Finding Knowledgeable Groups in Enterprise Corpora

Liang, S.; de Rijke, M.

Published in:
SIGIR '13: the proceedings of the 36th International ACM SIGIR Conference on Research & Development in Information Retrieval: July 28-August 1, 2013, Dublin, Ireland

DOI:
10.1145/2484028.2484109

Citation for published version (APA):
ABSTRACT
The task of finding groups is a natural extension of search tasks aimed at retrieving individual entities. We introduce a group finding task: given a query topic, find knowledgeable groups that have expertise on that topic. We present four general strategies to this task. 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 model directly estimates the degree to which a group is a knowledgeable group for a given topic. We construct a test collections based on the TREC 2005 and 2006 Enterprise collections. We find significant differences between different ways of estimating the association between a topic and a group. Experiments show that our knowledgeable group finding models achieve high absolute scores.

Categories and Subject Descriptors
H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing

Keywords
Group finding, expertise retrieval, language modeling

1. INTRODUCTION
A major challenge within any organization is managing the expertise within the organization such that groups with expertise in a particular area can be identified [2]. 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 [6].

Traditional approaches to finding knowledge, whether in individuals or in groups within an organization, often include two main steps. For a given task the expertise of the experts in each 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. In addition, this approach is usually restricted to a fixed set of expertise areas [7]. 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 [2]. To compute the expertise values of a group, in principle, many aggregation operators are available, e.g., sum or average. These can be employed to combine individual experts’ expertise values. There are at least 90 families of aggregation operators [11], which have been put to use in a range of applications. But the problem of how to aggregate expertise values of experts within a group so that the expertise scores of different groups can be easily compared and ranked is unknown.

We treat the problem of finding a knowledgeable group differently. Four distinct models are proposed. Our models are based on probabilistic language modeling techniques. Each model ranks groups according to the probability of the group being a knowledgeable group for a query topic, but the models differ in how this is performed. Three types of variable 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 on 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.

2. RELATED WORK
Significant research effort has been invested in locating a group of individuals in an organization. Yang et al. [10] 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. Sozio and Gionis [9] study a query-dependent variant of the community-detection problem: given a graph, and a set of nodes in the graph as their input query, find a subgraph that contains the input query nodes and is densely connected. Lappas et al. [6] study the problem of given a task, a pool of individuals χ with different skills and a social network that captures the compatibility among them, finding a subset of χ, who together have the skills to complete a task with minimal communication costs. Kargar and An [5] design communication cost functions for two types of communication structures.

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 is to answer questions such as “Which groups are knowledgeable groups on topic T?” whilst the second is to answer the question “What does group G know?” We focus on the first sub-problem.
3. MODELING GROUP FINDING

In our modeling of the knowledgeable group finding task, groups, documents, and queries are considered in different orders. Groups are ranked according to how likely they have expertise on the given query according to the estimated language model.

**Problem definition and context.** We address the following problem: what is the probability of a group $g$ being a knowledgeable group given query topic $q$? We have to estimate the probability of a group $g$ given a query $q$ and then rank groups according to this probability. 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 $p(g|q) = p(q|g)p(g)p(q)^{-1}$, where $p(q)$ is the probability of a query and $p(g)$ the probability of a group, both of which can be assumed to be uniform for a query and a group, considering that $q$ is the same during retrieval and there is no group that is more likely to be relevant. Hence, ranking groups according $p(g|q)$ boils down to ranking a query topic given a group: $p(g|q)$. To determine $p(g|q)$ or $p(q|g)$ we consider experts, groups, documents and queries in different orders, so as to arrive at four distinct models.

