolingual document retrieval for European languages. *Information Retrieval*, 7(1):33–52, 2004.

[119] Lars E. Holzman and Willian M. Pottenger. Classification of emotions in internet chat: An application of machine learning using speech phonemes. Technical Report LU-CSE-03-002, Lehigh University, 2003.

[120] John Horrigan. Home broadband adoption 2006. *Pew Internet & American Life Project*, May 2006.

[121] David Huffaker. Gender similarities and differences in online identity and language use among teenage bloggers. Master's thesis, Georgetown University, April 2004.

[122] Matthew Hurst. GIS and the blogosphere. In *WWW 2005 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[123] Matthew Hurst. 24 hours in the blogosphere. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[124] Matthew Hurst. Temporal text mining. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[125] Umbria Inc. Spam in the blogosphere. white paper, 2006.

[126] Intelliseek. Consumer-generated media exceeds traditional advertising for influencing consumer behavior, 2005. Available online at http://www.nielsenbuzzmetrics.com/release.asp?id=141, accessed December 2006.

[127] Kazunari Ishida. Extracting latent weblog communities: A partitioning algorithm for bipartite graphs. In *WWW 2005 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[128] Bernard J. Jansen and Udo Pooch. Web user studies: a review and framework for future work. *Journal of the American Society of Science and Technology*, 52(3):235–246, 2001.

[129] Bernard J. Jansen and Amanda Spink. An analysis of Web searching by European AlltheWeb.com users. *Information Processing & Management (IPM)*, 41(2):361–381, 2005.

[130] Bernard J. Jansen, Amanda Spink, and Tefko Saracevic. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Processing & Management (IPM)*, 36(2):207–227, 2000.

[131] Akshay Java, Pranam Kolari, Tim Finin, and Tim Oates. Modeling the spread of influence on the blogosphere. In *WWW 2006 Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '06: the 15th international conference on World Wide Web*, 2006.

[132] Rong Jin, Luo Si, and Chengxiang Zhai. A study of mixture models for collaborative filtering. *Inf. Retr.*, 9(3):357–382, 2006.

[133] Thorsten Joachims. Text categorization with suport vector machines: Learning with many relevant features. In *ECML '98: Proceedings of the 10th European Conference on Machine Learning*, pages 137–142, London, UK, 1998. Springer-Verlag.

[134] Quentin Jones. Virtual-communities, virtual settlements & cyber-archaeology: A theoretical outline. *Journal of Computer-Mediated Communication*, 3(3), 1997.

[135] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Prentice Hall PTR, Upper Saddle River, NJ, USA, 2000.

[136] Adam Kalai, Stanley F. Chen, Avrim Blum, and Ronald Rosenfeld. On-line algorithms for combining language models. In *Proceedings of the International Conference on Accoustics, Speech, and Signal Processing (ICASSP '99)*, 1999.

[137] Jaap Kamps, Maarten Marx, Rob J. Mokken, and Maarten de Rijke. Using WordNet to measure semantic orientations of adjectives. In *Proceedings LREC 2004*, 2004.

[138] Hiroshi Kanayama and Tetsuya Nasukawa. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 355–363, Sydney, Australia, July 2006. Association for Computational Linguistics.

[139] Jussi Karlgren. *Stylistic Experiments for Information Retrieval.* PhD thesis, Stockholm University, 2000.

[140] KDD 2005 Cup URL: http://kdd05.lac.uic.edu/kddcup.html, accessed January 2006.

[141] Alistair Kennedy and Diana Inkpen. Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2):110–125, May 2006.

[142] Adam Kilgarriff. Comparing corpora. *International Journal of Corpus Linguistics*, 6(1):1–37, 2001.

[143] Soo-Min Kim and Eduard Hovy. Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pages 1367–1373, 2004.

[144] Soo-Min Kim and Eduard Hovy. Automatic detection of opinion bearing words and sentences. In *Second International Joint Conference on Natural Language Processing (IJCNLP-05)*, 2005.

[145] J. Peter Kincaid, Robert P. Fishburn, Richard L. Rogers, and Brad S. Chissom. Derivation of new readability formulas for navy enlisted personnel. Technical Report Research Branch Report 8-75, Millington, Tenn, Naval Air Station, 1975.

[146] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

[147] Jon M. Kleinberg. Bursty and hierarchical structure in streams. *Data Min. Knowl. Discov.*, 7(4):373–397, 2003.

