# Job Information Retrieval based on Document Similarity

Jingfan Wang[1], Yunqing Xia[2], Thomas Fang Zheng[2] and Xiaojun Wu[2]

[1] Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
wangjf@cst.cs.tsinghua.edu.cn
[2] Center for Speech and Language Technologies, RIIT, Tsinghua University, Beijing 100084, China
{yqxia, fzheng}@tsinghua.edu.cn
wuxj@cst.cs.tsinghua.edu.cn

**Abstract.** Job information retrieval (IR) exhibits unique characteristics compared to common IR task. First, searching precision on job posting full text is low because job descriptions cannot be properly used in common IR methods. Second, job names semantically similar to the one mentioned in the searching query cannot be detected by common IR methods. In this paper, job descriptions are handled under a two-step job IR framework to find job postings semantically similar to seeds job posting retrieved by the common IR methods. Preliminary experiments prove that this method is effective.

## 1 Introduction

Similar to common information retrieval (IR), job information retrieval aims to help job seekers to find job postings on the Web promptly. The task is made unique due to the following two characteristics. Firstly, job names are usually used as search queries directly in job IR. However, they can be expressed by numerous alternatives in natural language. For example, *manager* can be worded as "经理","主管", "总监" and "主任" in Chinese. As an extreme case, the job name "美工(*art designer*)" holds nine semantically similar job names. According to our study on query log, people with different background hold different preferences in selecting job names. This brings job IR a challenging issue to find job postings with conceptually similar job names but not necessarily with literally same job name in the query.

Secondly, job posting usually comprises of two fields, i.e. title and description. The title field is pretty short (1 to 6 words) and presents the most important points for the job while the description is a bit longer (20 to 100 words) and provides detailed requirements of the job. The most interesting point is that, the job name is usually contained in the title only and is scarcely mentioned in the description. To summarize, title and description depict the same job but share very few common words.

Problems arising from the two characteristics of job IR are two-fold. First, the title field provides too short text for the vector space model (VSM) to locate similar job

postings. Second, as it shares little common word with job name, job description provides very little contribution in finding the relevant job postings directly. This is also proved by our experiments (see Section 4), which shows that searching in job posting full text (title and description) yields very little performance gain over searching merely in the title. Discarding the job descriptions is certainly not a good idea, then how could we make use of job description properly?

In this paper we propose to make use of document similarity to locate relevant job postings. The basic assumption is that job description usually provides sufficient and unambiguous information, referred to as semantic clues behind the job name. We argue that the semantic clues can be used to find similarity job postings. In our job IR system, a two-step framework is designed to retrieval this goal. In the first step, queries are used to locate literally relevant job names. In the second step, the job posting full text is used to find relevant job postings. To re-rank all relevant job postings, a combined ranking model is proposed, which considers query-document relevance score and document-document similarity score in one formula.

The rest of the paper is organized as follows. The unique job IR task is described in Section 2. Then the two-step method for finding the similar job postings is presented in Section 3. In Section 4, experiments and discussions are presented. We summarize the related works in Section 5 and conclude this paper in Section 6.

## 2 Job Information Retrieval, a Unique IR Task

Job information retrieval system aims to facilitate job seekers to find job postings in a large scale online job posting collection. Basically, the job seekers type in job names as the queries directly.

The job posting is a piece of natural language text that contains two fields, title and job description. Two typical example job postings are given as follows.

---

**Job posting example 1:**

<title>软件工程师 (Software Engineer)</title>

<description>熟练掌握 Java, j2EE；熟悉 Eclipse 插件开发优先；富有激情；良好的团队合作精神,中英文流利 (Strong in programming with Java，J2EE; Priority to those who are familiar with Eclipse plug-in development; Self-motivated; Excellent teamwork spirit and communication skills; Fluent in English and Chinese)</description>

**Job posting example 2:**

<title>程序员(Programmer)</title>

<description>须有 Java 开发及 Eclipse 插件开发相关经验，有 RFT 使用经验者优先；计算机或相关专业研究生(Experienced in Java programming；

Experience in Eclipse plug-in development, Priority to those familiar with RFT; Master of Computer Science or related) </description>

As shown in the two examples, job names mostly appear only in the title field. Since most users use job name as query keywords directly, only job postings containing the job name within title field can be successfully retrieved by the traditional VSM. Another finding is that, the two job postings are semantically similar. Users who are interested in the one may also be interested in the other. Unfortunately, they can not be retrieved with one query using VSM because their job name strings are literally different.

