How to Overcome Tiredness

Estimating Topic-Mood Associations

Krisztian Balog   Maarten de Rijke

ISLA, University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam
The Netherlands
kbalog,mdr@science.uva.nl

Abstract

We address the task of associating moods with a given topic, using a large set of mood-annotated blog posts. We argue that a simple frequency-based baseline does not suffice as it fails to capture topic-dependence. Instead, we propose three models based on language modeling techniques to accomplish the topic-mood association task. Based on anecdotal evidence and other considerations, including complexity and efficiency, we identify a clearly preferred model.

1. Introduction

The potential of blogs to serve as a source of information about people's responses to current events or products and services has been recognized by many; see, e.g., [1, 2]. Blogs are an obvious target for sentiment analysis, opinion mining, and, more generally, for methods analyzing non-objective aspects of online content. Some blogging platforms, including LiveJournal, allow bloggers to tag their post with their *mood* at the time of writing; users can either select a mood from a predefined list of 132 common moods such as "shocked" or "thankful", or enter free-text. A large percentage of Live-Journal bloggers use the mood tagging feature, which results in a stream of many thousands of mood-tagged blog posts per day.

MoodViews [7] is a set of tools for tracking and analyzing the stream of mood-tagged blog posts made available by Live-Journal. The MoodViews tools available at present offer different views on this stream, ranging from tracking the mood levels (the aggregate across all postings of the various moods), predicting them, and explaining sudden swings in mood levels. New MoodViews tools that are currently under development are focused on exploring the relationship between mood levels and the content of the mood-tagged blog posts. Given a topic, *Moodspotter* is a tool that returns the moods associated with the topic. There is an obvious baseline approach to implementing this functionality: given a topic t, simply retrieve all mood-tagged posts that talk about t, count, say on an hourly or daily basis, the frequencies of each of the mood tags, and return the most frequent one(s). In Figure 1 we show two example topics for November 2006: *shopping* and *thanksgiving*. The height of the bars reflect the number of blog posts relevant to the topic, while the color of the bars denote the most dominant mood for each day according to

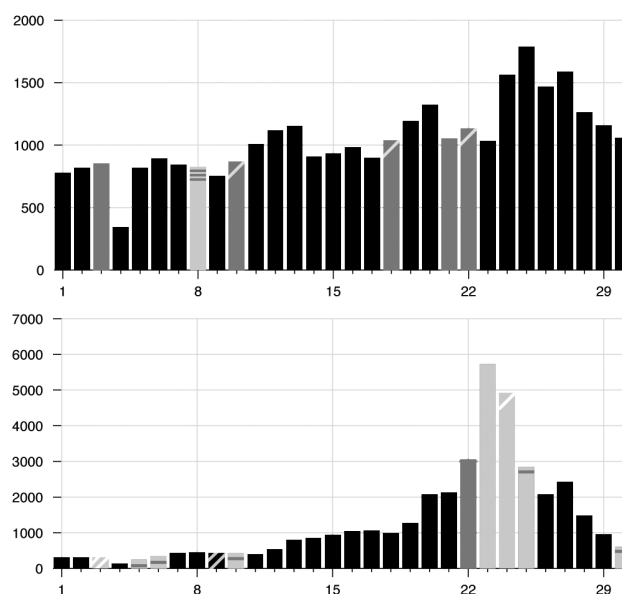the frequency-based baseline just mentioned (see Section 3.1 for a mood-color map, relating moods and colors).

Fig. 1: *Two example topics for November 2006. (Top): shopping. (Bottom): thanksgiving. The height of the bars reflect the number of blog posts relevant to the topic, while the color of the bars denote the most dominant mood for each day.*

The problem with this frequency-based approach is that given a topic, it picks the most frequent mood, which is not necessarily the most closely associated mood. When nothing "unusual" happens—such as e.g., Thanksgiving on November 23—, the baseline takes the most frequent mood to be the most dominant one, irrespective of the topic: *tired*. When looking for the mood that is most closely associated to a topic, this result is not necessarily the mood that is the most appropriate one. Our aim with this paper is to investigate how to overcome this problem of *tiredness*, i.e., how to select the most closely associated mood for a topic, instead of the most dominant one. To this end, we propose and compare three (non-baseline) topic-mood association models.