[148] Asako Koike, Yoshiki Niwa, and Toshihisa Takagi. Automatic extraction of gene/protein biological functions from biomedical text. *Bioinformatics*, 21(7):1227–1236, 2005.

[149] Pranam Kolari, Tim Finin, and Anupam Joshi. SVMs for the blogosphere: Blog identification and splog detection. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[150] Pranam Kolari, Akshay Java, and Tim Finin. Characterizing the splogosphere. In *WWW 2006 Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '06: the 15th international conference on World Wide Web*, 2006.

[151] Moshe Koppel, Shlomo Argamon, and Anat R. Shimoni. Automatically categorizing written texts by author gender. *Literary and Linguistic Computing*, 17(4):401–412, 2002.

[152] Moshe Koppel and Jonathan Schler. Authorship verification as a one-class classification problem. In *ICML '04: Proceedings of the twenty-first international conference on Machine learning*, page 62, New York, NY, USA, 2004. ACM Press.

[153] Jason Kottke. Weblogs and power laws, February 2003. http://www.kottke.org/03/02/weblogs-and-power-laws, accessed June 2006.

[154] Sandeep Krishnamurthy. The multidimensionality of blog conversations: The virtual enactment of september 11. In *Internet Research 3.0*, 2002.

[155] Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. Opinion extraction, summarization and tracking in news and blog corpora. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[156] Ravi Kumar, Jasmine Novak, Prabhakar Raghavan, and Andrew Tomkins. On the bursty evolution of blogspace. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, 2003.

[157] Ravi Kumar, Jasmine Novak, Prabhakar Raghavan, and Andrew Tomkins. Structure and evolution of blogspace. *Communications of the ACM*, 47(12):35–39, 2004.

[158] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, D. Sivakumar, Andrew Tompkins, and Eli Upfal. The web as a graph. In *PODS '00: Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 1–10, New York, NY, USA, 2000. ACM Press.

[159] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew Tomkins. Extracting large-scale knowledge bases from the web. In *VLDB '99: Proceedings of the 25th International Conference on Very Large Data Bases*, pages 639–650, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[160] Takeshi Kurashima, Taro Tezuka, and Katsumi Tanaka. Blog map of experiences: Extracting and geographically mapping visitor experiences from urban blogs. In *WISE '05: 6th International Conference on Web Information Systems Engineering*, pages 496–503, 2005.

[161] Anísio Lacerda, Marco Cristo, Marcos André Gonçalves, Weiguo Fan, Nivio Ziviani, and Berthier A. Ribeiro-Neto. Learning to advertise. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 549–556, 2006.

[162] Marc Langheinrich, Atsuyoshi Nakamura, Naoki Abe, Tomonari Kamba, and Yoshiyuki Koseki. Unintrusive customization techniques for web advertising. In *WWW '99: Proceeding of the eighth international conference on World Wide Web*, pages 1259–1272, 1999.

[163] Victor Lavrenko. Optimal mixture models in IR. In *Advances in Information Retrieval: Proceedings 24th European Conference on IR Research (ECIR 2002)*, pages 193–212, 2002.

[164] Joon Ho Lee. Analyses of multiple evidence combination. In *SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 267–276, 1997.

[165] Amanda Lenhart and Susannah Fox. Bloggers: a portrait of the internet's new storytellers. *Pew Internet & American Life Project*, July 2006.

[166] Amanda Lenhart, John Horrigan, and Deborah Fallows. Content creation online. *Pew Internet & American Life Project*, February 2004.

[167] Gilly Leshed and Joseph Kaye. Understanding how bloggers feel: recognizing affect in blog posts. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1019–1024, New York, NY, USA, 2006. ACM Press.

[168] Lun Li, David Alderson, Reiko Tanaka, John C. Doyle, and Walter Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Mathematics*, 2(4):431–523, 2005.

[169] Xiaoyan Li and W. Bruce Croft. Time-based language models. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 469–475, 2003.

[170] David Liben-Nowell, Jasmine Novak, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Geographic routing in social networks. *Proceedings of the National Academy of Sciences*, 102(33):11623–11628, 2005.

[171] Jia Lin and Alexander Halavais. Mapping the blogosphere in America. In *WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '04: the 13th international conference on World Wide Web*, 2004.

