International Journal of Science and Engineering Applications
Volume 7 – No.5, 64-67, May 2018, ISSN: 2319 - 7560

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 hiring processes within job marketplaces. Such research includes work related to estimation of employee reputation for optimal hiring decisions [17], as well as work related to ranking and relevance aspects of job matching in labor market places [18, 19, 20]. There has been work related to the theory of optimal hiring process, e.g., on the problem of finding the right hire for a job (the hiring problem), as well as on the classical secretary problem, where a growing company continuously interviews and decides whether to hire applicants [21, 22].Authors of [23] investigated job marketplace as a two-sided matching market using locally stable matching algorithms for solving the problem of finding a new job using social contacts.

3. Types of Recommender Systems:

Distinction of four basic algorithm types has been proposed in RS [6]:

3.1 Collaborative Filtering (CF) Recommenders: They utilize social knowledge (typically ratings of items by a community of users) to generate recommendations.

A new user is matched against a database to discover neighbors, i.e. other users who, historically, had similar interests with him. Thus, the items that his/her neighbors liked are recommended to the user because he/she will probably like them too [7]. Fig. 1 illustrates the CF Recommendation concept. [11]

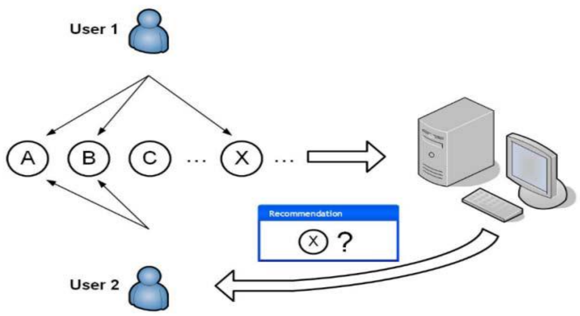

Fig. 1. User 1 selected A, B and X items, User 2 selected A and B, the CF system will suggest X item to User 2

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 find 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].

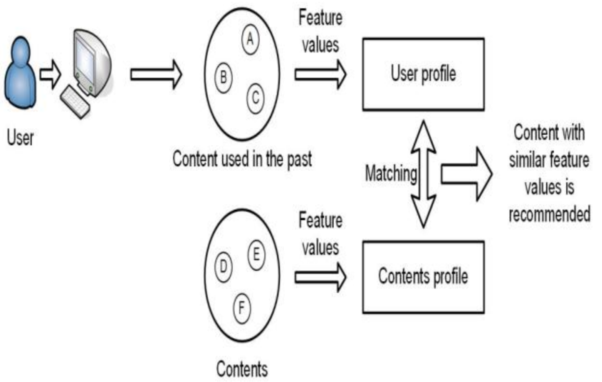

Fig.2. The CB recommender utilizes item features to recommend items similar to those a user has liked in the past Collaborative filtering can work

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:

for each job$_i$ user$_i$ applied
{
for each co-applied user$_j$ who applied job$_i$
{
find out jobs(job$_j$) that user$_j$ applied;
add these jobs to candidate set;
}

delete job$_i$ from candidate set;
}

Second, for every job Item$_j$ in the candidate set {Item1…Item$_p$}, compute the predict preference grade for it using The Jaccard similarity index. Jaccard similarity is calculated using formula:
Number of user common for *job$_i$ and job $_j$* (intersection) divided by number of users either for *job$_i$* or *job$_j$* (union)
At last, sort all the grades and choose top N jobs as the result set. The procedure is presented below:

for each item$_j$ in candidate set
{
compute pref (U$_i$,Item$_j$);
}
sort these pref (U$_i$,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(ˆ u ,ˆ i) , content-based filtering would predict the optimal recommendation based on the utility functions of f(ˆ u , I h) which is the historical rating information of user ˆ u on items (I h) similar with ˆ 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

[1] K. Wei, J. Huang, and S. Fu. A survey of e-commerce recommender systems. In 2007 International Conference on Service Systems and Service Management, pages 1{5, June 2007.