Text in the description field is a bit longer, and semantic clues can be found such as professional experience, technical skills and education background. The semantic clues cannot be properly used in the VSM based query-document relevance measuring scheme, but helpful in finding semantically relevant job postings.

Our observations on job postings provide two assumptions: 1) similar job names hold semantically similar job descriptions; 2) semantically similar job descriptions in turn determine similar items. Enlightened by the two assumptions, we designed a two-step framework for job IR. The traditional VSM is applied on the title field in the first step, and similarity between job postings over full text is calculated to find the semantically similar job postings in the second step.

## 3 Finding Relevant Job Postings

### 3.1 The Two-Step Framework and the Combined Ranking Model

The key ideas of the two-step framework are summarized as follows.

(1) Each job posting is considered as a piece of semi-structured data comprising of two fields, i.e. title and description, which are treated differently in two steps.
(2) Query-document relevance score and document-document similarity score are combined to find semantically similar job postings.

Objective of job IR is achieved in two steps. In the first step, the standard VSM is applied on the title to retrieve relevant job postings according to query-document relevance score. Then job postings with relevance scores bigger than the threshold are selected as seeds for searching result expansion. In the second step, we calculate document similarity on full text to find the semantically similar job postings to the seed ones.

To re-rank the relevant job postings, a combined ranking model is proposed as follows, considering query-document relevance and document-document similarity in one formula as follow:

$$rel^*(q,d) = \max_{d_i}\{rel(q,d), rel(q,d_i) \times sim(d,d_i)\} \quad (1)$$

where $rel^*(q,d)$ denotes final relevance score between document $d$ and query $q$, $rel(q,d)$ the general relevance score between $d$ and $q$ calculated in the first step, and $sim(d,d_i)$ ($\in[0,1]$) the similarity score between document $d$ and $d_i$ calculated in the second step.

### 3.2 The First Step Job Information Retrieval based on VSM

In the first step, we retrieve job postings using the VSM. Two query-document relevance measures are implemented, i.e. cosine and inner product. As shown in [1][2], vector-length normalization causes a drop for cosine similarity when it is applied to very short string. So the inner product might be a good choice in our case. We calculate the classical *tf-idf* value as term weight.

As a result, relevant job postings are retrieved as well as relevance scores. We setup a threshold to get the seed job postings for further process in the second step.

### 3.3 Expanding Relevant Jobs Using Similarity between Job Postings

In this step, we use full text of each job posting to construct a *tf-idf* weighted feature vector, and attempt to find the job postings that are semantically similar to the seed ones by document similarity within the VSM.

**Features and Similarity Measures**

We choose two kinds of features, i.e. word and character bi-gram, which are proved by [3] to be the best feature types for Chinese text classification. We apply stop word list and finally obtain 25,000 word features and 140,000 character bi-gram features.

Two similarity measures are implemented in this paper, i.e. cosine and the extended Jaccard, which are found to be the best measures in the document cluster [5] and commonly used in Chinese text processing.

**Feature Selection**

A major characteristic of VSM is the high dimensionality rendering spare data problem. This problem is usually addressed by some automatic feature selection schemes. Yang and Pedersen [4] prove that feature selection technology can improve performance of text classification. In our work, two feature selection schemes are implemented, i.e. *DF* (document frequency) and $\chi^2$ statistics (Chi-square) [4].

*DF* is the number of documents where a feature occurs. Terms with low *DF* score will be eliminated in this feature selection scheme.

$\chi^2$ statistics originally estimates how one feature is independent from one class in text classification. For our case, we apply a clustering algorithm to generate the class labels required in $\chi^2$ calculation. The $\chi^2$ score for the term $t$ and the class label $c$ is defined as follows.

$$\chi^2(t,c) = \frac{N(AD-BC)^2}{(A+C)(B+D)(A+B)(C+D)} \quad (2)$$

where $A$ denotes the number of documents with class label $c$ and containing feature $t$, $B$ is the number of documents with class label $c$ while not containing feature $t$, $C$ is the number of documents without class label $c$ and containing feature $t$, $D$ is the number of documents without class label $c$ and not containing feature $t$, and $N$ is the total number of documents. Finally goodness score for each feature is defined as the maximum cluster-specific $\chi^2$ score as follows.

$$\chi^2_{\max}(t) = \max_c \{\chi^2(t,c)\} \quad (3)$$

We compute *DF* and $\chi 2$ score for each unique feature and remove a certain proportion of features.