[172] Yu-Ru Lin, Wen-Yen Chen, Xiaolin Shi, Richard Sia, Xiaodan Song, Yun Chi, Koji Hino, Hari Sundaram, Jun Tatemura, and Belle Tseng. The splog detection task and a solution based on temporal and link properties. In *TREC*, Gaithersburg, Maryland USA, 2006.

[173] Yu-Ru Lin, Hari Sundaram, Yun Chi, Jun Tatemura, and Belle Tseng. Discovery of blog communities based on mutual awareness. In *WWW 2006 Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '06: the 15th international conference on World Wide Web*, 2006.

[174] Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, 2003.

[175] Bing Liu, Minqing Hu, and Junsheng Cheng. Opinion observer: analyzing and comparing opinions on the web. In *WWW2005: the 14th international conference on World Wide Web*, pages 342–351, New York, NY, USA, 2005. ACM Press.

[176] Hugo Liu, Henry Lieberman, and Ted Selker. A model of textual affect sensing using real-world knowledge. In *IUI '03: Proceedings of the 8th international conference on Intelligent user interfaces*, pages 125–132, New York, NY, USA, 2003. ACM Press.

[177] Yong Liu. Word of mouth for movies: Its dynamics and impact on box office revenue. *Journal of Marketing*, 70(3), 2006.

[178] LiveJournal Statistics. URL: http://www.livejournal.com/stats.bml, 2006. Accessed December 2006.

[179] Levon Lloyd, Prachi Kaulgud, and Steven Skiena. News vs. blogs: Who gets the scoop? In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[180] Melissa Ludtke, editor. *NIEMAN REPORTS: Journalist's Trade - Weblogs and Journalism*, volume 57,3. Bob Giles, 2003.

[181] Thomas R. Lynam, Chris Buckley, Charles L. A. Clarke, and Gordon V. Cormack. A multi-system analysis of document and term selection for blind feedback. In *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 261–269, 2004.

[182] Craig Macdonald and Iadh Ounis. The TREC Blogs06 Collection: Creating and analysing a blog test collection. Technical Report TR-2006-224, Department of Computing Science, University of Glasgow, 2006.

[183] Chris Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999.

[184] Cameron Marlow. Audience, structure and authority in the weblog community. In *The 54th Annual Conference of the International Communication Association*, 2004.

[185] Cameron Marlow. Investment and attention in the weblog community. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[186] Cameron Marlow, Mor Naaman, Danah Boyd, and Marc Davis. HT06, tagging paper, taxonomy, Flickr, academic article, to read. In *HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia*, pages 31–40, New York, NY, USA, 2006. ACM Press.

[187] Mary Hodder. A comparison of how some blog aggregation and RSS search tools work, 2005. http://napsterization.org/stories/archives/000500.html and http://napsterization.org/stories/archives/000502.html, accessed November 2005.

[188] Tony McEnery and Michael Oakes. *Handbook of Natural Language Processing*, chapter Authorship Studies / Textual Statistics. Marcel Dekker, 2000.

[189] G. Harry McLaughlin. SMOG grading: A new readability formula. *Journal of Reading*, 12(8):639–646, 1969.

[190] Merriam-Webster. Merriam-Webster's words of the year, 2004. http://www.m-w.com/info/04words.htm, accessed July 2006.

[191] Donald Metzler and W. Bruce Croft. A Markov random field model for term dependencies. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 472–479, 2005.

[192] Hongcheng Mi and I-Heng Mei. Searching sentiments in blogs. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[193] Rada Mihalcea and Hugo Liu. A corpus-based approach to finding happiness. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[194] Gilad Mishne. Experiments with mood classification in blog posts. In *Style2005 – 1st Workshop on Stylistic Analysis of Text for Information Access, at SIGIR '05: the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005.

[195] Gilad Mishne. AutoTag: A collaborative approach to automated tag assignment for weblog posts. In *WWW '06: Proceedings of the 15th international World Wide Web conference on Alternate track papers & posters*, 2006.

[196] Gilad Mishne. Information access challenges in the blogspace. In *IIIA-2006 - International Workshop on Intelligent Information Access*, 2006.

[197] Gilad Mishne. Multiple ranking strategies for opinion retrieval in blogs. In *TREC*, Gaithersburg, Maryland USA, 2006.

