Information Processing and Management 000 (2016) 1–21

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

Formal language models for finding groups of experts☆

Shangsong Liang[1,*], Maarten de Rijke

Informatics Institute, University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands

ARTICLE INFO

Article history:
Received 24 December 2014
Revised 10 October 2015
Accepted 25 November 2015
Available online xxx

Keywords:
Group finding
Entity retrieval
Enterprise search

ABSTRACT

The task of finding groups or teams has recently received increased attention, as a natural and challenging extension of search tasks aimed at retrieving individual entities. We introduce a new group finding task: given a query topic, we try to find knowledgeable groups that have expertise on that topic. We present five general strategies for this group finding task, given a heterogenous document repository. The models are formalized using generative language models. Two of the models aggregate expertise scores of the experts in the same group for the task, one locates documents associated with experts in the group and then determines how closely the documents are associated with the topic, whilst the remaining two models directly estimate the degree to which a group is a knowledgeable group for a given topic. For evaluation purposes we construct a test collection based on the TREC 2005 and 2006 Enterprise collections, and define three types of ground truth for our task. Experimental results show that our five knowledgeable group finding models achieve high absolute scores. We also find significant differences between different ways of estimating the association between a topic and a group.

1. Introduction

A major challenge within any organization is managing the expertise of formal or informal groups of people within the organization such that groups with expertise in a particular area can be identified (Balog, Azzopardi, & de Rijke, 2006; Juang, Huang, & Huang, 2013). Rather than finding knowledgeable individuals, sometimes locating a *group* with appropriate skills and knowledge in an organization is of great importance to the success of a project being undertaken (Lappas, Liu, & Terzi, 2009; Li, Shan, & Lin, 2013; Neshati, Beigy, & Hiemstra, 2014). For instance, an engineering organization may want to find a group of scientists who have expertise for dealing with technical problems when constructing a long high-speed railway without having to trawl through descriptions of individuals or groups (if there are any). A group of doctors in a hospital may have to be found immediately so as to perform an operation for a seriously-ill patient. Identifying the right groups of experts with specific knowledge for a task at hand may reduce costs and save the lives of people.

Finding a group or a team that harbors expertise is different from first finding an expert and then sorting out to which team the expert belongs. Conceptually, the difference is that finding a group mainly focuses on how to collect evidence so

☆ This paper is a substantially revised and extended version of Liang and de Rijke (2013). New models have been added, the formal details of all models considered are included, and we report on substantially more elaborate experiments than in Liang and de Rijke (2013). The code to support the experiments in this paper is available from http://ilps.science.uva.nl/resources.

* Corresponding author. Tel.: +31684987399.
E-mail addresses: s.liang@uva.nl (S. Liang), derijke@uva.nl (M. de Rijke).
[1] Now at University College London.

http://dx.doi.org/10.1016/j.ipm.2015.11.005
0306-4573/

as to make a decision on whether the group is knowledgeable on the topic, whilst the approach of first locating an expert is mainly focused on collecting evidence to make a decision on whether the expert has expertise on the topic. Technically, there are important differences too. For instance, as we will see below, a group finding model that finds knowledgeable groups via documents directly is significantly outperformed by models that decompose the problem differently.

Traditional approaches to finding knowledge, whether in individuals or in groups within an organization, usually include two main steps. For a given task the expertise of the experts in a group is recorded and then the expertise of a group is computed by aggregating the expertise values of all group members. Both steps are traditionally done manually and require considerable effort to set up and maintain. In addition, this approach is usually restricted to a fixed set of expertise areas, making it hard to find knowledgeable groups in areas not explicitly coded (Pryor, Myles, Williams, & Anand, 1988).

To reduce the effort of recording and evaluating the expertise of people from their representations, many automatic approaches have been proposed. There has been an increasing move to automatically extract such representations for evaluating expertise from heterogeneous document collections, such as conference papers, corporate intranets and community question answering collections (Balog, Fang, de Rijke, Serdyukov, & Si, 2012).

To compute the expertise values of a group, in principle, many aggregation operators are available, such as maximum, sum, or average. These can simply be employed to combine the expertise values of each expert within a given group. There are at least 90 families of aggregation operators (Zhou, Chiclana, John, & Garibaldi, 2011); they have been put to use in a range of applications, e.g., in clustering (Beliakov, James, & Li, 2011), image segmentation (Ghosh, Kothari, Halder, & Ghosh, 2009), and control (Senge & Hullermeier, 2011). However, a solution to the problem of how to aggregate expertise values of all experts within a group so that the expertise scores of different groups can easily be compared and ranked by using a suitable aggregation operator, is still unknown.

We treat the problem of finding a knowledgeable group differently. Five distinct models are proposed. Our models are based on probabilistic language modeling techniques, which have been successfully applied in a range of related Information Retrieval (IR) tasks, such as ad hoc retrieval (Ponte & Croft, 1998; Zhai & Lafferty, 2004), expert finding (Balog et al., 2006; 2009; Balog et al., 2012; Fang, Si, & Mathur, 2010), similar people finding (Weerkamp et al., 2011), and republished article finding (Tsagkias, de Rijke, & Weerkamp, 2011). Language models are attractive because of their foundations in statistical theory, the great deal of complementary work on language modeling in speech recognition and natural language processing, and the fact that very simple language modeling applied to retrieval problems tends to perform very well empirically (Balog, Azzopardi, & de Rijke, 2009). Each group finding model that we consider ranks groups according to the probability of a group being a knowledgeable group given the query topic, but the models differ in how this is performed. Three types of variables play a key role in our estimations: groups (G), queries (Q) and documents (D). The order in which we estimate these is reflected in our naming conventions. E.g., the model named GDQ proceeds by first collecting evidence of whether a group is knowledgeable about the topic via the experts in the group (G), and then determining whether each expert in the group has expertise on the topic via documents (D), and finally whether a document is talking about the given query (Q) topic. In our Group-Query-Document (GQD) model and Group-Document-Query (GDQ) model, the expertise scores of each expert in a group are computed first and then aggregated into an overall score. These two models differ in the way in which they compute expertise scores for individual experts; in both cases, the experts in a group act as a latent variable between the group and the query. In our Document-Group-Query (DGQ) model, documents are ranked according to the query, and then we determine how likely a group is a knowledgeable group by considering the set of documents associated with them. Here, the documents act as a latent variable between the query and the group. Our last two models, the Query-Group-Document (QGD) model and the Query-Document-Group (QDG) model, rank groups according to the query, and then we determine how likely a person in the group is a knowledgeable expert by considering the set of documents associated with the expert; in these two models, it is the query that acts as a latent variable between the group and the experts in the group; we find that the QGD model actually yields the same ranking as the GQD model. Unlike early automatic group finding systems that tended to focus on specific document genres only, such as email (Campbell, Maglio, Cozzi, & Dom, 2003) or software and software documentation (Mockus & Herbsleb, 2002) to build profiles and find the entities, e.g., experts, our group finding algorithms can work on heterogeneous document genres and the profiles of groups and experts are not required to be given in advance.

For evaluation purposes, we use data from both the TREC 2005 and 2006 Enterprise tracks to create our test sets. As the data sets were created for expert finding (as opposed to knowledgeable group finding), some additional work is needed to turn them into a test set for group finding. We define three types of ground truth for our knowledgeable group finding task, implementing three readings of what makes a group a knowledgeable group. Familiar retrieval metrics such as NDCG, NDCG@k, MAP, and p@k are applied as evaluation metrics in our experiments. We perform a range of experiments to analyze our proposed knowledgeable group finding models, and find that some of our models perform similarly according to one metric but not according to another. E.g., GDQ and DGQ models are not statistically significantly different when using NDCG as a performance metric on our datasets; but when using MAP as our metric, the observed differences are statistically significant. Our main research goals in experimentation are to understand how the five models listed above compare.

In summary, the contributions of this paper are the following:

(i) We introduce a new information retrieval task: given a topic, find knowledgeable groups that have expertise on the topic.
(ii) We propose five language modeling approaches to tackle the challenge of automatically finding knowledge groups in heterogeneous document collections.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

(iii) For the purpose of providing evaluation resources for the group finding task, a data set is created based on a publicly available corpus used in the TREC Enterprise tracks[2] and three types of ground truth are defined.
(iv) We provide a detailed analysis of the performance of the proposed knowledgeable group finding models.

The remainder of the paper is organized as follows. Related work is discussed in Section 2. Then, in Section 3 we describe five ways of modeling the group finding task. Section 4 is devoted to smoothing strategies for our five models and to computing associations between experts and documents and between experts and groups. Next, we describe our ground truth for the task and present experimental evaluations of our group finding methods in Sections 5 and 6. We conclude in Section 7.

2. Related work

We distinguish between three directions of related work in this paper: group finding, expert finding and language models.

2.1. Group finding

In recent years, significant research efforts have been invested in locating a group of individuals in an organization, work that is usually called *group finding* or *team finding*. Yang, Chen, Lee, and Chen (2011) define their group finding problem as follows. The authors try to find a group of attendees familiar with a given activity initiator, and ensure each attendee in the group to have tight social relations with most of the members in the group (as determined using a social graph). Sozio and Gionis (2010) study a query-dependent variant of the community-detection problem, which they call the community-search problem: given a graph G, and a set of nodes in the graph as their input query, find a subgraph of G that contains the input query nodes and is densely connected. Other group finding problems have also been studied. Lappas et al. (2009) study the problem of given a task, a pool of individuals χ with different skills and a social network that captures the compatibility among these individuals, finding χ', a subset of χ, whose members together have all of the required skills to complete a given specific task and also have minimal communication costs among them. Kargar and An (2011) continue to study this problem by designing two communication cost functions for two types of communication structures. Neshati et al. (2014) and Li et al. (2013) tackle the problem of expert group formation (i.e., expert matching) to optimally assign a set of available experts to a project. Juang et al. (2013) propose two algorithms for the problem of finding an expertise team with a leader that has the required skills and minimal communication cost, where they make the strong assumption that a set of skills of each experts, the skills the project requires and the communication cost between each expert pairs are known in advance. Chen, Zeng, and Yuan (2013) present a matrix factorization based unified framework that recommends groups of users in a single system by examining their mutual contributions. Garcia and Sebastia (2014) address the group recommendation problem for a group of users where each user has special preferences and expectations about the resulting group profile.

What we do in this paper is about group finding, but the problem we deal with is different. We introduce a new group finding task: given a topic query, determine a list of knowledgeable groups within which the experts have expertise on the topic. Our group finding problem includes two sub-problems. The first sub-problem is to answer questions such as "Which groups are knowledgeable groups on topic T?" whilst the second sub-problem is to answer the question "What does group G know?" In this paper we focus exclusively on the first sub-problem, for which we cannot simply apply existing group finding algorithms.