**Feature Re-weighting for Ad-hoc Retrieval**

The document similarity measure discussed above is independent from query thus can be calculated off-line. However, query contributes more or less to feature weighting. Carpineto [6] shows that features actually play different roles in automatic query expansion for ad-hoc retrieval. We thus propose to make use of query to re-weight features.

In our case, we use the top $N$ job postings obtained in the first step to select useful features. The usefulness score is calculated by the Rocchio's formula [7] as follows.

$$w_i^* = \alpha * w_{iq} + \frac{\beta}{|R|}\sum_{d_j \in R} w_{ij} \quad (4)$$

where $R$ denotes the pseudo-feedback job posting sets retrieved in the first step, $w_{iq}$ denotes the weight of term $t_i$ in the original query, and $w_{ij}$ the weight of term $t_i$ in document $d_j$, $\alpha$ and $\beta$ are two constants.

The top $K$ features with high score are deemed useful and their weights are doubled in our work.

### 3.4 Some Critical Issues

In the two-step framework, two critical issues are worth noting. We in fact combine the IR model and the similarity measure of two piece of document into one model. In the first step, no extra calculation is involved compared to VSM. In the second step, several similarity measures for relevant job expansion are implemented, most of which are independent from the query thus can be calculated off-line. The exception is the re-weighting scheme, where the similarity scores can be updated for the selected features, rather than be re-calculated between every pairs of documents on-line. Therefore, computational complexity of our method can be appropriately controlled.

The second issue is retrieval quality. In the two-step framework, quality of the first retrieval is crucial. We set an appropriate threshold to get enough number of the rele-

vant job postings as accurate as possible in the first retrieval. Meanwhile, the combined ranking model (see Section 3.1) is helpful to discard the false job postings.

# 4 Experiments

## 4.1 Setup

**Data**

Our job posting collection contains around 55,000 Chinese job postings downloaded from job-hunting websites including *ChinaHR* (www.chinahr.com) and *51Job* (www.51job.com). Title and description filed of each job posting can be detected by an HTML document parser. The query set contains 100 random queries, which in real applications are actually job names.

**Evaluation Criteria**

We use precision at top ranked *N* feedbacks, i.e. *p@N*, as evaluation criteria in our experiments. That is, for each of the 100 queries, we compute searching precision as percentage of job postings correctly retrieved in *top ranked N feedbacks*. To be practical, we set *N* as 1, 5, 10, 20, 30 and 40 our evaluation. Around 5000 job postings are judged manually whether they are relevant to the 100 queries.

## 4.2 Experiment 1: The First VSM Retrieval

In this experiment, we evaluate job IR methods on the title field vs. the full text using VSM. We use words as features and two query-document relevance measures, i.e. cosine and inner product. Experimental results are shown in Fig. 1.

Fig. 1 shows that searching on the full text obtains very little performance gain over that on title only. Two conclusions can be drawn. First, search intension can be reflected by the title rather than the description. Second, the description filed contributes very little in matching to the query using VSM though it is longer. This stimulates the idea to make use of the description in other manners.

Note that we use the VSM based on "*title + inner product*" as our baseline in the following experiments since it achieves relatively better performance at most points in Fig. 1.

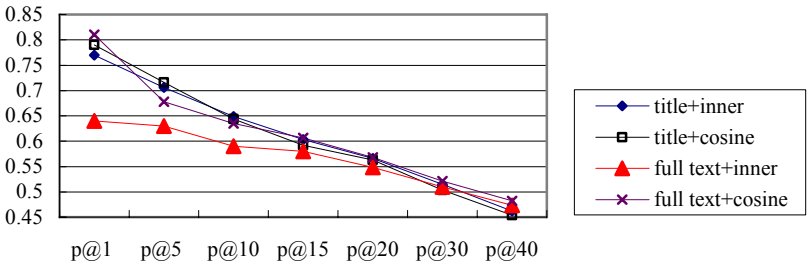


**Fig. 1.** Searching precision on title vs. full text using two similarity measures.

### 4.3 Experiment 2: Relevant Job Expansion

In this experiment we attempt to expand the relevant job postings starting from the seed job postings using document similarity.