[198] Gilad Mishne. Using blog properties to improve retrieval. In *ICWSM 2007 – International Conference on Weblogs and Social Media*, 2007.

[199] Gilad Mishne, David Carmel, and Ronny Lempel. Blocking blog spam with language model disagreement. In *First International Workshop on Adversarial Information Retrieval on the Web, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[200] Gilad Mishne and Maarten de Rijke. Boosting web retrieval through query operations. In D.E. Losada and J.M. Fernández-Luna, editors, *Advances in Information Retrieval: Proceedings 27th European Conference on IR Research (ECIR 2005)*, pages 502–516, 2005.

[201] Gilad Mishne and Maarten de Rijke. Capturing global mood levels using blog posts. In *AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[202] Gilad Mishne and Maarten de Rijke. Deriving wishlists from blogs: Show us your blog, and we'll tell you what books to buy. In *WWW '06: Proceedings of the 15th international World Wide Web conference on Alternate track papers & posters*, 2006.

[203] Gilad Mishne and Maarten de Rijke. Language model mixtures for contextual ad placement in personal blogs. In *FinTAL - 5th International Conference on Natural Language Processing*, 2006.

[204] Gilad Mishne and Maarten de Rijke. MoodViews: Tools for blog mood analysis. In *AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[205] Gilad Mishne and Maarten de Rijke. A study of blog search. In M. Lalmas, A. MacFarlane, S. Rüger, A. Tombros, T. Tsikrika, and A. Yavlinsky, editors, *Advances in Information Retrieval: Proceedings 28th European Conference on IR Research (ECIR 2006)*, 2006.

[206] Gilad Mishne, Maarten de Rijke, and Valentin Jijkoun. Using a reference corpus as a user model for focused information retrieval. *Journal of Digital Information Management*, 3(1):47–52, 2005.

[207] Gilad Mishne and Natalie Glance. Leave a reply: An analysis of weblog comments. In *WWW 2006 Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '06: the 15th international conference on World Wide Web*, 2006.

[208] Gilad Mishne and Natalie Glance. Predicting movie sales from blogger sentiment. In *AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[209] Michael Mitzenmacher. A brief history of generative models for power law and lognormal distributions. *Internet Mathematics*, 1(2):226–251, 2004.

[210] Maria C. Monard and Gustavo E.A.P.A. Batista. Learning with skewed class distributions. In *Advances in Logic, Artificial Intelligence and Robotics*, pages 173–180, 2002.

[211] Movable Type Blacklist Filter, with content filtering, URL: http://www.jayallen.org/projects/mt-blacklist/, accessed June 2006.

[212] Tony Mullen and Nigel Collier. Sentiment analysis using support vector machines with diverse information sources. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, 2004.

[213] Tony Mullen and Robert Malouf. A preliminary investigation into sentiment analysis of informal political discourse. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[214] Matt Mullenweg. Trackback spam stats, 2006. http://www.sixapart.com/pipermail/trackback-protocol/2006-February/000088.html, accessed February 2007.

[215] David Nadeau, Catherine Sabourin, Joseph De Koninck, Stan Matwin, and Peter D. Turney. Automatic dream sentiment analysis. In *Workshop on Computational Aesthetics at AAAI-06: The Twenty-first National Conference on Artificial Intelligence*, pages 70–73, 2006.

[216] Shinsuke Nakajima, Junichi Tatemura, Yoichiro Hino, Yoshinori Hara, and Katsumi Tanaka. Discovering important bloggers based on a blog thread analysis. In *WWW 2005 Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '05: the 14th international conference on World Wide Web*, 2005.

[217] Tomoyuki Nanno, Toshiaki Fujiki, Yasuhiro Suzuki, and Manabu Okumura. Automatically collecting, monitoring, and mining japanese weblogs. In *WWW '04: Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, 2004.

[218] Bonnie A. Nardi, Diane J. Schiano, and Michelle Gumbrecht. Blogging as social activity, or, would you let 900 million people read your diary? In *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*, 2004.

[219] Kazuyuki Narisawa, Yasuhiro Yamada, Daisuke Ikeda, and Masayuki Takeda. Detecting blog spams using the vocabulary size of all substrings in their copies. In *WWW 2006 Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '06: the 15th international conference on World Wide Web*, 2006.