Evaluation of the proposed solutions is highly non-trivial: there is no "ground truth" for associations between topics and moods, and we do not have the resources to set up a large scale user study. Instead, we use the following dimensions to favor one model over another one: (1) anecodotal

evidence; (2) complexity of the proposed methods; (3) implementation effort involved; (4) pragmatic reasons (uses existing results/resources, less IO/network, allows for better caching, etc.); (5) incremental nature of the method (so that we can display intermediate results); (6) extendibility.

The rest of the paper is organized as follows. In Section 2 we describe our models for estimating topic-mood associations. Then, in Section 3 we compare our methods and report on our findings. We conclude in Section 4.

2. From Topics to Moods

We formalize the problem of identifying moods associated with a given topic, as follows: *what is the probability of a mood m being associated with the query topic q?* That is, we determine $p(m|q)$, and rank moods m according to this probability. The top k moods are deemed to be most probably associated ones for the given topic. Now, instead of computing this probability directly, we apply Bayes' Theorem, and obtain

$$p(m|q) = \frac{p(q|m)p(m)}{p(q)}, \qquad (1)$$

where $p(m)$ is the probability of a mood and $p(q)$ is the probability of a topic. Since $p(q)$ is a constant, it can be ignored for the purpose of ranking. Thus we have:

$$p(m|q) \propto p(q|m)p(m). \qquad (2)$$

The task, then, is to estimate $p(q|m)$—the probability of a topic q given a mood m. We consider three models, based on language modeling techniques [3, 6]. The language modeling setting allows us to use blog posts to build associations between topics and moods in a principled manner.

2.1 Model 1: Mood model

Our first model for estimating the probability $p(q|m)$ builds on well-known intuitions from standard language modeling techniques applied to document retrieval. A mood m is represented by a multinomial probability distribution over the vocabulary of terms (i.e., $p(t|m)$). Since $p(t|m)$ may contain zero probabilities, due to data sparseness, it is standard to employ smoothing. Therefore, we infer a mood model θ_m for each mood m, such that the probability of a term given the mood model is $p(t|\theta_m)$. We can then estimate the probability of a query topic being generated by the mood model. The query likelihood is obtained by taking the product across all the terms in the query, such that

$$p(q|\theta_m) = \prod_{t \in q} p(t|\theta_m). \qquad (3)$$

To obtain an estimate of $p(t|\theta_m)$, we first construct an empirical model $p(t|m)$, and then smooth this estimate with the background collection probabilities:

$$p(t|\theta_m) = (1-\lambda) \sum_{d \in D_m} p(t|d) + \lambda p(t),$$

where D_m is the set of blog posts labeled with the mood m, $p(t|d)$ is the maximum-likelihood estimate of the term, and $p(t)$ is the background model. If we put together our choices so far, the mood model becomes:

$$p(q|\theta_m) = \prod_{t \in q} \left\{ (1-\lambda) \sum_d p(t|d) + \lambda p(t) \right\}. \qquad (4)$$

In words, Model 1 amasses all the term information from all the blog posts labeled with the mood and uses this to represent that mood. This model is used to predict how likely this mood would produce a query q.

2.2 Model 2: Post model

In our second model we look at the blog posts that best describe the query topic, and then look at moods that are most strongly associated with these posts. The topic and the mood are considered to be conditionally independent, and we use blog posts as a bridge to resolve their connection:

$$p(q|m) = \sum_{d \in D_m} p(q|\theta_d), \qquad (5)$$

where D_m is the set of blog posts labeled with the mood m. To obtain the probability of a query, given the blog post (i.e., $p(q|\theta_d)$) we use a standard language modeling approach:

$$p(q|\theta_d) = \prod_{t \in q} p(t|\theta_d) \qquad (6)$$

$$p(t|\theta_d) = (1-\lambda)p(t|d) + \lambda p(t), \qquad (7)$$

where $p(t|d)$ is the maximum-likelihood estimate of the term and $p(t)$ is the background model. Putting everything together, we obtain

$$p(q|m) = \sum_{d \in D_m} \left\{ \prod_{t \in q} \left((1-\lambda)p(t|d) + \lambda p(t) \right) \right\}. \qquad (8)$$

Under this model, we can think of the process of finding the associated moods as follows. Given a collection of blog posts ranked according to the topic, we examine each post and if relevant, we then examine the mood label of this post.