**Four group finding models.** The first of four models for group finding is presented in some detail; because of lack of space, the others are presented much more concisely. We start with 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 means that the evidence of whether a group is knowledgeable on the topic is collected via the experts in the group ($G$), then how likely each expert in the group has expertise on each subtopics in the query ($Q$) topic is computed via the documents ($D$). GDQ denotes that the evidence of whether a group is knowledgeable 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. We assume that experts in the same group $g$ are conditionally independent given the group, such that:

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

where $ex$ is an expert belonging to group $g$, $p(ex|g)$ is the probability of how likely an expert $ex$ belonging to a group $g$, 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 $p(ex|g) = p(q|ex)p(ex)p(q)^{-1}$, 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 we assume that each expert is equally important, $p(ex)$ is assumed to be constant. Additionally, for each query topic, $p(q)$ is the same, hence, $p(ex|g)$ is proportional to $p(ex|q)$. So, $p(g|q)$ becomes

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

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

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

where $p(t|d)$ is the probability of a term given a document and $n(t,q)$ is the number of occurrences of term $t$ in query $q$. Combined, we can rewrite $p(q|ex)$ as follows.

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

To obtain $p(t|d)$, we take the sum over documents $d$ in the collection. This can be expressed as $p(t|d) = \sum_{d} p(t|d)p(d|ex)$, where $p(t|d)$ is the probability of term $t$ given document $d$, and $p(d|ex)$ is the probability of $d$ given expert $ex$. Now we can obtain the probability of a group given a query, i.e., our GQD model:

$$p(q|g) \propto \prod_{ex \in g} \left\{ \prod_{d \in D} p(t|d)p(d|ex) \right\}^{as(ex,g)}.$$

**The GDQ Model.** We can compute the probability of a query $q$ given an expert $ex$ in a different way. 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 of document $d$ given query $q$, respectively. Based on this, we obtain our second aggregation model, i.e., our GDQ model:

$$p(q|g) \propto \prod_{ex \in g} \left\{ \sum_{d} \left\{ \prod_{t \in q} p(t|d)^{n(t,q)} \right\} p(d|ex) \right\}^{as(ex,g)}.$$

**The DGQ model.** Next we consider a document model. Instead of aggregating expertise scores of all the experts within a group as in our aggregation models, as the key terms DGQ in this model’s name suggests, the probability $p(q|g)$ can be computed directly via the documents ($D$). For each we compute how likely the group ($G$) is associated with the query topic, and how likely it is talking about the given query ($Q$) topic, such that: $p(q|g) = \sum_{d} p(g|d)p(d|q)$, 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. This, then, is how $p(q|g)$ can be represented, i.e., our DGQ model:

$$p(q|g) \propto \sum_{d} \left\{ \prod_{t \in q} p(t|d)^{n(t,q)} \right\} \left\{ \prod_{t \in q} p(t|d) \right\}^{as(ex,g)}.$$
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 expert \( ex \) and group \( g \) to which the expert belongs. We define the following group expert association \( as(ex, g) = \frac{1}{|g|} \), where \(|g|\) is the total number of experts within the group to which they belong.

Smoothing strategies. In our four models, the term \( p(g|q) \) may contain zero probabilities due to data sparsity. E.g., 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. Hence, we have to infer a group model \( \theta_g \), such that the probability of a group given a query model is \( p(\theta_g|q) \). We employ Jelinek-Mercer smoothing [4] to estimate \( p(\theta_g|q) \): we consider two types.

To facilitate comparisons and for the sake of uniformity, instead of estimating \( p(g|q) \) directly, we can easily infer a document model \( \theta_d \) such that the probability of term \( t \) given a document \( d \) model is \( p(t|\theta_d) \), and infer an expert model \( \theta_{ex} \) such that the probability of a document \( d \) given an expert \( ex \) is \( p(d|\theta_{ex}) \). The document model, then, is a linear interpolation of the background model \( p(t) \) and the smoothed estimate: \( p(t|\theta_d) = (1 - \alpha)p(t|\theta_d) + \alpha p(t) \), where \( \alpha \) is a smoothing parameter (\( 0 < \alpha < 1 \)). The expert model is a linear interpolation of the background model \( p(d) \) and the smoothed estimate: \( p(d|\theta_{ex}) = (1 - \beta)p(d|\theta_{ex}) + \beta p(d) \), where \( \beta \) is a smoothing parameter (\( 0 < \beta < 1 \)). Let \( \theta(t, a, d) \) be short for \( p(t|\theta_d) \) (\( 1 - \alpha \)) \( p(t) + \alpha p(t|\theta_d) \), and \( \vartheta(\beta, d, ex) \) be short for \( p(d|\theta_{ex}) \) (\( 1 - \beta \)) \( p(d) + \beta p(d|\theta_{ex}) \). Then, the group finding model GQD can be smoothed and estimated as

