Applying Learning To Rank and Document Aggregation Techniques on a Translator-Expert Retrieval Framework

Mihai Lupu
Information and Software Engineering Group
Vienna University of Technology
Favoritenstrasse 9-11/188
A-1040 Vienna, Austria
Email: lupu@ifs.tuwien.ac.at

Navid Rekabsaz
Faculty of Informatics
Vienna University of Technology
Email: rekabsaz.n@gmail.com

always use university/company email on papers - never yahoo/gmail/hotmail/etc

***Abstract*—Expertise Retrieval, whose task is to suggest people with relevant expertise regarding to a query, has received increasing interest in recent years. In this paper, we propose an Expert Retrieval platform narrowed down to translators as experts. Issues and obstacles during design and development as well as acquired solutions and results are reported. Ranking the translators using Learning to Rank as well as documents' aggregation functions are two main issues which are specifically studied in the paper.**

I. Introduction

The goal of expertise retrieval is to link humans to expertise areas, and vice versa. In other words, the task of expertise retrieval is to identify a set of persons with relevant expertise for the given query [1] [5].

With the development of information retrieval (IR) techniques, many research efforts in this field have been made to address high-level IR and not just traditional document retrieval, such as entity retrieval and expertise retrieval [2]. The launch of the Expert Finding task at TREC has generated a lot of interest in expertise retrieval, with rapid progress being made in terms of modeling, algorithms, and evaluation aspects [3] [5].

do not use references as nouns. Instead of "are proposed in [3]" you should say "are proposed by Cao and colleagues [3]"

Two principal approaches are proposed in [3] based on probabilistic language modeling techniques. They were formalized as so-called candidate models and document models. The candidate-based approach, ~~which is~~ also referred to as profile-based method, builds a textual representation of candidate experts and then ranks them based on the query. The document models ~~is to~~ first find documents ~~that are~~ relevant to the topic and then locate the experts associated with these documents [1].

Ranking techniques are ~~mostly~~ one of the essential parts of IR frameworks. In recent years, Learning to Rank (L2R) has been studied extensively specially for document retrieval. It refers to machine learning techniques for training the model in a ranking task [1]. In essence, expert search is a ranking problem and thus the existing L2R techniques can be naturally applied to it [6].

As well as ranking techniques, aggregation functions have a solid effect on the performance of IR systems. Aggregate tasks are those where documents' similarities are not the final outcome, but instead an intermediary component. In expert search, a ranking of candidate persons with relevant expertise to a query is generated after aggregation of their related documents [8].

what does it mean to have a 'solid' effect?

This paper addresses the problem of searching translators as experts. We have applied Learning to Rank in a candidate-based approach. ~~Besides,~~ Additionally, different aggregation algorithms related to documents of translators ~~has~~ have been studied.

The remaining of the paper is organized as follows. In Section II, the Translator-Expert Retrieval framework is described in detail. Then, Section III explains methods used in the study. In Section IV, we report the result of applied methods on the framework. Finally, we conclude the study in Section V.

II. Case Study

The data flow of expert searching is depicted in Figure1. The client submits a document and searches for translators with a specific target language. Based on query document, the framework figures out offered price, delivery time, proficiency of translators and number of cooperation times related to each translator. The result is processing by a ranking system and the most related translators are offered to the client.

As it is shown in Figure1, the essential components of the framework are *Ranking*, *Proficiency Estimator*, *Scheduler* and *Profiler*.

Ranking system uses L2R to return the most related translators to the client. The training data is provided by a group of evaluators who are familiar with business of company, using an evaluation system. The evaluation suggests three translators and by comparing between their factors, the evaluators choose one. In order to prevent bias in evaluation, the translators are suggested randomly and