2.2. Expert finding

Entity retrieval problems have been widely studied in the literature. For instance, given a topic query, search for specific entities such as people, products, or locations (Balog, Bron, & de Rijke, 2011). A specific instance of this task, one that is especially relevant to us, is to retrieve a list of people who have expertise on a given topic. Balog et al. (2006, 2009) present two general strategies to expert searching that are formalized using generative probabilistic models. In their first model, they compute scores of an expert's expertise based on the documents that the expert is associated with, whilst in their second model, they locate documents that are related to the query and then find the associated experts. Building on this, Fang et al. (2010) use some documents as training data in their relevance-based discriminative learning framework and derive specific discriminative models for expert retrieval. Tung et al. (2010) develop an expert retrieval system based on reasoning approaches and incorporate domain expertise into their methods with a role-based access control model to suggest appropriate experts for problem solving. However, the roles for reasoning are expensive to establish. To return a list of experts sorted by their level of expertise regarding the user query, Moreira and Wichert (2013) introduce an approach for combining multiple estimators of expertise based on a multi-sensor framework together with the Dempster–Shafer theory of evidence and Shannon's entropy. They define three sensors that detect heterogeneous information derived from textual content, from the graph structure of the citation patterns for the community of experts, and from profile information about academic experts. In practice, however, only the textual content is available in many scenarios, not the graph structure of each group or the profile information of the experts.

[2] http://www.ins.cwi.nl/projects/trec-ent.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

Initial manual approaches of finding knowledgeable groups first compute the scores of experts for a given topic and then aggregate the scores into an overall score for each group. Subsequently, groups are ranked and retrieved according to the scores of the groups (Pryor et al., 1988). Inspired by these approaches, we can tackle our knowledgeable group finding problem by aggregating the experts' scores of each group and then ranking the groups based on the aggregated scores. Today, more than 90 families of aggregation operators have been proposed (Zhou et al., 2011). Most aggregation algorithms excel in a very specific area only (Zhou et al., 2011), leaving the problem of how to aggregate expertise scores of experts in the same group to be tackled. In the IR literature, many models exist for aggregating ranked lists, with or without scores, so as to produce a new ranked list; however, obtaining an effective and robust setting for different aggregation ranking tasks is still quite difficult to achieve (Macdonald & Ounis, 2011). So, adapting existing aggregation algorithms to address our knowledgeable group finding task is not an easy way out.

2.3. Language models

In recent years, variants of language modeling approaches to information retrieval that deal with more variables than just queries and documents, or even queries, entities and documents, have attracted considerable attention (Cimiano, Schultz, Sizov, Sorg, & Staab, 2009; Gerani, Carman, & Crestani, 2010; Ko, Si, Nyberg, & Mitamura, 2010; Lv & Zhai, 2009; Sun, Wang, Sun, & Lin, 2011; Wang, Li, & Gao, 2010; Zhao & Yun, 2009). A novel positional language model (PLM) is proposed by Lv and Zhai (2009) for ad hoc document retrieval, the idea of which is to define a model for each so-called position of a document, and then score a document based on the scores of its PLMs. Zhao and Yun (2009) study the integration of term proximity information into the unigram language model. They propose a new proximity language model that views proximity centrality of query terms as the Dirichlet hyper-parameter that weights the parameters of the unigram document language model. Wang et al. (2010) propose a language modeling approach to Web document retrieval in which each document is characterized by a mixture model with components corresponding to the various text streams associated with the document (such as the body, the title and the URL of the document). To retrieve documents with opinions, Gerani et al. (2010) use a general opinion lexicon and propose an opinion propagation language model to calculate the opinion density at each point in a document. Sun et al. (2011) propose a language model based approach for tag recommendation that recommends tags by ranking them with their similarity to the given document and leverages the content information from both tag and document for ranking. Ko et al. (2010) present two probabilistic language models for answer ranking in the multilingual question-answering task, which finds exact answers to a natural language question written in different language.

Even though a large number of effective language models have been proposed that aggregate scores of smaller units (positions, paragraphs, …) into scores of larger units, almost all of them are aimed at dealing with a specific IR problem of aggregating evidence obtained from specific textual units. To the best of our knowledge, specific language models for our knowledgeable group finding task have not been introduced yet.

3. Modeling group finding

In this section, we describe our approaches to modeling the group finding task in detail. We first provide some background to language modeling applied to entity retrieval and then introduce and analyze our group retrieval task. We pay considerable attention to modeling our task in five distinct ways.

During the past decade, research on entity retrieval and entity profiling has generated considerable interested from the IR community (Balog et al., 2011; Fang et al., 2010; Lv & Zhai, 2009). Relatively simple and transparent language modeling-based approaches have performed well on the tasks for which they were defined. In these approaches, one usually ranks entities based on the estimated language models, which are either estimated from the documents or the queries. In our modeling of the knowledgeable group finding task, multiple kinds of documents in a heterogeneous collection are used to collect evidence for expertise of each group. Groups, documents and queries are considered in different orders to estimate our language models. Groups are ranked according to how likely the groups have expertise on the given query according to the estimated language model.

3.1. Problem definition and context

We address the following ranking problem: given a query topic, identify knowledgeable groups that have expertise on that topic. We formulate the problem as follows: what is the probability of a group g being a knowledgeable group given query topic q? That is, we have to estimate the probability of a group g given a query q and then rank groups according to their probabilities. The top k groups will be considered to be the most knowledgeable groups for the given query topic. Instead of computing this probability directly, we apply Bayes' Theorem, and obtain

$$p(g|q) = \frac{p(q|g)p(g)}{p(q)},$$

where $p(q)$ is the probability of a query and $p(g)$ is the probability of a group. The priori probability $p(q)$ is the same when searching knowledgeable groups with the same query; thus, we can set $p(q)$ to be a constant. The same choice can be made

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

for the priori probability $p(g)$, i.e., $p(g)$ is a constant. Hence, ranking groups according to the probability of a group given a query topic $p(g|q)$ boils down to ranking a query topic given a group $p(q|g)$, i.e.,

$$p(g|q) \stackrel{rank}{=} p(q|g).$$

In the remainder of this section, we detail our proposed five knowledgeable group finding language models. To determine $p(g|q)$ or $p(q|g)$ we consider experts, groups, documents and queries in different orders, and adapt generative probabilistic language modeling techniques, so as to arrive at five distinct models. Two of these models (the GQD and GDQ models; see Section 3.2) aggregate expertise scores of each expert in the same group to an overall expertise score for the group to which they belong. In the DGQ model (see Section 3.3) we collect evidence of how likely a group of people has expertise on the given query topic via heterogeneous documents in the repository. Two remaining models, the QDG and QGD models (see Section 3.4), directly estimate the degree to which a group is a knowledgeable group for a given topic.

3.2. Two aggregation models: GQD and GDQ

We have two types of aggregation model: the Group-Query-Document (GQD) model and the Group-Document-Query (GDQ) model. The order of the key terms in these names signifies the following: GQD indicates that the evidence of whether a group is a knowledgeable group on the topic is collected via the experts in the group (G), then how likely each expert in the group has expertise on the query (Q) is computed via the documents (D). GDQ indicates that the evidence of whether a group is a knowledgeable group on the topic is collected via the experts in the group (G), then via each document (D) the expertise of each expert in the group on the query (Q) topic is computed directly via the documents. In both models, experts in the same group g are conditionally independent given the group, such that

$$p(g|q) = \prod_{ex \in g} p(ex|q)^{as(ex,g)},$$

where ex is an expert belonging to group g, $p(ex|q)$ is the probability of an expert ex given a query q, and $as(ex, g)$ is the association between an expert ex and the group g. Instead of computing $p(ex|g)$ directly, we apply Bayes' Theorem, and obtain

$$p(ex|q) = \frac{p(q|ex)p(ex)}{p(q)},$$

where $p(q|ex)$ is the probability of a query given an expert, $p(ex)$ is the probability of an expert, and $p(q)$ is the probability of the query. As each expert is a common member in a group, $p(ex)$ can be set to be constant. Additionally, for each query topic, $p(q)$ is the same, hence, $p(ex|q)$ is proportional to $p(q|ex)$. So, $p(g|q)$ can be represented as follows

$$p(g|q) \stackrel{rank}{=} \prod_{ex \in g} p(q|ex)^{as(ex,g)}.$$

The GQD model. To obtain $p(q|ex)$, we assume that each term t in the query q is conditionally independent given expert ex, such that

$$p(q|ex) = \prod_{t \in q} p(t|ex)^{n(t,q)},$$

where $p(t|ex)$ is the probability of a term given an expert and $n(t, q)$ is the number of occurrences of term t in query q. Putting things together, we can rewrite $p(g|q)$ as follows

$$p(g|q) \stackrel{rank}{=} \prod_{ex \in g} \left\{ \prod_{t \in q} p(t|ex)^{n(t,q)} \right\}^{as(ex,g)}.$$

To obtain $p(t|ex)$, we take the sum over all documents d in the collection. Formally, this can be expressed as

$$p(t|ex) = \sum_{d} p(t|d)p(d|ex),$$

where $p(t|d)$ is the probability of term t given document d, and $p(d|ex)$ is the probability of document d given expert ex. Now we can obtain the probability of a group given a query

$$p(g|q) \stackrel{rank}{=} \prod_{ex \in g} \left\{ \prod_{t \in q} \left\{ \sum_{d} p(t|d)p(d|ex) \right\}^{n(t,q)} \right\}^{as(ex,g)} \tag{1}$$

This is our GQD model. In brief, it first tries to aggregate expertise scores of each expert in the same group for a certain query. Then the expertise of an expert is computed based on how likely this expert would produce the given query, which can be interpreted as the probability of this expert talking about this topic. Finally, the evidence of how much an expert

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

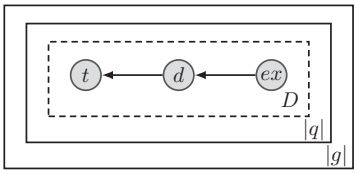


Fig. 1. Graphical representation of the Group-Query-Document model.

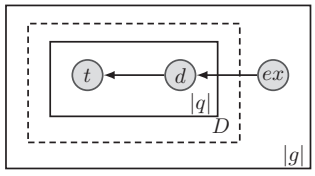


Fig. 2. Graphical representation of the Group-Document-Query model.

knows about the topic is collected from all the documents in the collection. The model in (1) first considers experts in a group, then terms in the given query, and finally the documents; this is why we call it the GQD model. The following group finding language models have similar naming conventions. The construction of the GQD model can be viewed as the following generative process: the group g is generated by query q by first generating a set of documents associated with each expert in the group, then generating each term in q from these documents, and finally generating the group by considering each expert in the group. A graphical representation of the GQD model is shown in Fig. 1.

The GDQ model. We can compute the probability of a query q given an expert ex in a different way from what we did previously. By taking the sum over all documents d, $p(q|ex)$ can be obtained. Formally, this can be expressed as

$$p(q|ex) = \sum_{d} p(q|d)p(d|ex),$$

where $p(q|d)$ and $p(d|ex)$ are the probability of query q given document d and the probability of document d given expert ex, respectively. Therefore, $p(g|q)$ can be rewritten as

$$p(g|q) \overset{rank}{=} \prod_{ex \in g} \left\{ \sum_{d} p(q|d)p(d|ex) \right\}^{as(ex,g)}.$$

To obtain $p(q|d)$, we assume again that each term t in the given query q is conditionally independent given a document d, such that

$$p(q|d) = \prod_{t \in q} p(t|d)^{n(t,q)}.$$

Based on this, we obtain our second aggregation model

$$p(g|q) \overset{rank}{=} \prod_{ex \in g} \left\{ \sum_{d} \left\{ \prod_{t \in q} p(t|d)^{n(t,q)} \right\} p(d|ex) \right\}^{as(ex,g)}. \tag{2}$$

This is our GDQ model. It first tries to aggregate expertise scores of each expert in the same group for a certain query, but the way of aggregating expertise scores is different from that in the GQD model. To aggregate scores, the GDQ model collects the evidence of how likely it is that an expert knows about the given topic via all documents in the collection, and then computes how likely the documents are relative to the given query. The construction of the GDQ model can be understood as the following generative process: the group g is generated by query q by first generating a set of documents associated with each expert in the group, and for each of these documents generating the query q from the document and considering the probability of how likely the expert can generate the document. A graphical representation of the GDQ model is shown in Fig. 2.

