Employment Recommendation System: A Review

Roshan G. Belsare  
Department of Computer Engineering  
PRMITR, Badnera  
Maharashtra, India  
roshanbelsare@gmail.com

Dr. V. M. Deshmukh  
Department of Computer Engineering  
PRMITR, Badnera  
Maharashtra, India  
vmdeshmukh@mitra.ac.in

Abstract: Enormous amounts of jobs are posted on the job websites on daily basis and large numbers of new resumes are also added to job websites daily. In such scenario it’s a very tough job to suggest matching jobs to the job applicants. A recommendation system can solve this problem to the great extent. A recommendation system has already been proved to be very effective in the area of Online shopping websites and Movie recommendation. Given a user, the goal of an employment recommendation system is to predict those job positions that are likely to be relevant to the user. An Employment recommendation system would suggest matching jobs to the users using matching, collaborative filtering and content based recommendation based on ranking.

Keywords: recommendation system, collaborative filtering, content based filtering, matching

1. INTRODUCTION
The recommender system technology plays an important role in various e-commerce applications by helping individuals to find right items in a large option space, which match their interests. Recommender systems are software tools and techniques providing suggestions for items to be of interest to a user such as videos, songs, or news articles. Driven by this success, more application domains have adopted recommender systems to reduce the information overload by generating personalized suggestions. Also for recruitment scenarios, in which applicants search for suitable job offers, recommender systems are a useful tool for job candidates, recruiters, as well as platforms that connect both [12]. The mainstream approaches to recommender systems are classified into four categories:

Collaborative Filtering (CF), Content-Based Filtering (CBF), knowledge-based and hybrid approaches [1]. Besides, utility-based and demographic approaches also exist. The main advantage of CF approaches is that they can find the patterns among user ratings data and work well for complex objects[2]. The problem of recommending jobs to users is fundamentally different from traditional recommendation system problems such as recommending books, products, or movies to users. While all of the above have a common objective to maximize the engagement rate of the users, one key difference is that a job posting is typically meant to hire one or a few employees only, whereas the same book, product, or movie could be potentially recommended to hundreds of thousands of users for consumption.[13] Ideal job recommendation system would need to achieve three goals simultaneously: (1) Recommend the most relevant jobs to users. (2) Ensure that each job posting receives sufficient number of applications from qualified candidates.

2. Related Work
Job Recommendation work resides in the domain of online recommender systems, which are widely adopted across many web applications, e.g., movie recommendations [14], e-commerce item recommendations [15], job recommendations [16] and so forth, where authors mainly concentrate on the relevance retrieval and ranking aspects of the recommendation system. There is insightful research and modeling of the
Collaborative filtering can be categorized into basic types namely Item based collaborative filtering and user based collaborative filtering. In the item based collaborative filtering similar items are found out to recommend it to users and in Content based recommendation users past activities are analyzed to suggest new recommendations.

3.2 Content Based (CB) Recommenders: They utilize item features to recommend items similar to those a user has liked in the past. A CB system analyzes a set of characteristics of items that are rated by a user and build the profile of the user interests based on the features of the items that are rated by her [7]. The recommendation process matches up the attributes of the user profile against the set of properties of a content item [8], [9],[10].

3.3 Knowledge-based (KB) Recommenders: Knowledge-based (KB) Recommenders use domain knowledge to generate recommendations.

3.4 Hybrid Recommender systems:

Hybrid Recommender systems combine two or more techniques to gain better results with fewer drawbacks.

4. Employment recommendation system challenges

The job matching process normally takes into consideration of the data available in the resume and match against the data listed in the list of open vacancies. One of the most challenging tasks of this type of job matching is that there are usually too many data to match against. Furthermore, these data usually submitted in free form, as each individual has their own preference to prepare the data. For example, Person A claimed that he has a total of 8 years’ experience in Oracle product. A Database Administrator will interpret that this person has 8 years of experience in Oracle Database. A Platform Leader will make assumption that it’s 8 years of experience in Oracle Commerce Platform, while a Lead Programmer will think that is a 8 years of experience in Java programming. Therefore, an intelligent job matching engine is required to overcome this issue.

Recommending a job is different than recommending a product or movies as it involves large number of parameters and filtering based on different criteria.

A same job cannot be recommended to all the people all over the world as demographic area also needs to be considered for recommendation of a particular job to particular user.

5. Employment recommendation using Collaborative Filtering

The traditional Item-based CF processes as follow: First, for each job which current user (useri) applied in the past (we regard user-applied jobs as user-liked jobs), find out other users who applied this job (userj) (we regard these users as co-applied users), and then find out other jobs these co-applied users also applied, except for the current job (user-liked jobs), uses these jobs as candidate set. The procedure is presented below:

```java
for each job: user: applied
  for each co-applied user; who applied job:
    find out jobs(job) that user; applied;
    add these jobs to candidate set;
  
  delete job; from candidate set;
```

Second, for every job Itemi in the candidate set {Item1…Itemn}, compute the predict preference grade for it using The Jaccard similarity index. Jaccard similarity is calculated using formula:

Number of user common for jobs and jobi (intersection) divided by number of users either for jobs or jobi (union)

At last, sort all the grades and choose top N jobs as the result set. The procedure is presented below:

```java
for each item, in candidate set
  compute pref (U,Item);
sort these pref (U,Item);
```

6. Employment recommendation using Content based Filtering

One of the most popular recommender approaches is content-based filtering, which exploits the relations between (historically) applied jobs and similar features among new job opportunities for consideration. Generally speaking, the goal of content based filtering is to define recommendations based upon feature similarities between the items being considered and items which a user has previously rated as interesting. For the target user-item rating f(“u”, “i”) , content-based filtering would predict the optimal recommendation based on the utility functions of f(“u”, I h) which is the historical rating information of “u” on items (I h) similar with “i”. [23] Given their origins out of the fields of information retrieval and information filtering, most content-based filtering systems are applied to items that are rich in textual information.
7. REFERENCES