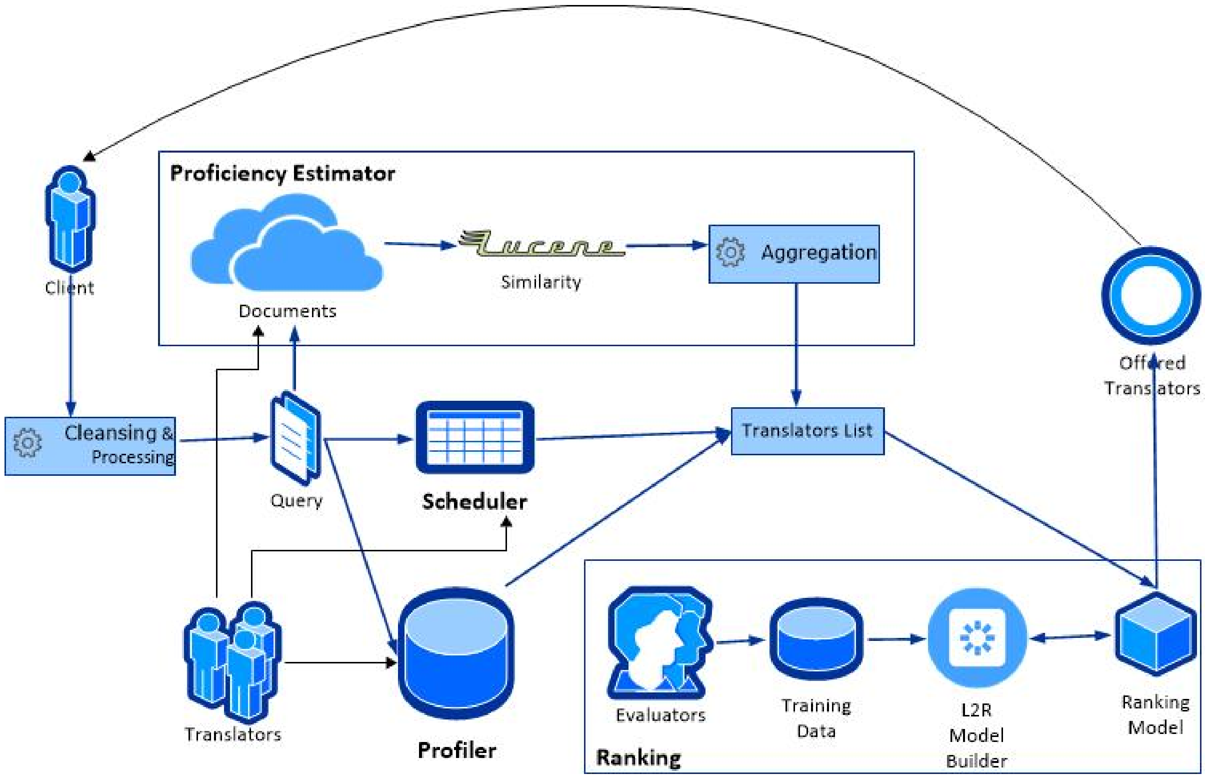

Figure 1. System Data Flow

without name and picture. Applied learning to rank methods and results are described in Section IV. *Proficiency Estimator* stores the previous-translated documents in the cloud and indexes them using Lucene library. The similarity between query and indexed documents is used as a base for estimation of translator's proficiency. In order to find a value proficiency, the similarity scores are aggregated. section 3? The applied aggregation function is described in Section IV. *Scheduler* system figures out the delivery time based on timetable of translators. The scheduler builds a special data structure to ~~outperform~~ calculate/estimate the response time. The detail of the process is out of scope of the paper. At last, *Profiler* accumulates personal information, translators' preferences as well as offered price and translation duration per word.

Beside translator, a proofreader selected by the client revises the final translation. As well as reviewing, the proofreader assesses the quality of translation from different points of view (grammar, style, accuracy, content and language). The assessment is defined as a value between ~~one~~ 1 (very bad) ~~to~~ and ~~five~~ 5 (perfect). As it is discussed in Section IV, it is used to evaluate aggregation algorithms for translator's proficiency.

III. Methods

In this section, we study different methods and algorithms regarding to Aggregation Functions and Learning to Rank.

A. *Aggregation Functions*

The aggregation function has a significant impact on the performance of Expert Retrieval system. As a usual scenario in expert retrieval systems, first each document related to an expert is scored and ranked regarding to query. Then, the top N document scores associated with a candidate expert are aggregated in order to rank the experts.