3.3. A document model: DGQ

Instead of thinking about aggregating expertise scores of all the experts within a group as in our aggregation models, the probability $g(g|q)$ can also be computed directly via the documents (D), as the order of the key terms in the model DGQ's

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

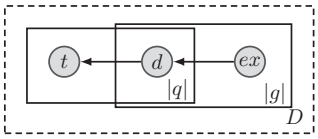


Fig. 3. Graphical representation of the Document-Group-Query model.

name indicates. For each document we then compute how likely the group (G) is associated with it, and how likely it is talking about the given query (Q) topic, such that

$$p(g|q) = \sum_{d} p(g|d)p(d|q),$$

where $p(g|d)$ and $p(d|q)$ are the probability of group g given document d and the probability of d given query q, respectively. We assume that all experts in the same group g are conditionally independent given document d. So $p(g|q)$ can be rewritten as follows.

$$p(g|d) = \prod_{ex \in g} p(ex|d)^{as(ex,g)},$$

where $p(ex|d)$ is the probability of an expert belonging to group g given document d. In order to compute $p(d|q)$, we apply Bayes' Theorem, and obtain

$$p(d|q) = \frac{p(q|d)p(d)}{p(q)},$$

where $p(d)$ is the probability of document d, which is set to be uniform. As $p(q)$ is a constant for a given query q, the term $p(d|q)$ is proportional to $p(q|d)$ and can be expressed as: $p(d|q) \propto p(q|d)$. So $p(g|q)$ can be represented in the following manner

$$p(g|q) \stackrel{rank}{=} \sum_{d} p(g|d)p(q|d),$$

which indicates that ranking according to $p(g|q)$ means we have to collect the evidence of how a document can generate the given query, and how likely the group is associated with the document, via all documents in the collections. To obtain the probability of query q given document d, we again assume that each term in the query is conditionally independent, such that

$$p(q|d) = \prod_{t \in q} p(t|d)^{n(t,q)}.$$

Again, for a given document, each expert ex in the same group g is conditionally independent, and $p(ex|d)$ is proportional to $p(d|ex)$ when applying Bayes' Theorem, resulting in $p(g|d)$ being expressed as

$$p(g|d) = \prod_{ex \in g} p(ex|d)^{as(ex,g)},$$

where $p(ex|d)$ is the probability of an expert given a document, which measures how likely the document belongs to the expert. We apply Bayes' theorem, and obtain $p(ex|d) = p(d|ex)p(ex)p(d)^{-1}$, where both $p(ex)$ and $p(d)$ are fixed for each expert ex and each document d in the collection, respectively. So $p(ex|d)$ is proportional to $p(d|ex)$, and then $p(g|d)$ can be represented as

$$p(g|d) \stackrel{rank}{=} \prod_{ex \in g} p(d|ex)^{as(ex,g)}.$$

This, then, is how $p(g|q)$ can be represented

$$p(g|q) \stackrel{rank}{=} \sum_{d} \left\{ \prod_{ex \in g} p(d|ex)^{as(ex,g)} \right\} \left\{ \prod_{t \in q} p(t|d)^{n(t,q)} \right\}. \tag{3}$$

This is our DGQ model. For each document, we compute $p(g|d)$ and $p(q|d)$ at the same time. It assumes that all documents in the collection are not only associated with each expert in the same group but also with the given query. So this model tries to collect evidence of the probability of an expert to "own" the document and how likely the document talks about the given query. The construction of the DGQ model can be viewed as the following generative process: the group g is generated by query q by first generating the query from each document in the collection and at the same time the document is generated by each expert in the group. A graphical representation of the DGQ model is shown in Fig. 3.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

3.4. Two query models: QDG and QGD

Next, we present two query models for the knowledgeable group finding task. Before we start, two comments are in order. First, we actually define two models – QGD and QDG – , but one of them (QGD) will be shown to coincide with one of the aggregation models (GQD). Second, we use "query model" not in the sense of building rich representations of the query but to indicate that our estimations of a group finding model start with the query. As the names QGD and QDG indicate, both consider how likely a group knows about a query (Q) topic first. The order QGD signifies that this can be computed by first determining how likely it is that each expert in the group (G) has expertise on the topic via the documents (D), whilst QDG computes this via documents (D) and then determines how likely each expert in the group (G) is associated with each document. For both models, we first apply Bayes' Theorem, so that ranking according to the probability $p(g|q)$ is equivalent to ranking according to $p(q|g)$:

$$p(g|q) \overset{rank}{=} p(q|g),$$

where $p(q|g)$ is the target to be computed. We assume that terms in the given query q are conditionally independent given a group g, such that

$$p(q|g) = \prod_{t \in q} p(t|g)^{n(t,q)},$$

where $p(t|g)$ is the probability of a term t given a group g. We compute $p(t|g)$ in two ways.

The QGD model. Instead of computing $p(t|g)$ directly, we apply Bayes' Theorem, and obtain $p(t|g) = p(g|t)p(t)p(g)^{-1}$, where $p(g|t)$ is the probability of group g given term t. As $p(t)$ is a constant once term t in query q is selected and $p(g)$ is assumed constant for each group, $p(t|g)$ is proportional to $p(g|t)$:

$$p(t|g) \overset{rank}{=} p(g|t).$$

Each expert ex in the same group is conditionally independent for a given term t, we can obtain

$$p(g|t) = \prod_{ex \in g} p(ex|t)^{as(ex,g)},$$

where $p(ex|t)$ is the probability of an expert given term t. In sum, we can rewrite $p(g|q)$ as follows:

$$p(g|q) \overset{rank}{=} \prod_{t \in q} \left\{ \prod_{ex \in g} p(ex|t)^{as(ex,g)} \right\}^{n(t,q)}.$$

To compute $p(ex|t)$, we collect evidence of how likely expert ex is talking about a topic associated with the term t via all the documents in the collection, such that

$$p(ex|t) = \sum_{d} p(ex|d)p(d|t),$$

where $p(ex|d)$ is the probability of expert ex given document d and $p(d|t)$ is the probability of document d given term t. Instead of computing $p(d|t)$, we apply Bayes' Theorem again, and obtain $p(d|t) = p(t|d)p(d)p(t)^{-1}$, where $p(d)$ is the probability of document d, which is assumed to be uniform. In addition, $p(t)$ is fixed once the query is given, so we obtain the following formula

$$p(d|t) \overset{rank}{=} p(t|d)$$

Now, the final version of $p(g|q)$ can be represented as

$$p(g|q) \overset{rank}{=} \prod_{t \in q} \left\{ \prod_{ex \in g} \left\{ \sum_{d} p(d|ex)p(t|d) \right\}^{as(ex,g)} \right\}^{n(t,q)}. \tag{4}$$

This is our QGD model. It first computes the probability how likely a group is talking about a given query topic. Then in each group, for each expert we consider how likely he or she is talking about the query topic. Finally, the expertise of each expert within a group is collected via all documents in the collection. The construction of the QGD model can be viewed as the following generative process: the group g is generated by query q by first considering each document in the collection, generating a term from the document and how likely an expert in the group generates the document, and finally generating the query by the group. A graphical representation of the QGD model is shown in Fig. 4.

It is worth pointing out that QGD and GQD are the same models, although the ways of finding knowledgeable groups are different, as both (4) (the QGD model) and (1) (the GQD model) can be represented as, and rank groups equivalently to

$$p(g|q) \overset{rank}{=} \log p(g|q) = \sum_{t \in q} \sum_{ex \in g} n(t,q) \cdot as(ex,g) \cdot \log \sum_{d} p(d|ex)p(t|d).$$

Hence, in our experimental evaluation in Section 6, we only consider GQD; the results for QGD are exactly the same.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

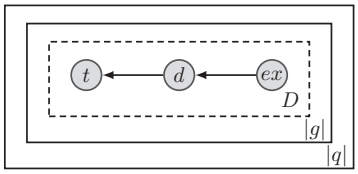


Fig. 4. Graphical representation of the Query-Group-Document model.

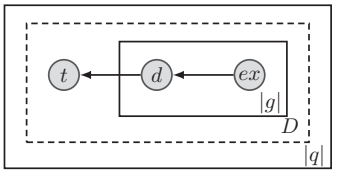


Fig. 5. Graphical representation of the Query-Document-Group model.

The QDG model. Instead of computing how likely an expert is talking about the given topic, we compute $p(t|g)$ in an alternative way. We collect the evidence of how group g being a knowledgeable group via all documents in the collection, and obtain

$$p(t|g) = \sum_{d} p(t|d)p(d|g),$$

where $p(d|g)$ is the probability of document d given group g. Again, instead of computing $p(d|g)$ directly, we apply Bayes' Theorem, and obtain $p(d|g) = p(g|d)p(d)p(g)^{-1}$, where both $p(d)$ and $p(g)$ are fixed values for each document and each group. So $p(g|q)$ can be represented as follows

$$p(g|q) \overset{rank}{=} \prod_{t \in q} \left\{ \sum_{d} p(t|d)p(g|d) \right\}^{n(t,q)},$$

where $p(g|d)$ is the probability of group g given document d. Each expert ex in the same group is conditionally independent for a given document d, we can represent $p(g|d)$ as

$$p(g|d) = \prod_{ex \in g} p(ex|d)^{as(ex,g)},$$

where $p(ex|d)$ is the probability of an expert ex given a document d, which is proportional to $p(d|ex)$.

Now, the final version of $p(g|q)$ can be represented as

$$p(g|q) \overset{rank}{=} \prod_{t \in q} \left\{ \sum_{d} p(t|d) \prod_{ex \in g} p(d|ex)^{as(ex,g)} \right\}^{n(t,q)}. \tag{5}$$

And this is our QDG model. It first computes the probability of how likely a group is talking about a given query topic just as in QGD model, but the way of computing this is different. The QDG model collects evidence of how likely the group is a knowledgeable group for a given query via all documents in the collection. For each expert within a group, we determine how likely the expert is associated with the documents. The construction of the QDG model can be viewed as the following generative process: the group g is generated by query q by first generating a document from each expert in the group, generating a term in the query by a set of the document associated with the group, and finically generating the query. A graphical representation of the QDG model is shown in Fig. 5.

This concludes the introduction of our group finding models. Before we can turn to an experimental comparison, we need to settle two things: estimating associations and smoothing.