[220] Tetsuya Nasukawa and Jeonghee Yi. Sentiment analysis: capturing favorability using natural language processing. In *K-CAP '03: Proceedings of the 2nd international conference on Knowledge capture*, pages 70–77, New York, NY, USA, 2003. ACM Press.

[221] Mark E.J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[222] Kamal Nigam and Matthew Hurst. Towards a robust metric of opinion. In *The AAAI Symposium on Exploring Attitude and Affect in Text (AAAI-EAAT)*, 2004.

[223] Kamal Nigam and Matthew Hurst. Measuring aggregate opinion from text with confidence bounds. Intelliseek TR, 2005.

[224] Stephanie Nilsson. The function of language to facilitate and maintain social networks in research weblogs. Umeå Universitet, 2003.

[225] Joint statement from Yahoo, Google, and others regarding the "nofollow" tag, URLs: http://www.google.com/googleblog/2005/01/preventing-comment-spam.html, http://www.ysearchblog.com/archives/000069.html.

[226] No nofollow: fight spam, not blogs, URL: http://www.nonofollow.net, accessed June 2005.

[227] Scott Nowson. *The Language of Weblogs: A study of genre and individual differences.* PhD thesis, University of Edinburgh, 2006.

[228] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. Detecting spam web pages through content analysis. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 83–92, New York, NY, USA, 2006. ACM Press.

[229] Douglas Oard, Tamer Elsayed, Jianqiang Wang, Yejun Wu, Pengyi Zhang, Eileen Abels, Jimmy Lin, and Dagbert Soergel. TREC-2006 at Maryland: Blog, Enterprise, Legal and QA Tracks. In *TREC*, Gaithersburg, Maryland USA, 2006.

[230] Jon Oberlander and Scott Nowson. Whose thumb is it anyway? classifying author personality from weblog text. In *ACL '06: Proceedings of the 44th Annual Meeting on Association for Computational Linguistics, Poster Session*, pages 627–634, 2006.

[231] Paul Ogilvie and James P. Callan. Experiments using the lemur toolkit. In *TREC*, Gaithersburg, Maryland USA, 2001.

[232] Tsutomu Ohkura, Yoji Kiyota, and Hiroshi Nakagawa. Browsing system for weblog articles based on automated folksonomy. In *WWW 2006 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '06: the 15th international conference on World Wide Web*, 2006.

[233] Mizuki Oka, Hirotake Abe, and Kazuhiko Kato. Extracting topics from weblogs through frequency segments. In *WWW 2006 Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '06: the 15th international conference on World Wide Web*, 2006.

[234] Charles E. Osgood, George J. Suci, and Percy Tannenbaum. *The Measurement of Meaning.* University of Illinois Press, 1967.

[235] Iadh Ounis, Maarten de Rijke, Craig Macdonald, Gilad Mishne, and Ian Soboroff. Overview of the TREC-2006 Blog Track. In *TREC*, Gaithersburg, Maryland USA, 2006.

[236] Sara Owsley, Sanjay Sood, and Kristian J. Hammond. Domain specific affective classification of documents. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[237] Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 271–278, Barcelona, Spain, July 2004.

[238] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86, 2002.

[239] Perseus Development Corporation. The blogging iceberg – a blog survey, 2003. http://www.perseus.com/blogsurvey/iceberg.html, accessed May 2005.

[240] Perseus Development Corporation. The blogging geyser – a blog survey, 2005. http://www.perseus.com/blogsurvey/geyser.html, accessed May 2005.

[241] John M. Pierre. Mining knowledge from text collections using automatically generated metadata. In *PAKM '02: Proceedings of the 4th International Conference on Practical Aspects of Knowledge Management*, pages 537–548, London, UK, 2002. Springer-Verlag.

[242] Jay M. Ponte. Language models for relevance feedback. In *Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval*, 2000.

[243] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281. ACM Press, 1998.

[244] Hsiao Tieh Pu and Shui Lung Chuang. Auto-categorization of search terms toward understanding web users' information needs. In *ICADL 2000: Intern. Conference on Asian Digital Libraries*, 2000.

[245] Yan Qi and K. Selçuk Candan. CUTS: CUrvature-based development pattern analysis and segmentation for blogs and other Text Streams. In *HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia*, pages 1–10, New York, NY, USA, 2006. ACM Press.

[246] Feng Qiu, Zhenyu Liu, and Junghoo Cho. Analysis of user web traffic with a focus on search activities. In *WebDB*, pages 103–108, 2005.