2.3 Model 3: Topic model

Instead of attempting to model the query generation process via mood or blog post models, here we build a topic model to represent the query. Given a collection of posts and a query topic q, we assume that there exists an unknown topic model θ_k that assigns probabilities $p(t|\theta_k)$ to the term occurrences in the topic posts. Both the query and the posts are sampled from θ_k (as opposed to the previous approaches, where a query is assumed to be sampled from a specific post or mood model). The main task is to estimate $p(t|\theta_k)$, the probability of a term given the topic model. Lavrenko and Croft [4] suggest a reasonable way of obtaining such an approximation:

$$p(t|\theta_k) \approx p(t|q) = \frac{p(t, q_1, \ldots, q_m)}{p(q_1, \ldots, q_m)} \qquad (9)$$

$$= \frac{p(t, q_1, \ldots, q_m)}{\sum_{t'} p(t', q_1, \ldots, q_m)}. \qquad (10)$$

In order to estimate the joint probability $p(t, q_1, \ldots, q_m)$, we follow [4] and assume t and $q_1, \ldots, q_m$ are mutually independent, once we pick a source distribution from the set of underlying source distributions U. If we assume U to be the set of blog posts, we get:

$$p(t, q_1, \ldots, q_m) = \sum_{d \in U} p(d) \left\{ p(t|\theta_d) \prod_{i=1}^m p(q_i|\theta_d) \right\}. \qquad (11)$$

Here, $p(d)$ denotes some prior distribution over the set U, which is now taken to be uniform; $p(t|\theta_d)$ specifies the probability of observing t if we pick a random term from blog post d. We compute $p(t|\theta_d)$ using Eq. 7.

In order to rank moods according to the topic model defined, we use the KL-divergence to measure the difference between the mood models and the topic model. Moods with smaller divergence from the topic model are considered to be more likely to be associated with that topic.

$$KL(\theta_k||\theta_m) = \sum_t p(t|\theta_k) \log \frac{p(t|\theta_k)}{p(t|\theta_m)}. \qquad (12)$$

The mood model θ_m is defined in Eq. 4. By using the KL-divergence, instead of the probability of a mood given the topic model $p(ca|\theta_k)$, we avoid normalization problems.

2.4 Mood prior

We use the prior $p(m)$, introduced in Eq. 2, to correct for highly frequent moods. This is expressed as

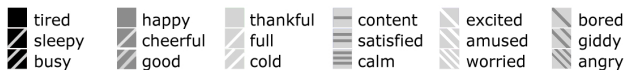$$p(m) = 1 - \frac{n(m)}{\sum_{m'} n(m')}, \qquad (13)$$

where $n(m)$ is the number of posts labeled with mood m.

3. Comparing the Three Models

Now that we have introduced three models for capturing the association between topics and moods, we compare them along the dimensions put forward in the introduction. Most of this section is devoted to a small number of case studies.

3.1 Case studies

Our data set consists of a collection of blog posts from Live-Journal.com, annotated with moods. We present the following set up. Users are provided with an interface where they can choose a topic and select a period of one month. In response, the system returns a histogram with the most strongly associated mood per day, as well as a list of the top three moods per day. For visualization purposes, we use the following mood-color map:

■ tired	happy	thankful	content	excited	bored
sleepy	cheerful	full	satisfied	amused	giddy
busy	good	cold	calm	worried	angry

Below, we consider two types of examples: with a significant event, and without a significant event.[1]

Shopping. We start we an example of a topic/period combination for which no significant event appears to have taken place: *shopping* in November 2006. Shopping is an activity which has been shown to be a reason for happiness [5], therefore, we expect that "positive" moods, such as *happy* or *cheerful*, are associated with it. Using the term *shopping* as a topic, Model 1 returns "random" moods, a different one for each day. Model 2 returns the result we expect, *happy* and *cheerful* are dominating, however *tired* is still present. Model 3 returns *tired* in the first place, while the 2nd and 3rd ranked mood is always *content*, *happy*, or *cheerful*. Figure 2 shows the associated moods returned by Model 2 and 3.

iPod. This is another topic without a significant event, where it is extremely hard to phrase any expectations. The baseline and Model 3 return *tired* for almost each day. In case of Model 1 and Model 2 we witness a wide range of moods returned. The average number of blog post relevant to the topic *iPod* in our collection was around 250 per day on average—with 132 moods in total, this leaves very sparse data.