We first evaluate our method on different features types, i.e. words and character bi-grams, with cosine as similarity measure. Experimental results are shown in Fig. 2.

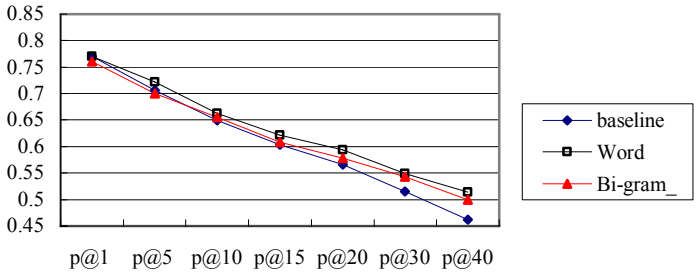


**Fig. 2.** Searching precision by expanding seed job postings using two feature types.

It is shown that 1) using similar job posting as expansion for seed job postings can improve searching quality; 2) word outperforms character bi-gram as feature type for document similarity measuring.

Using word as feature type, we then compare two document measures, i.e. cosine and the extended Jaccard. Experimental results are presented in Fig. 3. It is shown that cosine outperforms the extended Jaccard at all points.

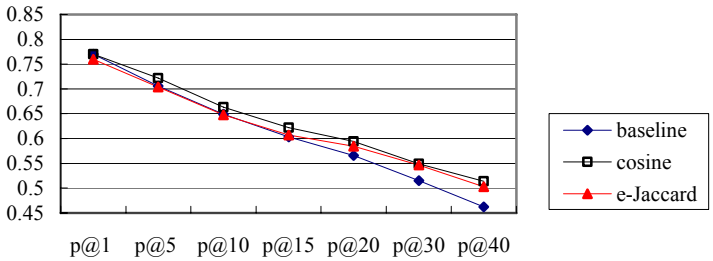


**Fig. 3**. Searching precision by expanding seed job postings using cosine vs. the extended Jaccard as document similarity measure.

### 4.4 Experiment 3: Feature Selection

In the following experiment, two feature selection schemes on word features are compared, i.e. *DF* and $\chi^2$ statistics (CHI). For $\chi^2$ statistics, we select k-1 repeated-bisection clustering method by the CLUTO package [8] to generate class labels. The experimental results are presented in Fig. 4.

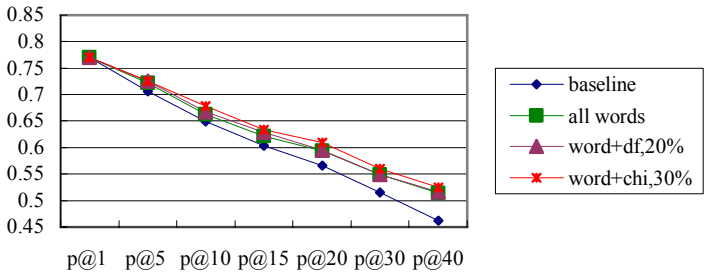


**Fig. 4**. Searching precision by expanding baseline searching results using two feature selection schemes. The percentages represent the proportions of features that remain after feature selection that yield best searching quality with certain setup.

Fig. 4 shows that both *DF* and $\chi^2$ statistics can remove more than 70% terms and improves searching quality. $\chi^2$ statistics on word improves most over baseline by 0.06 and over all-words by 0.011 at p@40.

It should be pointed out that the motivation to incorporate the clustering technique in our method is to separate the data set into a finite set of "natural" structure, namely clusters or subsets within job postings holding internal homogeneity and external separation, rather than accurate characterization or class label predefined as classification, so that the $\chi^2$ statistics based supervised feature selection methods can make use of the labels to estimate goodness score of each feature. We have tried several

clustering algorithms in CLUTO to obtain these labels. It is disclosed in our experiments that goodness of the clusters does not bias the feature selection much.