The effect of different features on aggregation function is studied in [4]. As it is shown, number of documents is tightly related such that the performance of different queries are optimal for different values of N. Comparing query-based features using statistical measures, it inferred that the features may not, in general, be able to predict the optimal number of documents to aggregate for each query. Individual Expert Features is discussed in the next step. It is shown that relevant experts are associated with a higher ranked document than non-relevant experts. More interestingly, relevant experts are associated with less documents on average.

Macdonald et al.[7] looks to expert search as a voting problem, where documents vote for the candidates with relevant expertise. Eleven data fusion as well as three statistically different document weighting were tested. In practice, the approach considers both number of documents and expert features regarding to the ranking score of the documents. The results show that while some of adapted

Algorithm	Top1	Top5	GP2
Spearman correlation	0.049	0.076	0.14

Table I
Comparison between algorithms and feedbacks

voting techniques are most likely outperform others, the proposed approach is effective when using appropriate one.

Later on, focusing on related features discussed before Cummins et al.[4] ~~comes up with a novel idea.~~ It uses genetic programming to learn a formula for the weights of document associations within the candidate profiles. The formula denoted as $GP2$ is as follows:

too colloquial

$$GP2 = \frac{\sqrt{\sqrt{2/no\_docs_{x_i}}}/(\sqrt{(10/R)+R}}{\sqrt{sq(10/R)+R+sq(10/R)+\sqrt{R*2}}}$$

where R is the rank of the document in the initial ranking and $no\_docs_{x_i}$ is the total number of documents associated with expert x_i.

B. *Learning To Rank*

Learning to rank refers to machine learning techniques for training the model in a ranking task. Due to its importance, learning to rank has been drawing broad attention in the machine learning community recently. TODO

IV. Apply The Methods and Results

In this section, we applied different approaches on the platform. By comparing the methods, we aim to discover the most appropriate one regarding to the project's characteristics and data.

A. *Aggregation Functions*

Similar to [4], outcome of three algorithms are compared. Top1 and Top5 are selected as two common forms TopN aggregation algorithm. TopN refers to algorithm that summarizes the N top documents (i.e. Top1 only using the top associated document to rank the candidates).

Feedbacks of Proof-readers after every translation are used as a basis for evaluating the algorithms. Since feedbacks are a measure for quality of translation, the more similar the ranking of algorithms to feedbacks are the better it is.

The result of applying Spearman correlation is shown in TableI. As the outcome shows, GP2 outperforms the other algorithms. In comparison to Top1, Top5 has slightly better performance. The results is also partly the same when comparing based on language-pairs.

B. *Learning To Rank*

TODO

V. Conclusion

TODO

References

[1] Balog, Krisztian, Yi Fang, Maarten de Rijke, Pavel Serdyukov, and Luo Si. *Expertise Retrieval - Foundations and Trends in Information Retrieval 6*, pages 127–256. Number 2–3. 2012.

[2] K. Balog, T. Bogers, L. Azzopardi, M. de Rijke, and A. van den Bosch. Broad expertise retrieval in sparse data environments. *SIGIR*, 2007.

[3] Y. Cao, J. Liu, S. Bao, and H. Li. Research on expert search at enterprise track of trec 2005. *TREC*, 2005.

[4] Cummins, Ronan, Mounia Lalmas, and Colm O'Riordan. Learning aggregation functions for expert search. *ECAI*, 2010.

[5] Deng, Hongbo, Irwin King, and Michael R. Lyu. Enhanced models for expertise retrieval using community-aware strategies. *Systems Man and Cybernetics Part B: Cybernetics IEEE Transactions*, 2012.

[6] Hang L. I. A short introduction to learning to rank. *IEICE TRANSACTIONS on Information and Systems*, 2011.

[7] Macdonald, Craig, and Iadh Ounis. Voting for candidates: adapting data fusion techniques for an expert search task. *the 15th ACM international conference on Information and knowledge management*, 2006.

[8] Macdonald, Craig, and Iadh Ounis. Learning models for ranking aggregates. *Advances in Information Retrieval. Springer Berlin Heidelberg*, 2011.