\[
p(g|q) \approx \prod_{ex \in g} \left\{ \prod_{t \in q} \theta(t, a, d) \cdot \vartheta(\beta, d, ex) \right\}^{n(t,q)}
\]

where \( as' \) abbreviates \( as(ex, g) \). The other group finding models, GDQ, DGQ, and QGD, can be smoothed and estimated in an analogous manner. As to the GDQ model:

\[
p(g|q) \approx \prod_{ex \in g} \left\{ \sum_d \left( \prod_{t \in q} \theta(t, a, d) \cdot \vartheta(\beta, d, ex) \right) \right\}^{as'(ex,g)}
\]

with \( as' \) as before. For the DGQ model we have

\[
p(g|q) \approx \sum_d \left( \prod_{t \in q} \theta(t, a, d) \cdot \vartheta(\beta, d, ex) \right)^{as(ex,g)} \prod_{t \in q} \theta(t, a, d)^{n(t,q)}
\]

and the QGD model can be smoothed and estimated as

\[
p(g|q) \approx \prod_{t \in q} \left\{ \sum_d \theta(t, a, d) \cdot \vartheta(\beta, d, ex) \right\}^{n(t,q)}
\]

For the GDQ model, we also consider a second type of smoothing strategy with one parameter: \( p(g|q) \approx \prod_{ex \in g} \prod_{t \in q} \left\{ (1 - \lambda) \sum_d p(t|d) p(d|ex) + \lambda p(t) \right\}^{n(t,q)} \).
Table 1: Evaluation results for all optimal models with two smoothing parameters, using the binary, graded and number ground truths. For each metric, we report the evaluation results, followed by the optimal smoothing parameters \( \alpha \) and \( \beta \).

<table>
<thead>
<tr>
<th>Ground truth</th>
<th>Model NDCG</th>
<th>MAP</th>
<th>NDCG@5</th>
<th>MAP@5</th>
</tr>
</thead>
<tbody>
<tr>
<td>binary</td>
<td>GDQ</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>.8661</td>
<td>.1</td>
<td>.2 .8165</td>
<td>.1 .7850</td>
</tr>
<tr>
<td></td>
<td>.1 .7127</td>
<td>.1 .3</td>
<td>.8041</td>
<td>.1 .9 .7306</td>
</tr>
<tr>
<td></td>
<td>.9009</td>
<td>.1 .9 .8291</td>
<td>.1 .3 .7936</td>
<td>.1 .2 .7552</td>
</tr>
<tr>
<td></td>
<td>.9133</td>
<td>.1 .9 .8680</td>
<td>.1 .9 .8420</td>
<td>.1 .9 .7772</td>
</tr>
<tr>
<td></td>
<td>.7623</td>
<td>.1 .9 .5064</td>
<td>.1 .1 .5568</td>
<td>.1 .2 .4882</td>
</tr>
<tr>
<td>graded</td>
<td>GDQ</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>.8245</td>
<td>.2 .3</td>
<td>.7237</td>
<td>.2 .3 .7595</td>
</tr>
<tr>
<td></td>
<td>.8457</td>
<td>.1 .4</td>
<td>.7675</td>
<td>.1 .3 .7945</td>
</tr>
<tr>
<td></td>
<td>.8631</td>
<td>.1 .2</td>
<td>.7991</td>
<td>.5 .8 .8160</td>
</tr>
<tr>
<td></td>
<td>.3916</td>
<td>.1 .8</td>
<td>.1012</td>
<td>.1 .1 .1282</td>
</tr>
<tr>
<td>number</td>
<td>GDQ</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>.7964</td>
<td>.1 .9</td>
<td>.6210</td>
<td>.1 .8 .6730</td>
</tr>
<tr>
<td></td>
<td>.8160</td>
<td>.1 .9</td>
<td>.6496</td>
<td>.1 .9 .6908</td>
</tr>
<tr>
<td></td>
<td>.7907</td>
<td>.1 .9</td>
<td>.6062</td>
<td>.1 .9 .6788</td>
</tr>
<tr>
<td></td>
<td>.6222</td>
<td>.1 .2</td>
<td>.3328</td>
<td>.1 .1 .4258</td>
</tr>
</tbody>
</table>