4. Associations and smoothing

In this section, we detail how we estimate the probability that an expert ex in a group g is associated with a document d and the probability of an expert ex being associated with the group he/she belongs to. In addition, we devise strategies for smoothing our five knowledgeable group finding models.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

4.1. Expert-document associations

For all models described in the previous section, we need to be able to estimate the probability of an expert *ex* in group *g* being associated with document *d*. In recent years, this problem has attracted considerable attention (Balog et al., 2006; Balog, Bogers, Azzopardi, de Rijke, & van den Bosch, 2007; Balog et al., 2011; Balog & de Rijke, 2008; Fang et al., 2010). Following Balog et al. (2006), to define this probability, we assume that associations $a(d, ex)$ between experts *ex* and documents *d* have been calculated and define

$$p(d|ex) = \frac{a(d, ex)}{\sum_{d' \in \mathcal{D}} a(d', ex)}, \tag{6}$$

where $\mathcal{D}$ is the set of documents in the collection, and $a(d, ex)$ can simply be defined as

$$a(d, ex) = \begin{cases} 1 & rel(ex, d) = 1 \\ 0 & \text{otherwise}, \end{cases} \tag{7}$$

where $rel(ex, d) = 1$ if the full name or email address of expert *ex* (exactly) appears in document *d*, otherwise $rel(ex, d) = 0$. Research by Balog et al. (2006); Balog et al. (2007) on the task of finding experts shows that using more sophisticated ways of recognizing expert information to compute $a(d, ex)$ may boost the effectiveness of identifying an expert given a query topic. As our focus in this paper is on modeling group finding, we do not define $a(d, ex)$ as in (Balog et al., 2006) and just use the simple but effective definition given above.

4.2. Group-expert associations

For all of the group finding models described in the previous section, we also need to be able to estimate the strength of the association between an expert *ex* and the group *g* to which the expert belongs. We define the following group expert association

$$as(ex, g) = \frac{1}{|g|}, \tag{8}$$

where $|g|$ is the total number of experts within the group *g* to which they belong. This is a baseline way to obtain an association score between group and expert, which simply indicates that each expert is an equal member of the group to which they belong.

4.3. Smoothing strategies

In our five knowledgeable group finding models, the term $p(g|q)$ may contain zero probabilities due to data sparsity. For instance, in our aggregation models, GQD and GDQ, $p(g|q)$ will contain zero probabilities if there exist experts who have no expertise on the given query, i.e., if for *ex* in *g* we have that $p(q|ex) = 0$ is true, then $p(g|q) = 0$. Therefore, we have to infer a group model θ_g, such that the probability of a group given a query model is $p(\theta_g|q)$. Many smoothing methods have been proposed and used in language modeling (Zhai & Lafferty, 2004). We employ Jelinek–Mercer smoothing method (Jelinek & Mercer, 1980) to estimate $p(\theta_g|q)$; we consider two types.

Two-parameter smoothing. To facilitate comparisons and for the sake of uniformity, in all of the five models, instead of estimating $p(g|q)$ directly, we can easily infer a document model θ_d such that the probability of a term *t* given a document *d* model is $p(t|\theta_d)$, and infer an expert model θ_{ex} such that the probability of a document *d* given an expert *ex* is $p(d|\theta_{ex})$. The document model is then constructed as a linear interpolation of the background model $p(t)$ and the smoothed estimate

$$p(t|\theta_d) = (1 - \alpha)p(t|d) + \alpha p(t),$$

where α is a smoothing parameter ($0 < \alpha < 1$). The expert model is also constructed as a linear interpolation of the background model $p(d)$ and the smoothed estimate

$$p(d|\theta_{ex}) = (1 - \beta)p(d|ex) + \beta p(d),$$

where β is a smoothing parameter ($0 < \beta < 1$). Let $\theta(\alpha, t, d)$ be short for $p(t|\theta_d) = (1 - \alpha)p(t|d) + \alpha p(t)$, and $\vartheta(\beta, d, ex)$ be short for $p(d|\theta_{ex}) = (1 - \beta)p(d|ex) + \beta p(d)$. Then, the group finding model GQD (and model QGD) can be smoothed and estimated as

$$p(g|q) \overset{rank}{=} \prod_{ex \in g} \left\{ \prod_{t \in q} \left\{ \sum_{d} \theta(\alpha, t, d) \cdot \vartheta(\beta, d, ex) \right\}^{n(t,q)} \right\}^{as(ex,g)}. \tag{9}$$

Similarly, the GDQ model can be smoothed and estimated as

$$p(g|q) \overset{rank}{=} \prod_{ex \in g} \left\{ \sum_{d} \left\{ \prod_{t \in q} \theta(\alpha, t, d)^{n(t,q)} \right\} \vartheta(\beta, d, ex) \right\}^{as(ex,g)}. \tag{10}$$

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

The DGQ model can be smoothed and estimated as

$$p(g|q) \overset{rank}{=} \sum_{d} \left\{ \prod_{ex \in g} \vartheta(\beta, d, ex)^{as(ex,g)} \prod_{t \in q} \theta(\alpha, t, d)^{n(t,q)} \right\}. \tag{11}$$

Finally, the QDG model can be smoothed and estimated as

$$p(g|q) \overset{rank}{=} \prod_{t \in q} \left\{ \sum_{d} \theta(\alpha, t, d) \prod_{ex \in g} \vartheta(\beta, d, ex)^{as(ex,g)} \right\}^{n(t,q)}. \tag{12}$$

One-parameter smoothing. It is worth pointing out that for the GQD model (see (1)), $p(g|q)$ can be smoothed and estimated in a different way, so that a single smoothing parameter suffices. Instead of smoothing both $p(t|d)$ and $p(d|ex)$, we can infer another expert model Θ_{ex} such that the probability of a term t given an expert ex is $p(t|\Theta_{ex})$. This expert model is then constructed as a linear interpolation of the background model $p(t)$, and the smoothed estimate: $p(t|\Theta_{ex}) = (1-\lambda)\, p(t|ex) + \lambda p(t)$, where λ is a smoothing parameter ($0 < \lambda < 1$). When we use $p(t|\Theta_{ex})$, the GQD model (QGD model) can be smoothed and estimated as

$$p(g|q) \overset{rank}{=} \prod_{ex \in g} \left\{ \prod_{t \in q} \left\{ (1-\lambda) \sum_{d} p(t|d) p(d|ex) + \lambda p(t) \right\}^{n(t,q)} \right\}^{as(ex,g)}. \tag{13}$$

5. Experimental setup

In this section, we describe the experimental setup for testing our knowledgeable group finding methods. We describe our dataset, detail the different types of ground truth that we consider, and specify our research questions.

5.1. Experimental collection

For evaluation purposes we use data made available for the expert finding task at the TREC 2005 and 2006 Enterprise tracks (Balog et al., 2009; Craswell, de Vries, & Soboroff, 2005; Soboroff, de Vries, & Craswell, 2006). For these tracks, the document collections used are the same: a crawl of the World Wide Web Consortium (W3C), a heterogenous document repository containing a mixture of document types.[3] The six types of pages in the crawl are lists (email forum; 198,394 documents), dev (code; 62,509 documents), www (web; 45,975 documents), esw (wiki; 19,605 documents), other (miscellaneous; 3,538 documents), and people (personal home pages; 1,016 documents). In total, the W3C corpus contains 331,037 documents, adding up to 5.7GB.

We use the test topics and the ground truth made available by the TREC Enterprise 2005 track to build knowledgeable groups, where the names of groups are the test topics (50 test topics) in the TREC Enterprise 2005 track, resulting in 50 knowledgeable groups, and the expert-group pairs are generated based on the ground truth of the TREC 2005 track, resulting in 1509 expert-group pairs, with between 2 and 391 experts in the same group and approximately 30 experts per group on average. In other words, if an expert is related to a test topic, then this expert is one of the members in the group with the name of the test topic. In addition, after building the knowledgeable groups' information based on TREC Enterprise 2005 track data, we use the TREC Enterprise 2006 track to test the performance of our five proposed knowledgeable group finding models. For the TREC Enterprise 2006 track, 55 queries were created, but only 49 are provided with expert finding ground truth. The 2006 expert finding queries will be used to evaluate our group finding task and the ground truth for the group finding task is created based on the TREC Enterprise 2006 track; see below.

5.2. Three types of ground truth

So far, different ways to construct specific ground truths for the evaluation of the group finding have been considered. Lappas et al. (2009) and Kargar and An (2011) use the authorship information in academic papers to construct groups and use information about communication cost (see Lappas et al., 2009 for details) to evaluate their experimental results. In our dataset not all of the information used in Lappas et al. (2009), e.g., authorship information in each group, the skill information of each expert, communication costs between experts, is present. Hence, we cannot follow their way of construct our ground truth. Neshati et al. (2014) suppose that each expert's expertise is clear and they prefer a group of experts that can cover as many different areas of expertise as possible. We cannot follow this way to construct our group finding ground truth either because experts' expertise is unknown in our dataset. Thus, we construct our ground truth in a different way, viz. by using the ground truth of the TREC 2006 expert finding task. We propose three types of ground truth for our group finding task: *binary, graded* and *number*.

[3] http://www.w3c.org.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

Binary. A group g is considered relevant for topic q if there is at least one expert ex who is a member of g (according to the TREC 2005 expert finding ground truth) and has expertise on the topic q (according to the TREC 2006 expert finding ground truth). This is a weak notion of group relevance.

Graded. A slightly more evolved definition of group relevance uses grades: the level of relevance of group g for query q is defined based on the fraction of the experts in the group. We distinguish between $|L|$ different levels of relevance, i.e., $\{0, 1, 2, \ldots, |L-1|\}$. The relevance grade $l \in \{0, 1, 2, \ldots, |L-1|\}$ of group g for topic q is defined as follows. Let

$$f(g,q) = \frac{|\{ex \in g : rel(ex,q) = 1\}|}{|g|},$$

where $\{ex \in g : rel(ex,q) = 1\}$ is the set of experts in g with expertise on topic q according to the TREC 2006 expert finding ground truth and $|g|$ is the total number of experts in group g. If $\frac{1}{|L|} \cdot l \leq f(g,q) < \frac{1}{|L|} \cdot (l+1)$, the grade level for this group is l. In this paper, we set $|L| = 10$.

Number. Here, the level of relevance of group g for query q is defined based on the number of experts in the group. For instance, if there are 15 experts who have expertise on the given query topic, then the level of the relevance for this group is 15. The level of relevance ranges from 0 to 30 with a majority smaller than 4.

These three types of ground truth allow for different uses in the evaluation of the knowledgeable group finding task: the binary ground truth only allows us to determine whether there is at least one member in a group having expertise on a given topics; the graded ground truth considers not only whether there are experts in the group about a topic but also how many of them have expertise on the topic. In contrast, the number ground truth may favor larger knowledgeable groups because a larger group probably will have a larger number of experts than a very small group.

5.3. Runs

We run our experiments with all of the documents in the collection for our five knowledgeable group finding models. We perform a grid search to find optimal settings of the smoothing parameters (with 0.1 increments).

In addition to using the full collection for retrieval and estimation, we also generate runs for our group finding task based on a subset of documents defined by taking the top n documents returned by a standard document retrieval language model (Zhai & Lafferty, 2004) when using the topic as query, and then compare the performance results of the proposed group finding models.