Thanksgiving. See Table 1. Here we expect no particular dominant mood in the run-up to Thanksgiving, perhaps some anticipation of the significant event, and around Thanksgiving day itself, we expect increased levels of thankfulness and

---

[1] A significant event is when something unusual is happening, i.e., there is a significant growth in the number of relevant blog posts.
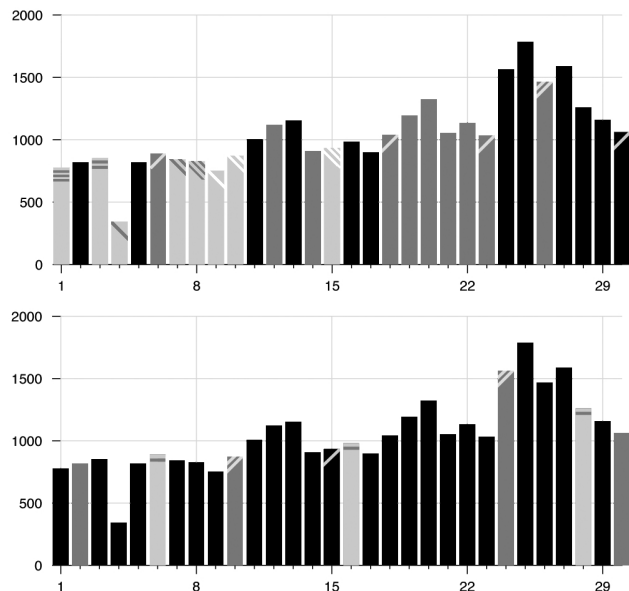

Fig. 2: *Moods associated with the topic* shopping. *(Top): Model 2. (Bottom): Model 3*

enjoyment (and similar positive moods). All display this type of behavior.

Steve Irwin. This example involves another significant event. On September 4, 2006 Australian conservationist and television personality Steve Irwin ("The Crocodile Hunter") was killed in a freak accident. Here we would expect to see mostly cheerful moods leading up to September 4, with negative moods for the days following Irwin's death (i.e., *sad*, *shocked*, *crushed*, etc.)—this is indeed what we observe for Model 2 and Model 3, while Model 1 produces fairly random results and *tired*ness rears its head according to Model 3 in the days prior to September 4; see Table 2.

3.2 Upshot

Let us step back and take stock. We saw two types of phenomena. If there is no significant event for a given topic/period combination (as with the iPod and shopping examples), then Model 1 returns "random" (infrequent) moods, a different one for each day. Model 2 favors frequent moods, but the results for our examples are closer to the expectation we described, while Model 3 returns the most frequent moods (mainly *tired*). The reason for the "failure" of Model 1 and 2 is the lack of data: usually very few (<10) posts are labeled with the same mood—this is where the "randomness" comes from. When Model 3 fails this is because the distribution of the topic is very similar to that of dominant moods.

In contrast, if there is a significant event for a topic/period combination (e.g., Thanksgiving, Steve Irwin), all models return reasonable results. Looking at the top 3 returned moods, we find that there is a clear order of models based on the ability of capturing the most closely associated mood, and this ranking is: Baseline < Model 1 < Model 2 < Model 3. We believe that Model 3 performs best, since the topic is represented here the most accurately, in the form of a probability distribution over terms.

The message from our anedoctal assessment, then, is this.

date	#rel	Baseline	Model 1	Model 2	Model 3
2006-11-19	1261	tired, happy, excited	thankful, intimidated, moody	happy, amused, excited	happy, content, tired
2006-11-20	2074	tired, content, cheerful	intimidated, thankful, rejuvenated	tired, cheerful, calm	tired, cheerful, content
2006-11-21	2124	tired, excited, happy	thankful, pissed, intimidated	excited, cheerful, cold	content, tired, cheerful
2006-11-22	3046	happy, tired, cheerful	thankful, grateful, rushed	happy, cheerful, chipper	content, cheerful, thankful
2006-11-23	5727	thankful, happy, hungry	jealous, thankful, grateful	thankful, hungry, happy	thankful, content, happy
2006-11-24	4918	full, thankful, content	pissed, full, thankful	full, thankful, happy	full, thankful, content
2006-11-25	2844	content, tired, happy	thankful, recumbent, full	content, happy, tired	thankful, content, happy
2006-11-26	2069	tired, content, happy	intimidated, thankful, irritated	tired, content, happy	content, tired, calm