Figure 1: Topic-level differences from the mean scores for GQD using the binary/number and graded ground truths.

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

Table 2: Two-tailed paired t-test between different models on NDCG and MAP metrics.

<table>
<thead>
<tr>
<th>metric</th>
<th>ground truth</th>
<th>GDQ vs. vs.</th>
<th>GDQ vs. vs.</th>
<th>GDQ vs. vs.</th>
<th>GDQ vs. vs.</th>
<th>GDQ vs. vs.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NDCG</td>
<td>binary</td>
<td>graded</td>
<td>number</td>
<td>binary</td>
<td>graded</td>
</tr>
<tr>
<td></td>
<td>.0107</td>
<td>.0005</td>
<td>.0000</td>
<td>.0496</td>
<td>.0000</td>
<td>.0000</td>
</tr>
<tr>
<td></td>
<td>.2349</td>
<td>.0616</td>
<td>.0000</td>
<td>.2282</td>
<td>.0000</td>
<td>.0000</td>
</tr>
<tr>
<td></td>
<td>.0086</td>
<td>.4107</td>
<td>.0000</td>
<td>.0570</td>
<td>.0000</td>
<td>.0000</td>
</tr>
<tr>
<td>MAP</td>
<td>bn/numb</td>
<td>.0013</td>
<td>.0000</td>
<td>.0000</td>
<td>.0198</td>
<td>.0000</td>
</tr>
<tr>
<td></td>
<td>graded</td>
<td>.2647</td>
<td>.0140</td>
<td>.0000</td>
<td>.0231</td>
<td>.0000</td>
</tr>
</tbody>
</table>

7. CONCLUSIONS

We have introduced a group finding task. We proposed four models, GQD, GDQ, DGQ and QDG. We also constructed an experimental collection by using the TREC 2005 and 2006 Enterprise collections. We introduced three kinds of ground truth and evaluated our models along many dimensions. Directly collecting expertise evidence from documents is the most effective way to find knowledgeable groups when using the binary or graded ground truths, and aggregating the expertise of each expert in the same group can also be a good way to find the groups. Our models are not very sensitive to changes of the parameters when using a two parameter smoothing strategy. We found statistically significant differences between the models when using MAP scores based on multiple types of ground truth.

Acknowledgements. We thank our reviewers for their helpful comments. This research was supported by the European Community’s Seventh Framework Programme (FP7/2007-2013) under agreements 258191 (PROMISE) and 288024 (LiMoSINe), the Netherlands Organisation for Scientific Research (NWO) under nrs 640.004.802, 727.011.005, 612.001.116, HOR-11-10, the Center for Creation, Content and Technology (CCCT), the BILAND project funded by the CLARIN-nl program, the Dutch national program COMMIT, the ESF Research Network Program ELIAS, the Elite Network Shifts project funded by the Royal Dutch Academy of Sciences (KNAW), and the Netherlands eScience Center under number 027.012.105.

8. REFERENCES


[1008]