5.4. Evaluation metrics

The evaluation metrics used for the group finding task are *normalized discounted cumulative gain* (NDCG) and NDCG@5, 10 (Clarke et al., 2008), *mean average precision* (MAP) (Manning, Raghavan, & Schütze, 2008), and *precision*@5, 10 (Manning et al., 2008) against our three types of group finding ground truth. MAP scores are of special interest to us: we hypothesize that the models have both a precision and recall-enhancing effect and we use MAP to measure this. We adopt precision@5, 10 as they are the official evaluation metrics used to assess expert finding in the TREC 2005 and 2006 Enterprise tracks that our dataset is built on. NDCG and NDCG@5, 10 are also of special interest to us: we want to know whether more relevant knowledgeable groups can be ranked higher than less relevant groups by the models we consider. Evaluation scores were computed done using the trec_eval program.[4]

NDCG. Given a ranked result set of documents (in our setting, groups) $\mathcal{S}$ and an ideal ordering of the same set of documents $\mathcal{R}$, the *discounted cumulative gain* (DCG) (Clarke et al., 2008) at a particular rank threshold k is defined as

$$\mathrm{DCG}(\mathcal{S},k) = \sum_{j=1}^{k} \frac{2^{r(j)}-1}{\log(1+j)},$$

where $r(j)$ is the judgment ($0 = \mathrm{Bad},\ 1 = \mathrm{Fair},\ 2 = \mathrm{Good},\ 3 = \mathrm{Excellent}$, etc.) at rank j in set $\mathcal{S}$. The ideally ordered set $\mathcal{R}$ contains all documents rated for the given query sorted descending by the judgment value. Then the *normalized discounted cumulative gain* (NDCG) (Clarke et al., 2008) at a particular rank threshold k is defined as

$$\mathrm{NDCG}(\mathcal{S},k) = \frac{\mathrm{DCG}(\mathcal{S},k)}{\mathrm{DCG}(\mathcal{R},k)}.$$

NDCG discounts the contribution of a document to the overall score as its rank increases. NDCG value at rank threshold k when the set $\mathcal{S}$ is clear from the context is often written as NDCG@k.

MAP. The *mean average precision* (MAP) (Manning et al., 2008) of a test query set is the mean of the *average precision* (AP) values of all queries in the query set. The average precision of a ranked result set in response to a given query is defined as

$$\mathrm{AP} = \frac{\sum_{j=1}^{k} P(j) * \mathrm{Relevance}(j)}{\sum_{j=1}^{k} \mathrm{Relevance}(j)},$$

[4] The trec_eval program is available from the TREC web site http://trec.nist.gov.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

where j is the position of the document (in our case, group), Relevance(j) denotes the relevance of the document (in our case, group) in position j, and $P(j) = \sum_{i=1}^{j} \text{Relevance}(i)/j$. Typically, a binary value for Relevance(j) is used by setting it to 1 if the document (group) in position j has a human judgment of Fair or better and 0 otherwise.

Precision. *Precision* at k (Manning et al., 2008) only considers the total number of relevant documents ranked within the top k positions and can be simply computed as

$$\text{p@}k = \frac{\#\text{relevant documents among the top } k}{k},$$

where "relevant documents" are those that have human judgments of Fair or better.

5.5. Research questions

Our experiments are meant to address the following groups of questions:

(i) How do different optimal group finding models with two smoothing parameters perform when compared against each other? Do relative differences between models change depending on the type of group finding ground truth used or when using different evaluation metrics? And what are the optimal settings of the two smoothing parameters? (See Section 6.1.)
(ii) How sensitive are the group finding models that use two smoothing parameters to the settings of those parameters? (See Section 6.2.)
(iii) How does the GQD model behave with only one smoothing parameter compared with two parameters and what are the optimal parameters? (See Section 6.3.)
(iv) How does the performance per query vary, for a given model? (See Section 6.4.)
(v). What is the effect of using a topically focused subset of documents on the performance on the group finding task? (See Section 6.5.)
(vi) And finally, given that the models all use the same basic components, are they really different from each other and what are the effect sizes for the models' comparisons? (See Section 6.6)

6. Results and analysis

In this section, we present and analyze our experimental results. We start by comparing the results of the optimized models and follow with an analysis of smoothing with two parameters vs. with a single parameter. Next, we examine performance differences across queries and present the results of using a topically focused set of documents. Finally, we test whether the models are statistically significantly different and provide the effect sizes for comparison.

6.1. Model comparison

How do our knowledgeable group finding models perform compared to each other? In the following set of experiments, for each specific performance evaluation metric, we compare the models using optimal smoothing parameters. We use two parameters α and β to smooth the proposed five knowledgeable group finding models, i.e., GQD, GDQ, DGQ and QDG, which were formalized in (9)–(12), respectively. Below, we do not report experimental results for the QGD model, as its results are the same as those of the GQD model; see Section 3.4.

Table 1 lists the scores for the various metrics. Clearly, DGQ outperforms the other models on all metrics using the binary and graded ground truth, but GDQ outperforms DGQ on all metrics using the number ground truth. (We further show whether the observed differences between any two group finding models are statistically different via a two-tailed paired t-test later in Section 6.6.) The QDG model is the worst performing model for all the metrics and against all types of ground truth. The table also shows that GQD, GDQ and DGQ have a similar performance for all the metrics against all types of ground truth. (The MAP and p@N scores against the number ground truth are the same as those against the binary ground truth, which is why we do not report them in the table.) Of course, they are built on similar language modeling components and the experimental outcomes suggest that but the differences in computing order are mostly immaterial.

To further illustrate the performance of our proposed group finding models, we also use a 5-fold validation strategy to do the experiments, i.e., we use a 4/1 split for our training and test sets, respectively. We train the models using values of the two smoothing parameters α and β that vary from 0.1 to 0.9. The best smoothing parameters are then chosen on the test set, and evaluated on the test queries. The train/test splits are permuted until all the queries have been chosen once for the test set. We repeat the experiments 10 times. Table 2 show the NDCG, NDCG@5, 10, MAP, p@5, and 10 evaluation results. As is shown in the table, the performance is similar to that in Table 1, which demonstrates that the proposed models are robust.

6.2. Smoothing with two parameters

Next we turn to smoothing with two parameters. To understand how the two smoothing parameters influence the performance of our models, we first change the smoothing parameter α from 0.1 to 0.9 with 0.1 steps and report the best

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

Table 1
Evaluation results using data from the TREC Enterprise 2006 track for all optimal models with two smoothing parameters, using the binary, graded as well as number ground truths. For each metric, we report the evaluation results, followed by the two optimal smoothing parameters α and β. We do not report the evaluation results for MAP and p@5, 10 when using the number ground truth, as they are the same as when using the binary ground truth.

					NDCG@					
Ground truth	Model	NDCG	α	β	5	α	β	10	α	β
Binary	GQD	.8861	.1	.2	.8165	.1	.2	.7850	.1	.3
	GDQ	.9009	.1	.9	.8291	.3	.5	.7936	.2	.5
	DGQ	**.9133**	.1	.9	**.8680**	.1	.9	**.8420**	.1	.9
	QDG	.7623	.1	.2	.5604	.1	.1	.5568	.1	.2
Graded	GQD	.8245	.2	.3	.7237	.2	.3	.7595	.1	.2
	GDQ	.8457	.1	.4	.7675	.2	.3	.7945	.1	.4
	DGQ	**.8631**	.2	.9	**.7991**	.5	.9	**.8160**	.7	.8
	QDG	.3916	.1	.8	.1012	.1	.1	.1282	.1	.2
number	GQD	.7964	.1	.9	.6210	.1	.8	.6730	.6	.8
	GDQ	**.8160**	.1	.9	**.6496**	.1	.9	**.6905**	.1	.9
	DGQ	.7907	.1	.9	.6062	.1	.9	.6758	.1	.9
	QDG	.6222	.1	.2	.3328	.1	.1	.4258	.1	.1
					p@					
Ground truth	Model	MAP	α	β	5	α	β	10	α	β
binary	GQD	.7127	.1	.7	.8041	.1	.9	.7306	.1	.3
	GDQ	.7552	.1	.9	.8122	.1	.6	.7408	.2	.8
	DGQ	**.7772**	.1	.9	**.8571**	.1	.9	**.7918**	.1	.9
	QDG	.4882	.1	.4	.5592	.1	.1	.5265	.1	.2
Graded	GQD	.7403	.1	.3	.6367	.1	.1	.5224	.1	.2
	GDQ	.7673	.1	.4	.6776	.1	.3	.5510	.1	.4
	DGQ	**.8092**	.6	.8	**.7102**	.5	.6	**.5531**	.7	.9
	QDG	.2182	.1	.4	.1714	.1	.1	.1673	.1	.1

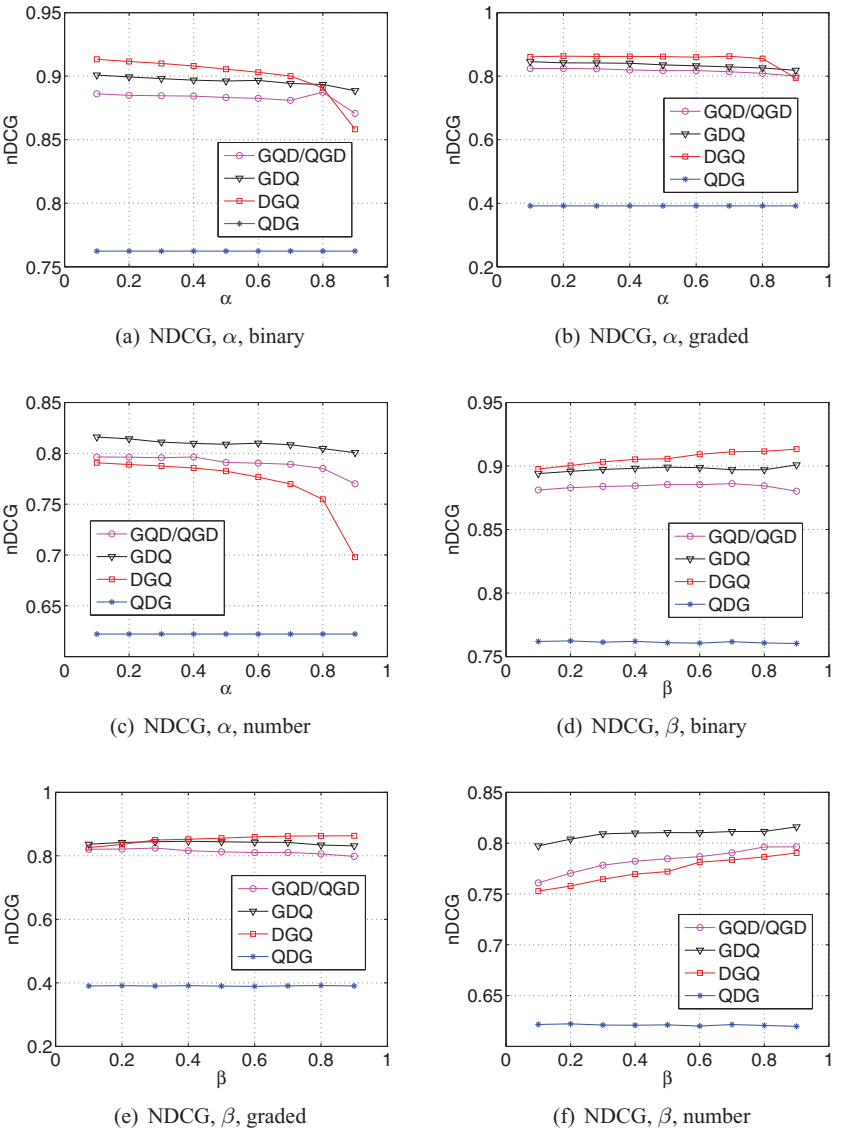
Table 2
Evaluation results using data from the TREC Enterprise 2006 track for models with 5-fold cross validation, using the binary, graded as well as number ground truths. For all the models, the evaluation results for MAP and p@5, 10 against the number ground truth are the same as those against the binary ground truth.