Table 1: *Topic: thanksgiving*

date	#rel	Baseline	Model 1	Model 2	Model 3
2006-09-02	201	tired, bored, chipper	giggly, pleased, pensive	giggly, amused, pensive	tired, amused, calm
2006-09-03	227	tired, cheerful, awake	sympathetic, drunk, nauseated	happy, cheerful, awake	tired, happy, content
2006-09-04	4943	sad, shocked, crushed	shocked, sympathetic, sad	sad, shocked, crushed	sad, shocked, crushed
2006-09-05	2025	sad, crushed, tired	sympathetic, shocked, sad	sad, crushed, depressed	sad, crushed, shocked
2006-09-06	698	sad, tired, contemplative	thankful, sad, sympathetic	sad, tired, disappointed	sad, thankful, contemplative
2006-09-07	542	sad, tired, calm	sympathetic, enraged, morose	sad, blah, devious	sad, contemplative, tired
2006-09-08	501	sad, contemplative, tired	numb, shocked, sad	sad, blank, depressed	sad, numb, contemplative
2006-09-09	360	tired, sad, calm	enthralled, cynical, sad	sad, depressed, sleepy	sad, enthralled, contemplative
2006-09-10	362	tired, bored, happy	impressed, silly, lethargic	bored, ecstatic, sad	sad, contemplative, tired

Table 2: *Topic: Steve Irwin*

When no significant event happens, associating moods with a topic is a hard task. When there is a significant event, we are able to capture the moods that are most closely associated with the topic.

3.3 Other considerations

Here we discuss the other criteria that are to be taken into account when deciding to favor one model over another. We assume that a document-term index, containing blog posts, is maintained and available, moreover, for each blog post the mood label and the date/time of the post can be accessed. To make things more tangible, let us consider a day as our time unit. To be able to produce one of the plots shown in this paper, $p(q|m)$ needs to be calculated for all mood and day pairs, that means $132 \times 31 = 4092$ times.

Model 1 calculates a probability distribution for each mood, which is extremely costly to do "on the fly." If these mood models are built and kept regularly updated in the background, retrieving the results for a given topic can be done efficiently.

Model 3 is closely related to Model 1, since it calculates the divergence between the mood and the topic models. Still, the topic model can not be pre-calculated, but can only be built after the query is issued by the user. Moreover, comparing two probability distributions (the step that has to be repeated 4092 times) is very expensive. Both Model 1 and 3 are able to show intermediate results.

Model 2 does not require a separate index. An effective implementation retrieves all the posts relevant to the topic, iterates over the result set only once, and aggregates the scores for the corresponding mood and day pairs. The intuition behind Model 2—summing up the relevance of the posts that are labeled with the given mood—does not necessarily require language models. If the task is only the ranking of moods any other retrieval engine/formula which produces relevance scores can be used. This makes this approach very attractive, since it can be easily and effectively implemented on top of an existing blog search engine. It is able to show intermediate results, if the date and mood fields are also indexed.

Taking all of the above features into consideration, Model 2 emerges as the best option.

4. Conclusions

We described three methods for capturing the association between topics and moods in mood-tagged blog posts. Based on anecdotal evidence concerning effectiveness and on other considerations concerning efficiency and flexibility, we identified a clear preferred model: Model 2, which captures topic-mood associations by first ranking posts according to the topic, and then examining the mood-tags associated with them.

As to future work, we are working on optimizations of our implementation of Model 2 (so as to make it more useable in an online setting). We are also examining the addition of time and/or sequential aspects to Model 2 where dominant moods on a given day may depend on moods in previous days.

Acknowledgments

5. References

[1] N. S. Glance, M. Hurst, and T. Tomokiyo. BlogPulse: Automated Trend Discovery for weblogs. In *WWW '04 Workshop on the Weblogging Ecosystem*, 2004.

[2] D. Gruhl, R. Guha, R. Kumar, J. Novak, and A. Tomkins. The predictive power of online chatter. In *Proceedings KDD '05*, pages 78–87, New York, NY, USA, 2005.

[3] D. Hiemstra. *Using Language Models for Information Retrieval*. PhD thesis, University of Twente, 2001.

[4] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings SIGIR'01*, pages 120–127, New York, NY, USA, 2001.

[5] R. Mihalcea and H. Liu. A corpus-based approach to finding happiness. In *AAAI Spring Symposium on Computational Approaches to Weblogs*, 2006.

[6] D. Miller, T. Leek, and R. Schwartz. A hidden Markov model information retrieval system. In *Proceedings SIGIR'99*, pages 214–221, 1999.

[7] Moodviews. Tools for blog mood analysis, 2006. URL: `http://www.moodviews.com/`.