### 4.4 Experiment 4: Feature Re-weighting

In this experiment we investigate on the feature re-weighting scheme. We apply Rocchio's formula to select features with high usefulness score and double their weights if they are determined as useful. Experimental results are presented in Fig. 5.

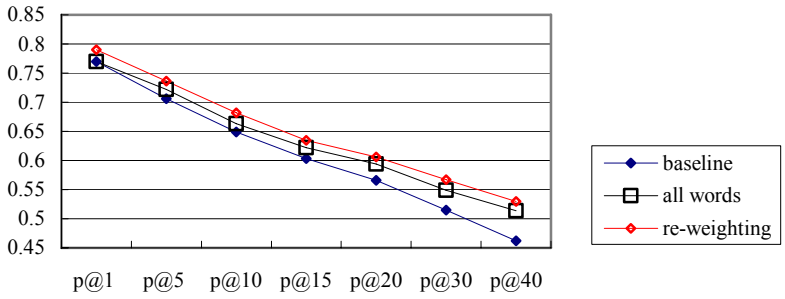


**Fig. 5**. Searching precision by expanding baseline searching results using Rocchio's formula.

It is shown in Fig.5 that feature re-weighting scheme improves by around 0.02 at every point over the method using all words as features. Compared to the baseline method, feature re-weighting scheme improves most by 0.06 at p@40.

### 4.6 Experiment 5: The All-in-one System

In this experiment, we evaluate our all-in-one job IR system which uses word as feature type, cosine as similarity measure, integrating $\chi^2$ statistics feature selection scheme and Rocchio's formula based feature re-weighting scheme.

To compare our method against the traditional query expansion method, the method based on traditional pseudo-relevance feedback is implemented. The method is briefed as follows. First, the initial search is performed to obtain the top *K* relevant documents, referred as pseudo-relevance feedback. Second, a number of terms are selected and reweighed from the feedback documents using certain scoring criteria to expand the initial query. Third, the expanded query is used to perform new search to get relevant documents. Corresponding to the first step within our method, we implement two methods to perform the initial search, i.e. using cosine as similarity measure on full text and inner product on job posting title. We do not perform query expansion on merely title field because the title field is too brief to yield valid extra query terms.

In this experiment, we implement several term-scoring functions [6][11] such as Rocchio, RSV, CHI, KLD, etc., in which Rocchio is found best in our job IR case. The experimental results are presented in Fig. 6.

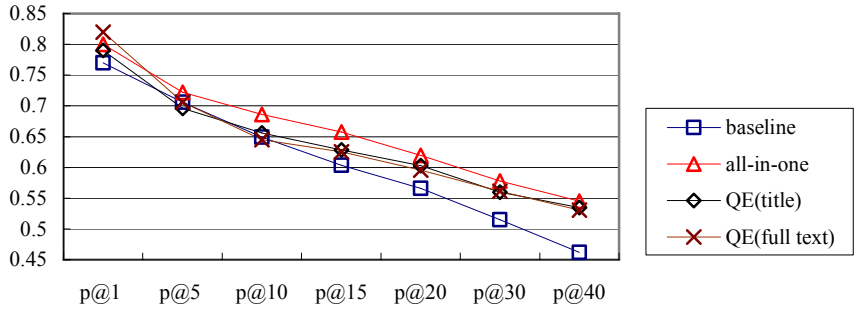


**Fig. 6.** Searching precision in baseline system vs. all-in-one system. QE(title) represents the query expansion method inner product as similarity measure on job posting title for initial search and QE(full text) the one using cosine on full text.

It is shown in Fig. 6 that our all-in-one scheme outperforms both traditional query expansion schemes at most points, in which p@40 is improved most by around 0.08325. This provides sufficient proof for the claim that our method for job IR is effective.

The second finding is that both query expansion methods outperform the baseline, in which the QE(title) outperforms the QE(full text). This accords to our results in the Experiment 1 where the "title+inner product" method outperforms "full text+cosine" at most points.

## 5 Related Works

The two-step framework we present in this paper is enlightened by the query expansion techniques [6][11], which have been used in the IR community for ages. The pseudo-feedback query expansion techniques also make use of the top documents to improve search performance, however, in a different way, that is, to use these documents to re-construct a new query first, while we apply document similarity to the pseudo-feedback documents to find the semantically similar job postings.