			NDCG@			p@	
Ground truth	Model	NDCG	5	10	MAP	5	10
Binary	GQD	.8826	.8078	.7743	.7045	.7918	.7184
	GDQ	.8940	.8123	.7912	.7307	.7878	**.7367**
	DGQ	**.8949**	**.8479**	**.8177**	**.7383**	**.8122**	.7327
	QDG	.7613	.5579	.5513	.4521	.5133	.4974
Graded	GQD	.7981	.6799	.7216	.7046	.6408	.5245
	GDQ	.8365	.7412	.7836	.7597	.6816	**.5755**
	DGQ	**.8594**	**.7805**	**.8056**	**.8065**	**.7020**	.5429
	QDG	.3824	.0936	.1176	.2047	.1635	.1547
Number	GQD	.7903	.6115	.6730			
	GDQ	**.8045**	**.6275**	**.6823**			
	DGQ	.7804	.5791	.6511			
	QDG	.6054	.3135	.4057			

performance on the metrics with optimal values of the smoothing parameter β. We then repeat this with the roles of α and β swapped. The results are shown in Figs. 6 and 8. The optimal values of β when changing α and the optimal values of α when changing β are listed in the captions in the figures.

A quick scan of Figs. 6 and 8 shows that QDG performs almost the same no matter how α and β change; it is always the worst performing model against every type of ground truth. No model is very sensitive to changes to the α and β parameters, no matter which type of ground truth and which metric (NDCG or MAP) we use. Fig. 6(a), 6(c) and 8(a) show that with α changing from 0.1 to 0.9, the performance of GQD and of GDQ decreases slightly. The performance of all of the models seems to level off in Figs. 6(b) and 8(b), and this is also true in Figs.6(e) and 8(d), which demonstrates that our models are not very sensitive to the changes of the two smoothing parameters in terms of NDCG and MAP against the graded ground truth. There are slight increases in performance in terms of NDCG and MAP against the binary and number ground truth in GQD, GDQ and DGQ models in Figs. 6(d), 6(f) and 8(c). To show the trend of the performance of our proposed models with two smoothing parameter more clearly, we also plot 3-D figures for the DGQ model; see Fig. 7. Other models have similar figures.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

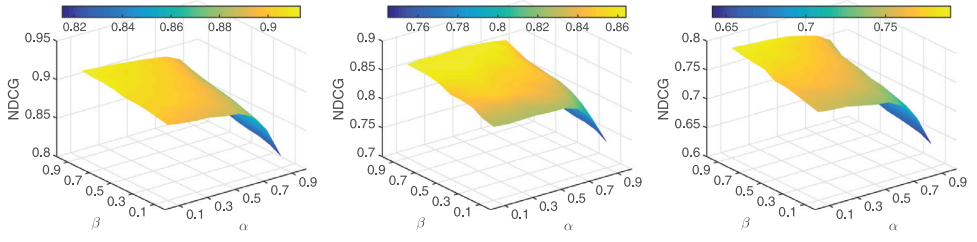


(a) NDCG, α, binary (b) NDCG, α, graded

(c) NDCG, α, number (d) NDCG, β, binary

(e) NDCG, β, graded (f) NDCG, β, number

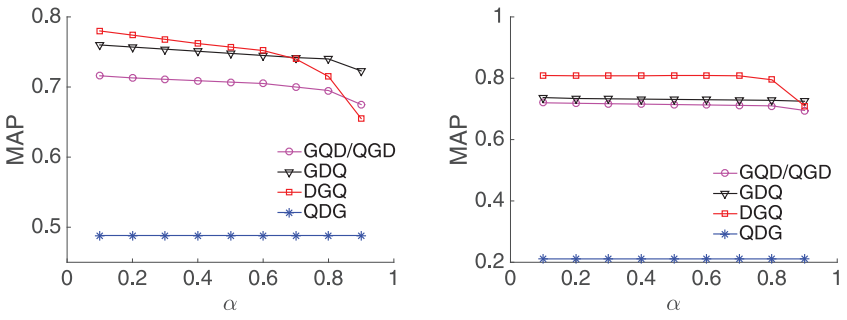
Fig. 6. NDCG performance of five models (GQD is the same as QGD), using two smoothing parameters against three types of ground truth: from Fig. 6(a)–6(c): smoothing parameter α changes from 0.1 to 0.9 with 0.1 step, and report metrics with optimal models. From Fig. 6(d)–(f): smoothing parameter β changes from 0.1 to 0.9 with 0.1 step, and report metrics with optimal models. Fig. 6(a) and (d) use NDCG against the binary ground truth, Fig. 6(b) and (e) use NDCG against the graded ground truth, Fig. 6(c) and (f) use NDCG against the number ground truth. For GQD/QGD, GDQ, DGQ and QDG models, the optimal β in Fig. 6(a) are 0.2, 0.9, 0.9, 0.2, respectively; optimal β in Fig. 6(c) are 0.3, 0.4, 0.9, 0.8, respectively; the optimal β in Fig. 6(b) are 0.9, 0.9, 0.9, 0.2, respectively. For these models, all optimal α in Fig. 6(d) are 0.1; the optimal α in Fig. 6(f) are 0.2, 0.1, 0.2, 0.1, respectively; and all optimal α in Fig. 6(e) are 0.1. (Best viewed in color).



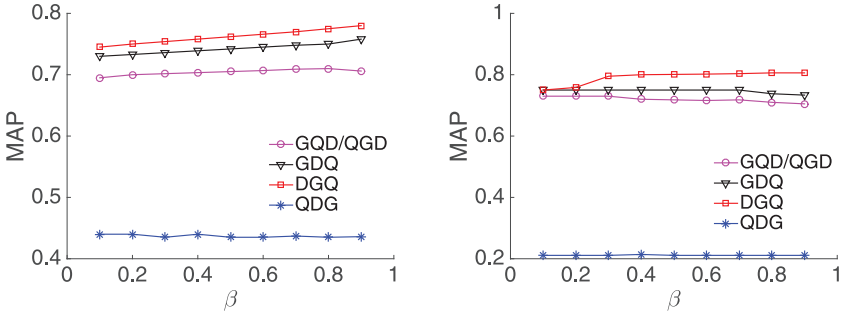
(a) DGQ, binary (b) DGQ, graded (c) DGQ, number

Fig. 7. NDCG performance of DGQ model in 3-D graphs. Fig. 7(a) is for DGQ model against the binary ground truth; Fig. 7(b) is for DGQ model against the graded ground truth; and Fig. 7(c) is for DGQ model against the number ground truth. (Best viewed in color).

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

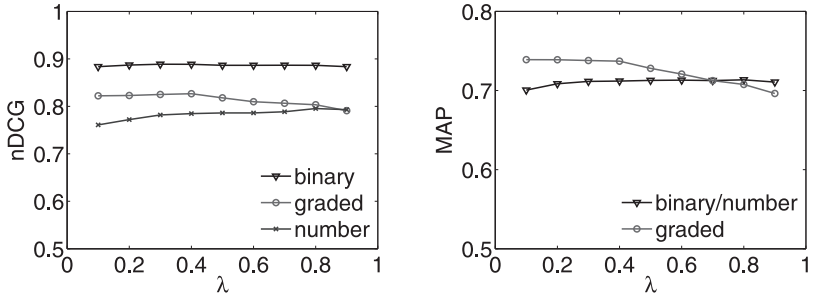


(a) MAP, α, binary (b) MAP, α, graded



(c) MAP, β, binary (d) MAP, β, graded

Fig. 8. NDCG and MAP of five models (GQD is the same as QGD), using two smoothing parameters against three types of ground truth: Fig. 8(a) and (b): smoothing parameter α changes from 0.1 to 0.9 with 0.1 step, and report metrics with optimal models. Fig. 8(c) and (d): smoothing parameter β changes from 0.1 to 0.9 with 0.1 step, and report metrics with optimal models. Fig. 8(a) and (c) use MAP against the binary/number ground truth, while Fig. 8(b) and (d) use MAP against the graded ground truth. For GQD/QGD, GDQ, DGQ and QDG models, the optimal β in Fig. 8(a) are 0.7, 0.9, 0.9 and 0.4, respectively; the optimal β in Fig. 8(b) are 0.3, 0.4, 0.8 and 0.4, respectively; all the optimal α in Fig. 8(c) are 0.1; and the optimal α in Fig. 8(d) are 0.1, 0.1, 0.6, 0.1, respectively. (Best viewed in color).



(a) NDCG performance (b) MAP performance

Fig. 9. NDCG and MAP scores of GQD, using one smoothing parameter against three types of ground truth. (Left): NDCG performance with changes in the smoothing parameter against the ground truths. (Right): MAP performance with changes in the smoothing parameter against the ground truths.

6.3. Smoothing with one parameter

If we smooth the GQD model (QGD model) with only one parameter (see (13)), how does it behave? Can it be better than smoothing using two parameters (see (9))? We change the smoothing parameter λ in (13) from 0.1 to 0.9 and observe its performance using NDCG and MAP against the binary, graded and number ground truths. It is clear from Fig. 9(a) that the performance of NDCG against the binary ground truth seems to level off with changes of λ (almost the same as when using two smoothing parameters, see Table 1), which means that the model is not sensitive to the smoothing parameter when using the binary ground truth. Fig. 9(a) also shows that the performance in terms of NDCG against the graded ground truth reaches its peak at $\lambda = 0.4$ with 0.8287 (almost the same as when using two smoothing parameters) and then decreases gradually to 0.7932 at $\lambda = 0.9$. In contrast, the performance in terms of NDCG against the number ground truth increases slowly and peaks at $\lambda = 0.8$ with 0.7956 (again, almost the same as when using two smoothing parameters) and

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

Table 3
Results of using a subset of top ranking documents.