[247] Hong Qu, Andrea La Pietra, and Sarah Poon. Blog classification using NLP: Challenges and pitfalls. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[248] Lee Rainie. The state of blogging. *Pew Internet & American Life Project*, January 2005.

[249] Jonathon Read. Recognising affect in text using pointwise-mutual information. Master's thesis, University of Sussex, 2004.

[250] Gartner Research. Gartner's top predictions for IT organizations and users, 2007 and beyond, December 2006.

[251] Paul Resnick and Hal R. Varian. Recommender systems. *Commun. ACM*, 40(3):56–58, 1997.

[252] Berthier Ribeiro-Neto, Marco Cristo, Paulo B. Golgher, and Edleno Silva de Moura. Impedance coupling in content-targeted advertising. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 496–503, 2005.

[253] Ellen Riloff, Janyce Wiebe, and Theresa Wilson. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, pages 25–32, 2003.

[254] Stephen E. Robertson, Steve Walker, Micheline Hancock-Beaulieu, Mike Gatford, and A. Payne. Okapi at TREC-4. In *TREC 4*, Gaithersburg, Maryland USA, 1995.

[255] Daniel E. Rose and Danny Levinson. Understanding user goals in web search. In *WWW '04: Proceedings of the 13th intern. conf. on World Wide Web*. ACM Press, 2004.

[256] Ronald Rosenfeld. Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE*, 88(8), 2000.

[257] Victoria L. Rubin, Jeffrey M. Stanton, and Elizabeth D. Liddy. Discerning emotions in texts. In *The AAAI Symposium on Exploring Attitude and Affect in Text (AAAI-EAAT)*, 2004.

[258] Lydia Saad. Blog readership bogged down, February 2006. Gallup Poll News Service.

[259] Marta Sabou, Chris Wroe, Carole Goble, and Gilad Mishne. Learning domain ontologies for web service descriptions: an experiment in bioinformatics. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 190–198, 2005.

[260] Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A Bayesian approach to filtering junk E-mail. In *Learning for Text Categorization: Papers from the 1998 Workshop*, Madison, Wisconsin, 1998. AAAI Technical Report WS-98-05.

[261] Franco Salvetti and Nicolas Nicolov. Weblog classification for fast splog filtering: A URL language model segmentation approach. In *HLT-NAACL 2006: Human Language Technology Conference /North American chapter of the Association for Computational Linguistics Annual Meeting*, 2006.

[262] Mark Sanderson and Justin Zobel. Information retrieval system evaluation: effort, sensitivity, and reliability. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 162–169, 2005.

[263] John W. Santrock. *Psychology*. McGraw-Hill, 2000.

[264] Bilge Say and Varon Akman. Current approaches to punctuation in computational linguistics. *Computers and the Humanities*, 30(6):457–469, 1996.

[265] Diane J. Schiano, Bonnie A. Nardi, Michelle Gumbrecht, and Luke Swartz. Blogging by the rest of us. In *CHI '04*, pages 1143–1146, 2004.

[266] Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James Pennebaker. Effects of age and gender on blogging. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[267] H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, Manchester, UK, 1994.

[268] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[269] Yuichiro Sekiguchi, Harumi Kawashima, Hidenori Okuda, and Masahiro Oku. Topic detection from blog documents using users' interests. In *MDM '06: Proceedings of the 7th International Conference on Mobile Data Management (MDM'06)*, page 108, Washington, DC, USA, 2006. IEEE Computer Society.

[270] James G. Shanahan, Yan Qu, and Janyce Wiebe, editors. *Computing Attitude and Affect in Text: Theory and Applications*, volume 20 of *Information Retrieval Series*. Springer-Verlag New York, Inc., 2005.

[271] Claude E. Shannon. Prediction and entropy of printed English. *Bell Sys. Tech. J.*, 30:50–64, 1951.

[272] Cyrus Shaoul and Chris Westbury. Usenet orthographic frequencies for 111,627 English words, 2006.

[273] Joseph A. Shaw and Edward A. Fox. Combination of multiple searches. In *TREC*, Gaithersburg, Maryland USA, 1994.

[274] Dou Shen, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Latent friend mining from blog data. In *ICDM '06: Proceedings of the Sixth IEEE International Conference on Data Mining*. IEEE Computer Society, 2006.