Feature and similarity measures within the VSM are explored in both IR and text categorization/cluster field. Term weighting in query-document relevance measuring is studies by [2][9]. Li et al. compare two feature types [3], i.e. word and character bigram in Chinese text categorization. Yang and Pedersen evaluate five feature selection methods [4], i.e. DF, IG, CHI, TS and MI, to reduce dimensionality of features in document categorization. In this work, we select DF and CHI as our feature selection method because it is an unsupervised method and CHI yields best performance in Yang's experiments. Strehl et al. compare four similarity measures on web-page

clustering [5], we use the cosine and e-Jaccard in our work which lead to best performance in their work. Besides, Liu et al. make use of clustering results as class labels so that the supervised feature selection methods can be applied in unsupervised way [10].

## 6 Conclusions and Feature Works

This paper presents a two-step framework for job IR, which in fact combine the IR model and the similarity measure schemes of two semi-structure documents together. In this work, we investigate on the most popular IR model, i.e.VSM, in job IR. Several document similarity measures commonly used in NLP fields are implemented including cosine and extended Jaccard. We also investigate on several feature selection and term re-weighting schemes in this work. The experiment results show that our all-in-one system outperforms all other methods in performing the task of job IR. Several other conclusions can be drawn as follows. Firstly, word is a better feature type than character bi-gram. Secondly, cosine is a better document similarity measure than the extended Jaccard here. Thirdly, feature selection schemes are helpful to improve accuracy of document similarity, in which $\chi^2$ statistics outperforms *DF*. Fourthly, feature re-weighting method is helpful for document similarity measuring. Finally, the traditional query expansion techniques are inferior to our method in the special job IR task.

Several future works are described as follows. Firstly, we will investigate on other IR models for the job IR task, such as the probabilistic models and language models. Secondly, we will investigate on information extraction techniques for the job IR task because the job postings are semi-structured and some job related information such as company information, responsibility, requirements, etc. can be easily recognized. We will try to use information of this kind to improve accuracy in job posting similarity measuring.

**Acknowledgement**
Research work in this paper is partially supported by NSFC (No. 60703051) and Tsinghua University under the Basic Research Foundation (No. JC2007049).

## References

1. B. Yuwono and D. L. Lee: WISE: A World Wide Web Resource Database System. In Proc. of ICDE-96, New Orleans, Louisiana.
2. G. Salton and C. Buckley: Term-weighting Approaches in Automatic Text Retrieval. Information Processing & Management 24,5,513-523 (1988).
3. J. Li, M. Sun and X. Zhang. A Comparison and Semi-quantitative Analysis of Words and Character-bigrams Features in Chinese Text Categorization. In Proc. Of ACL06, Sydney, Australia.
4. Y. Yang and J. O. Pedersen.: A Comparative Study on Feature Selection in Text Categorization. In Proc. of ICML-97, pp.412-420.

5. A. Strehl, J. Ghosh and R. Mooney.: Impact of Similarity Measures on Web-page Clustering. In Proc. of AAAI-2000 Workshop for Web Search, Austin.
6. C. Carpineto, R. de Mori, G. Romano and B. Bigian.: Information Theoretic Approach to Automatic Query Expansion. ACM trans. on Information System.2001: 19(1):1--27.
7. J. J. Rocchio.: Relevance Feedback in Information Retrieval. In Salton G. (Ed.), The SMART Retrieval System. 1971, Englewood Cliffs, N.J. : Prentice-Hall, Inc. pp313-323.
8. G. Karypis.: CLUTO: A Clustering Toolkit. Dept. of Computer Science, University of Minnesota, May, 2002.
9. J. Zobel and A. Moffat.: Exploring the Similarity Space. In Proc. Of SIGIR-1998, Melbourne, Australia.
10. T. Liu, G. Wu and Z. Chen.: An Effective Unsupervised Feature Selection Method for Text Clustering. Journal of Computer Research and Development, 2005,42(3),381~386.
11. J.H. Lee: Combining the evidence of different relevance feedback methods for information retrieval. Information Processing and Management, v34(6), pp.681 - 691.