Model	$n =$	NDCG	NDCG@5	MAP	p@5	p@10
GQD	$\|\mathcal{D}\|$	**.8826**	**.8025**	**.7003**	**.7755**	**.7204**
	10000	.8070	.6210	.6062	.6367	.5980
	5000	.8027	.6057	.5989	.6204	.6082
	1000	.7875	.5801	.5726	.5959	.5755
	500	.7674	.5666	.5496	.6122	.5571
	100	7870	.6068	.5575	.6122	.5741
GDQ	$\|\mathcal{D}\|$	**.8949**	**.8195**	**.7365**	**.8000**	**.7286**
	10000	.8408	.7050	.6563	.7143	.6388
	5000	.8354	.6710	.6464	.6653	.6408
	1000	.8076	.6123	.6068	.6163	.6143
	500	.7803	.5793	.5696	.6041	.5776
	100	.7894	.6090	.5668	.6163	.5857
DGQ	$\|\mathcal{D}\|$	**.8949**	**.8195**	**.7365**	**.8000**	**.7286**
	10000	.8488	.7146	.6727	.7143	.6714
	5000	.8435	.6965	.6670	.6980	.6673
	1000	.8225	.6503	.6263	.6449	.6469
	500	.8040	.6301	.5947	.6245	.6082
	100	.8066	.6440	.5889	.6367	.6061
QDG	$\|\mathcal{D}\|$	.7609	.5604	.4848	.5592	.5041
	10000	.7609	.5604	.4848	.5592	.5041
	5000	.7609	.5604	.4848	.5592	.5041
	1000	.7609	.5604	.4848	.5592	.5041
	500	.7609	.5604	.4848	.5592	.5041
	100	.7609	.5604	.4848	.5592	.5041

then decreases to 0.7931 at $\lambda = 0.9$. In addition, Fig. 9(b) shows that there is a downward trend in the performance of MAP against the graded ground truth, with a little decrease from 0.7431 (almost the same as when using two smoothing parameters) at $\lambda = 0.1$ to 0.7007 at $\lambda = 0.9$, while there is an upwards trend in the performance of MAP against the binary and number ground truth, with a minor increase from 0.7005 at $\lambda = 0.1$ to 0.7136 (almost the same as when using two smoothing parameters) at $\lambda = 0.8$.

6.4. Query-level analysis

Our aim here is to find out whether some queries are harder than others for the same model, using different metrics. We turn to a query-level analysis of the MAP performance for each model against the binary, number, and graded ground truth. We plot the differences in performance (per query) between the average AP score and the AP score per topic, sorted by performance difference. Fig. 10(a), (c), (e) and (g) show the plots for GQD, GDQ, DGQ and QDG when using the binary/number ground truth, respectively, whilst Fig. 10(b), (d), (f) and (h) show the plots for GQD, GDQ, DGQ and QDG when using the graded ground truth, respectively. From Fig. 10(a), (c), (e) and (g), it is clear that the performance in MAP does not differ dramatically in all of the models when against the binary or number ground truths. In comparison, for some queries the performance in terms of MAP is dramatically worse than the mean for GQD, GDQ and DGQ when using the graded ground truth. Fig. 10(h) shows that the performance of QDG is stable across queries in terms of AP.

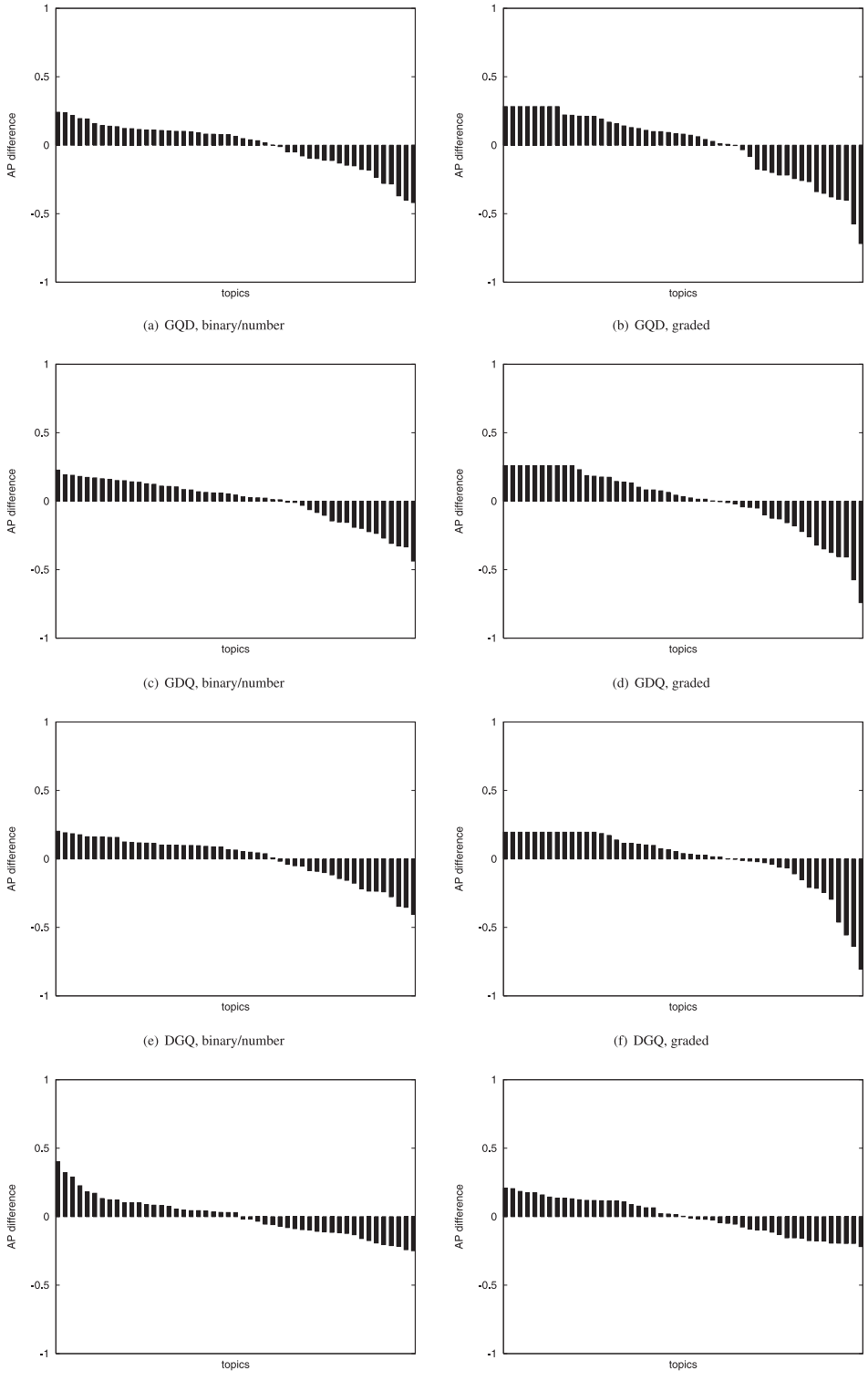
6.5. Topicality

Next we consider how the topicality of documents used to build the representations influences the performance on the knowledgeable group finding task. To answer this question, we use the full collection of the documents, as well as a subset of documents defined by taking the top n documents returned by a standard document retrieval run in response to the query. Table 3 shows the performance against the binary ground truth, achieved by using different values for n, the document cut-off. (We omit results against the other ground truths, as they are qualitatively similar to the results against the binary ground truth). In the table, $|\mathcal{D}|$ indicates that the models are built on the full document set. According to Table 3, it is clear that there is no improvement in performance when the size of the subset documents is reduced. For GQD, GDQ and DGQ, generally the use of a restricted subset of documents is not beneficial. Interestingly, the performance at $n = 100$ is slightly better than at $n = 500$. The absolute performance gains in moving from $n = 100$ to $n = 5000$ tend to be smaller than those obtained in moving from $n = 5000$ to $n = |\mathcal{D}|$.

6.6. Statistical significance and effect size

The aim of our final set of experiments is to determine whether the observed differences between our group finding approaches with two smoothing parameters strategy are statistically significant and what the effect sizes of the compared

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005



(a) GQD, binary/number (b) GQD, graded

(c) GDQ, binary/number (d) GDQ, graded

(e) DGQ, binary/number (f) DGQ, graded

(g) QDG, binary/number (h) QDG, graded

Fig. 10. Per query differences from the mean scores for different group finding models against the binary/number and graded ground truths.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

Table 4
p values of the two-tailed paired t-test between different models on NDCG and MAP metrics. Italics indicates that the two models are not statistically significantly different.

Metric	Ground truth	GQD vs. GDQ	GQD vs. DGQ	GQD vs. QDG	GDQ vs. DGQ	GDQ vs. QDG	DGQ vs. QDG
NDCG	Binary	.0107	.0005	.0000	.0496	.0000	.0000
	Graded	*.2349*	*.0616*	.0000	*.2282*	.0000	.0000
	Number	.0086	*.4107*	.0000	*.0570*	.0000	.0000
MAP	Binary/number	.0013	.0000	.0000	.0198	.0000	.0000
	Graded	*.2647*	.0140	.0000	.0221	.0000	.0000

Table 5
Effect size between different models on NDCG and MAP metrics against the binary, graded and number ground truths.

Metric	Ground truth	GQD vs. GDQ	GQD vs. DGQ	GQD vs. QDG	GDQ vs. DGQ	GDQ vs. QDG	DGQ vs. QDG
NDCG	Binary	0.119	0.121	0.699	0.009	1.111	0.802
	Graded	0.193	0.321	2.716	0.121	3.007	3.353
	Number	0.733	0.098	1.956	0.213	1.099	0.915
MAP	Binary/number	0.162	0.201	0.957	0.044	0.845	1.108
	Graded	0.234	0.451	2.706	0.214	3.145	3.663

models are. We use a two-tailed paired t-test between two different models on NDCG and MAP data and test for statistically significant differences at the 0.95 confidence level. Table 4 shows the result with *p* value of the two-tailed paired t-test between different models on NDCG and MAP metrics. Table 4 indicates that when using NDCG as a performance metric the differences between the GQD and GDQ models against the graded ground truth are not statistically significant. This is also true for the differences between GQD and DGQ, and between GDQ and DGQ against the graded and number ground truths. When using MAP as the metric, the differences between all models are statistically significant except for those between GQD and GDQ. We also test the differences between the optimal GQD model with smoothing with two parameters and the GQD model with smoothing with a single smoothing parameter based on NDCG and MAP against three ground truths, and find that there are no statistically significant differences at the 0.95 confidence level except based on NDCG against binary ground truth where the p value is 0.0270. This demonstrates that GQD with smoothing with two parameters and GQD with smoothing with a single parameter are almost the same models in our experiments.

Finally, we report on the effect sizes of the comparisons among different models to see whether the differences are really obvious. We use Cohen's *d* (Cohen, 1988) to compute the effect sizes. As can be seen in Table 5, the effect sizes between many different models are quite large, especially for those compared with QDG model. For some comparisons, e.g., the GQD vs. GDQ models against the graded ground truth, the effect sizes are small and the corresponding *p* values are large (i.e., the differences are not statistically significant).

In sum, our group finding models manage to achieve high absolute scores. Also, some of our models perform similarly. Specifically, GQD and GDQ models are not statistically significantly different when using NDCG as a performance metric and against the binary and graded ground truths. This is also true for the differences between GQD and DGQ, and between GDQ and DGQ against the graded and number truths. But when using MAP as our metric, the differences between all models are statistically significant except for those between the GQD and GDQ models. The differences between the optimal GQD with smoothing with two parameters and GQD with a single smoothing parameter against all metrics are not statistically significant. In addition, our query-level analysis shows that the performance of all proposed models does not differ dramatically when using the binary or number ground truths. In terms of the topicality of documents used to build the representations influencing the performance on our group finding task, for the GQD, GDQ and DGQ models the use of a restricted subset of documents is not obviously beneficial for the performance.