[275] Xuehua Shen, Susan Dumais, and Eric Horvitz. Analysis of topic dynamics in web search. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 1102–1103, 2005.

[276] Clay Shirky. Power laws, weblogs, and inequality, 2003. http://www.shirky.com/writings/powerlaw_weblog.html, accessed June 2006.

[277] Matthew A. Siegler. Private communications, October 2005.

[278] Craig Silverstein, Hannes Marais, Monika Henzinger, and Michael Moricz. Analysis of a very large web search engine query log. *SIGIR Forum*, 33(1):6–12, 1999.

[279] Ian Soboroff. Overview of the TREC 2004 Novelty Track. In *TREC*, Gaithersburg, Maryland USA, 2004.

[280] Ian Soboroff and Donna Harman. Overview of the TREC 2003 Novelty Track. In *TREC*, Gaithersburg, Maryland USA, 2003.

[281] Socialtext. European blogosphere summary, 2005. http://www.eu.socialtext.net/, accessed July 2006.

[282] Fei Song and W. Bruce Croft. A general language model for information retrieval. In *CIKM '99: Proceedings of the eighth international conference on Information and knowledge management*, pages 316–321, New York, NY, USA, 1999. ACM Press.

[283] Jeffrey S. Sparrow and Ilana R. Sparrow. Predicting movie grosses: Winners and losers, blockbusters and sleepers. *Chance*, 13:15–24, 2000.

[284] Amanda Spink and Bernard J. Jansen. *Web Search: Public Searching of the Web*, volume 6. Kluwer Academic Publishers, 2004.

[285] Amanda Spink, Bernard J. Jansen, Dietmar Wolfram, and Tefko Saracevic. From E-Sex to E-Commerce: Web search changes. *IEEE Computer*, 35(3):107–111, 2002.

[286] Nicola Stokes. *Applications of Lexical Cohesion in the Topic Detection and Tracking Domain*. PhD thesis, University College Dublin, 2004.

[287] Philip J. Stone, Dexter C. Dunphy, Marshall S. Smith, and Daniel M. Ogilvie. *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press, Cambridge, 1966.

[288] Wen Tau-Wih, Joshua Goodman, and Vitor R. Carvalho. Finding advertising keywords on web pages. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, 2006.

[289] Loren Terveen, Will Hill, Brian Amento, David McDonald, and Josh Creter. PHOAKS: a system for sharing recommendations. *Commun. ACM*, 40(3):59–62, 1997.

[290] Clive Thompson. The long tail theory: Why B-list blogs can make it, too. *New York Magazine*, February 2006.

[291] Tapanee Tirapat, Cleo Espiritu, and Eleni Stroulia. Taking the community's pulse: one blog at a time. In *ICWE '06: Proceedings of the 6th international conference on Web engineering*, pages 169–176, New York, NY, USA, 2006. ACM Press.

[292] Richard M. Tong. An operational system for detecting and tracking opinions in on-line discussions. In *Working Notes of the ACM SIGIR 2001 Workshop on Operational Text Classification*, 2001.

[293] Richard M. Tong and Mark Snuffin. Weblogs as market indicators: Tracking reactions to issues and events. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[294] Ericka Menchen Trevino. Blogger motivations: Power, pull, and positive feedback. In *Internet Research 6.0*, 2005.

[295] Belle L. Tseng, Junichi Tatemura, and Yi Wu. Tomographic clustering to visualize blog communities as mountain views. In *WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, at WWW '04: the 13th international conference on World Wide Web*, 2004.

[296] Peter .D. Turney. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 491–502, London, UK, 2001. Springer-Verlag.

[297] Peter D. Turney. Coherent keyphrase extraction via web mining. In *Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*, pages 434–442, 2003.

[298] Peter D. Turney and Michael L. Littman. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, 21(4):315–346, 2003.

[299] Fiona J. Tweedie and R. Harald Baayen. How variable may a constant be? measures of lexical richness in perspective. *Computers and the Humanities*, 32(5):323–352, 1998.

[300] Trystan Upstill, Nick Craswell, and David Hawking. Predicting fame and fortune: PageRank or indegree? In *Proceedings of the Australasian Document Computing Symposium, ADCS2003*, pages 31–40, Canberra, Australia, December 2003.