7. Conclusions

We have introduced a new group finding task: given a query, find knowledgeable groups that have expertise on the topic of the query. We have proposed to model the task in three ways, which has given rise to five distinct models, GQD, GDQ, DGQ, QDG and QGD. We have also constructed an experimental collection by using the TREC 2005 and 2006 Enterprise collections. We have introduced three kinds of ground truth and explored and evaluated our models along many dimensions.

We have conducted a large number of experiments and found that directly collecting expertise evidence from the documents is the most effective way to find knowledgeable groups when using the binary or graded ground truth, and aggregating the expertise of each experts in the same group can also a good way to find the groups. QDG appears to be the worst performing model. We have also found that only few of the models are sensitive to changes of parameters when using a

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

two parameter smoothing strategy. There is an overall trend that the more documents are used, the better the performance will be. But using a small subset of documents can also yield quite good knowledgeable group finding performance. We have found statistically significant differences between the models when using MAP against multiple types of ground truth in most cases.

Our five knowledgeable group finding language models may be interesting for those working on entity retrieval, e.g., expert finding, rank aggregation, and language modeling, as our group finding models contain these three core ingredients. As to future work, there are several possibilities to extend this research. Experts' profiles change from time to time (Rybak, Balog, & Nørvåg, 2014) and new expertise may emerge in individuals or their groups (van Dijk, Tsagkias, & de Rijke, 2015). Identifying the skills and knowledge of a group and tracking how they emerge and change over time is an important next research direction for group finding. Beside using full names of the experts, considering other information to capture the associations between documents and experts, such as the topics they can be linked with and share (Meij, Bron, Hollink, Huurnink, & de Rijke, 2011), is also an interesting research topic in the next step. We have so far tackled our knowledgeable group finding task by using unsupervised methods that focused on directly inferring information from heterogeneous documents. In the future, we plan to adopt learning to rank approaches for the group finding task.

Acknowledgments

We thank our anonymous reviewers for their helpful comments, which helped to improve the paper.

This research was partially supported by the China Scholarship Council, the UCL Big Data Institute, Amsterdam Data Science, the Dutch national program COMMIT, Elsevier, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nr 312827 (VOX-Pol), the ESF Research Network Program ELIAS, the HPC Fund, the Royal Dutch Academy of Sciences (KNAW) under the Elite Network Shifts project, the Microsoft Research Ph.D. program, the Netherlands eScience Center under project number 027.012.105, the Netherlands Institute for Sound and Vision, the Netherlands Organisation for Scientific Research (NWO) under project nrs 727.011.005, 612.001.116, HOR-11-10, 640.006.013, 612.066.930, CI-14-25, SH-322-15, the Yahoo Faculty Research and Engagement Program, and Yandex. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

Balog, K., Azzopardi, L., & de Rijke, M. (2006). Formal models for expert finding in enterprise corpora. In *Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '06* (pp. 43–50).

Balog, K., Azzopardi, L., & de Rijke, M. (2009). A language modeling framework for expert finding. *Information Processing & Management, 45*(1), 1–19.

Balog, K., Bogers, T., Azzopardi, L., de Rijke, M., & van den Bosch, A. (2007). Broad expertise retrieval in sparse data environments. In *Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 551–558).

Balog, K., Bron, M., & de Rijke, M. (2011). Query modeling for entity search based on terms, categories, and examples. *ACM Transaction on Information Systems, 29*(4), 1–31.

Balog, K., Fang, Y., de Rijke, M., Serdyukov, P., & Si, L. (2012). Expertise retrieval. *Foundations and Trends in Information Retrieval, 6*(2–3), 127–256.

Balog, K., & de Rijke, M. (2008). Associating people and documents. In *Ecir'08* (pp. 296–308).

Beliakov, G., James, S., & Li, G. (2011). Learning choquet-integral-based metrics for semisupervised clustering. *IEEE Transaction on Fuzzy Systems, 19*(3), 562–574.

Campbell, C. S., Maglio, P. P., Cozzi, A., & Dom, B. (2003). Expertise identification using email communications. In *Proceedings of the twelfth international conference on information and knowledge management, CIKM '03* (pp. 528–531). New York, NY, USA: ACM.

Chen, L., Zeng, W., & Yuan, Q. (2013). A unified framework for recommending items, groups and friends in social media environment via mutual resource fusion. *Expert Systems with Applications, 40*(8), 2889–2903.

Cimiano, P., Schultz, A., Sizov, S., Sorg, P., & Staab, S. (2009). Explicit versus latent concept models for cross-language information retrieval. In *Proceedings of the 21st international joint conference on artifical intelligence, IJCAI'09* (pp. 1513–1518).

Clarke, C. L., Kolla, M., Cormack, G. V., Vechtomova, O., Ashkan, A., Büttcher, S., & MacKinnon, I. (2008). Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '08* (pp. 659–666). ACM.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (second ed.). Academic press.

Craswell, N., de Vries, A. P., & Soboroff, I. (2005). Overview of the TREC 2005 enterprise track. In *Trec'05* (pp. 1–7).

van Dijk, D., Tsagkias, M., & de Rijke, M. (2015). Early detection of topical expertise in community question answering. In *Proceedings of SIGIR 2015: 38th international ACM SIGIR conference on research and development in information retrieval*. ACM.

Fang, Y., Si, L., & Mathur, A. P. (2010). Discriminative models of integrating document evidence and document-candidate associations for expert search. In *Proceedings of the 33rd annual international ACM SIGIR conference on research and development in information retrieval. SIGIR '10* (pp. 683–690).

Garcia, I., & Sebastia, L. (2014). A negotiation framework for heterogeneous group recommendation. *Expert Systems with Applications, 41*(4, Part 1), 1245–1261.

Gerani, S., Carman, M. J., & Crestani, F. (2010). Proximity-based opinion retrieval. In *Proceedings of the 33nd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 403–410).

Ghosh, S., Kothari, M., Halder, A., & Ghosh, A. (2009). Use of aggregation pheromone density for image segmentation. *Pattern Recognition Letters, 30*, 939–949.

Jelinek, F., & Mercer, R. (1980). Interpolated estimation of Markov sourceparameters from sparse data. *Pattern Recognition in Practice*, 381–402.

Juang, M.-C., Huang, C.-C., & Huang, J.-L. (2013). Efficient algorithms for team formation with a leader in social networks. *Journal of Supercomputing, 66*(2), 721–737.

Kargar, M., & An, A. (2011). Discovering top-k teams of experts with/without a leader in social networks. In *Proceedings of the 20th acm international conference on information and knowledge management. CIKM '11* (pp. 985–994).

Ko, J., Si, L., Nyberg, E., & Mitamura, T. (2010). Probabilistic models for answer-ranking in multilingual question-answering. *ACM Transactions on Information Systems, 28*(3), 16:1–16:37. doi:10.1145/1777432.1777439.

Lappas, T., Liu, K., & Terzi, E. (2009). Finding a team of experts in social networks. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining. KDD '09* (pp. 467–476).

Li, C.-T., Shan, M.-K., & Lin, S.-D. (2013). On team formation with expertise query in collaborative social networks. *Knowledge and Information Systems*, 1–23.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005

Liang, S., & de Rijke, M. (2013). Finding knowledgeable groups in enterprise corpora. In *Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval. SIGIR '13* (pp. 1005–1008).
Lv, Y., & Zhai, C. (2009). Positional language models for information retrieval. In *Proceedings of the 32nd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 299–306).
Macdonald, C., & Ounis, I. (2011). Learning models for ranking aggregates. In *Ecir'11, advances in information retrieval. Lecture Notes in Computer Science: 6611* (pp. 517–529). Springer Berlin Heidelberg.
Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. New York, NY, USA: Cambridge University Press.
Meij, E., Bron, M., Hollink, L., Huurnink, B., & de Rijke, M. (2011). Mapping queries to the linking open data cloud: a case study using dbpedia. *Journal of Web Semantics, 9*(4), 418–433.
Mockus, A., & Herbsleb, J. D. (2002). Expertise browser: a quantitative approach to identifying expertise. In *Proceedings of the 24th international conference on software engineering. ICSE '02* (pp. 503–512). New York, NY, USA: ACM.
Moreira, C., & Wichert, A. (2013). Finding academic experts on a multisensor approach using Shannon's entropy. *Expert Systems with Applications, 40*(14), 5740–5754.
Neshati, M., Beigy, H., & Hiemstra, D. (2014). Expert group formation using facility location analysis. *Information Processing & Management, 50*(2), 361–383.
Ponte, J. M., & Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval. SIGIR '98* (pp. 275–281).
Pryor, G. A., Myles, J. W., Williams, D. R. R., & Anand, J. K. (1988). Team management of the elderly patient with hip fracture. *The Lancet*, 401–403.
Rybak, J., Balog, K., & Nørvåg, K. (2014). Temporal expertise profiling. In *Proceedings of the 36th European conference on advances in information retrieval. ECIR '14* (pp. 540–546).
Senge, R., & Hullermeier, E. (2011). Top-down induction of fuzzy pattern trees. *IEEE Transaction on Fuzzy Systems, 19*(2), 241–252.
Soboroff, I., de Vries, A. P., & Craswell, N. (2006). Overview of the TREC 2006 enterprise track. In *Trec'06* (pp. 1–20).
Sozio, M., & Gionis, A. (2010). The community-search problem and how to plan a successful cocktail party. In *Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '10* (pp. 939–948).
Sun, K., Wang, X., Sun, C., & Lin, L. (2011). A language model approach for tag recommendation. *Expert Systems with Applications, 38*(3), 1575–1582.
Tsagkias, M., de Rijke, M., & Weerkamp, W. (2011). Hypergeometric language models for republished article finding. In *Proceedings of the 34th annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '11* (pp. 485–494).
Tung, Y.-H., Tseng, S.-S., Weng, J.-F., Lee, T.-P., Liao, A. Y., & Tsai, W.-N. (2010). A rule-based CBR approach for expert finding and problem diagnosis. *Expert Systems with Applications, 37*(3), 2427–2438.
Wang, K., Li, X., & Gao, J. (2010). Multi-style language model for web scale information retrieval. In *Proceedings of the 33rd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 467–474).
Weerkamp, W., Berendsen, R., Kovachev, B., Meij, E., Balog, K., & de Rijke, M. (2011). People searching for people: analysis of a people search engine log. In *Proceedings of the 34th annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '11* (pp. 45–54).
Yang, D.-N., Chen, Y.-L., Lee, W.-C., & Chen, M.-S. (2011). On social-temporal group query with acquaintance constraint. *Proceedings of the VLDB Endowment, 4*(6), 397–408.
Zhai, C., & Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transaction on Information Systems, 22*(2), 179–214.
Zhao, J., & Yun, Y. (2009). A proximity language model for information retrieval. In *Proceedings of the 32nd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 291–298).
Zhou, S.-M., Chiclana, F., John, R. I., & Garibaldi, J. M. (2011). Alpha-level aggregation: a practical approach to type-1 OWA operation for aggregating uncertain information with applications to breast cancer treatments. *IEEE Transaction on Knowledge and Data Engineering, 23*(10), 1455–1468.

Please cite this article as: S. Liang, M. de Rijke, Formal language models for finding groups of experts, Information Processing and Management (2016), http://dx.doi.org/10.1016/j.ipm.2015.11.005