[301] Fernanda B. Viégas. Bloggers' expectations of privacy and accountability: An initial survey. *Journal of Computer-Mediated Communication*, 10:3, 2005.

[302] Luis von Ahn, Manuel Blum, and John Langford. Telling humans and computers apart automatically. *Communications of the ACM*, 47(2):56–60, 2004.

[303] Ellen M. Voorhees and Donna K. Harman, editors. *TREC: Experiment and Evaluation in Information Retrieval.* The MIT Press, 2005.

[304] Jill Walker. Blog (definition). In Marie-Laure Ryan David Herman, Manfred Jahn, editor, *Routledge Encyclopedia of Narrative Theory.* Routledge, 2005.

[305] Chingning Wang, Ping Zhang, Risook Choi, and Michael D'Eredita. Understanding consumers attitude toward advertising. In *Eighth Americas Conference on Information Systems*, pages 1143–1148, 2002.

[306] Yong Wang and Ian H. Witten. Pace regression. Technical Report 99/12, Department of Computer Science, University of Waikato, September 1999.

[307] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, June 1998.

[308] Carolyn Wei. Formation of norms in a blog community. In L. Gurak, S. Antonijevic, L. Johnson, C. Ratliff, and J. Reyman, editors, *Into the Blogosphere; Rhetoric, Community and Culture of Weblogs*, 2004.

[309] Nancy White. Blogs and community, 2006. http://www.fullcirc.com/weblog/2006/07/blogs-and-community-part-4.htm, accessed July 2006.

[310] Janyce Wiebe. Learning subjective adjectives from corpora. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 735–740. AAAI Press / The MIT Press, 2000.

[311] Janyce Wiebe, Eric Breck, Chris Buckley, Claire Cardie, Paul Davis, Bruce Fraser, Diane J. Litman, David R. Pierce, Ellen Riloff, Theresa Wilson, David Day, and Mark T. Maybury. Recognizing and organizing opinions expressed in the world press. In *AAAI Spring Symposium on New Directions in Question Answering*, pages 12–19, 2003.

[312] Janyce Wiebe, Theresa Wilson, Rebecca Bruce, Matthew Bell, and Melanie Martin. Learning subjective language. *Comput. Linguist.*, 30(3):277–308, 2004.

[313] Dave Winer. What makes a weblog a weblog? In *Weblogs at Harvard Law, Berkman Center for Internet and Society*, 2003. http://blogs.law.harvard.edu/whatMakesAWeblogAWeblog, accessed May 2006.

[314] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2nd edition, 2005.

[315] Yejun Wu, Douglas Oard, and Ian Soboroff. An exploratory study of the W3C mailing list test collection for retrieval of emails with pro/con arguments. In *Third Conference on Email and Anti-Spam (CEAS 2006)*, 2006.

[316] Yi Wu and Belle L. Tseng. Important weblog identification and hot story summarization. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[317] *Annual WWW Workshops on the Weblogging Ecosystem*, 2004, 2005, 2006.

[318] *Collaborative Web Tagging Workshop at WWW*, 2006.

[319] Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11, 1996.

[320] Yahoo Development Network, URL: http://developer.yahoo.net.

[321] Yiming Yang and Xin Liu. A re-examination of text categorization methods. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 42–49, New York, NY, USA, 1999. ACM Press.

[322] Norihito Yasuda, Tsutomu Hirao, Jun Suzuki, and Hideki Isozaki. Identifying bloggers' residential area. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, 2006.

[323] Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, and Wayne Niblack. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *ICDM '03: Proceedings of the Third IEEE International Conference on Data Mining*, page 427, Washington, DC, USA, 2003. IEEE Computer Society.

[324] Elad Yom-Tov, Shai Fine, David Carmel, and Adam Darlow. Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 512–519, 2005.

[325] Ding Zhou, Xiang Ji, Hongyuan Zha, and C. Lee Giles. Topic evolution and social interactions: How authors effect research. In *CIKM '06: Proceedings of the fifteenth international conference on Information and knowledge management*, New York, NY, USA, 2006. ACM Press.

[326] Weizhong Zhu, Min Song, and Robert B. Allen. TREC 2005 enterprise track results from Drexel. In *TREC*, Gaithersburg, Maryland USA, 2005.

[327] Xiaojin Zhu and Roni Rosenfeld. Improving trigram language modeling with the world wide web. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'01)*, pages 533–536, 2001.