[2] Chenrui Zhang , Xueqi Cheng An Ensemble Method for Job Recommender Systems. RecSys Challenge '16, September 15 2016, Boston, MA, USA 2016 ACM

[3] N. D. Almalis, G. A. Tsihrintzis and N. Karagiannis, "A content based approach for recommending personnel for job positions," IISA 2014, The 5th International Conference on Information, Intelligence, Systems and Applications, Chania, 2014, pp. 45-49.

[4] M. Balabanovic, and Y. Shoham, "Fab: Content-based, Collaborative Recommendation. Communications of the ACM," vol. 40, no. 3, pp. 66- 72, 1997.

[5] Sovren Group, "Overview of the Sovren Semantic Matching Engine and Comparison to Traditional Keyword Search Engines," Sovren Group Inc, 2006.

[6] M. Ramezani, L. Bergman, R. Thompson, R Burke, and B. Mobasher, "Selecting and Applying Recommendation Technology," In proceedings of International Workshop on Recommendation and Collaboration in Conjuction with International ACM on Intelligence User Interface, 2008.

[7] BadulSarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-Based Collaborative Filtering Recommendation Algorithms," Proceedings of the 10th International Conference of World Wide Web, pp. 285-295, 2001.

[8] G. Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," IEEE Internet Computing, vol. 7, no. 1, pp. 76–80, 2003

[9] D. Mladenic, "Text-learning and Related Intelligent Agents: A Survey," IEEE Intelligent Systems, vol. 14, no. 4, pp. 44–54, 1999.

[10] RJ. Mooney and L. Roy, "Content-Based Book Recommending Using Learning for Text Categorization," in Proceedings of DL "00: Proceedings of the Fifth ACM Conference on Digital Libraries, New York, NY, ACM pp. 195-204, 2000.

[11] N. D. Almalis, G. A. Tsihrintzis and N. Karagiannis, "A content based approach for recommending personnel for job positions," IISA 2014, The 5th International Conference on Information, Intelligence, Systems and Applications, Chania, 2014, pp. 45-49.

[12] Toon De Pessemier, Kris Vanhecke, and Luc Martens. 2016. A scalable, high-performance Algorithm for hybrid job recommendations. In *Proceedings of the Recommender Systems Challenge* (RecSys Challenge '16). ACM, New York, NY, USA, Article 5, 4 pages. DOI: https://doi.org/10.1145/2987538.2987539

[13] Fedor Borisyuk, Liang Zhang, and Krishnaram Kenthapadi. 2017. LiJAR: A System for Job Application Redistribution towards Efficient Career Marketplace. In Proceedings of KDD '17, Halifax, NS, Canada, August 13-17, 2017, 10 pages. https://doi.org/10.1145/3097983.3098028

[14] Carlos A. Gomez-Uribe and Neil Hunt. 2015. The Netflix Recommender System: Algorithms, Business Value, and Innovation. ACM Trans. Manage. Inf. Syst. (2015). https://doi.org/10.1145/2843948

[15] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon.Com Recommendations:Item-to-Item Collaborative Filtering. IEEE Internet Computing 7, 1 (2003), 76–80. https://doi.org/10.1109/MIC.2003.1167344

[16] Fedor Borisyuk, Krishnaram Kenthapadi, David Stein, and Bo Zhao. 2016. CaSMoS:A Framework for Learning Candidate Selection Models over Structured Queries and Documents. In KDD. https://doi.org/10.1145/2939672.2939718

[17] Maria Daltayanni, Luca de Alfaro, and Panagiotis Papadimitriou. 2015. WorkerRank: Using Employer Implicit Judgements to Infer Worker Reputation. In WSDM. https://doi.org/10.1145/2684822.2685286

[18] Viet Ha-Thuc, Ye Xu, Satya Pradeep Kanduri, Xianren Wu, Vijay Dialani, Yan Yan, Abhishek Gupta, and Shakti Sinha. 2016. Search by Ideal Candidates: Next Generation of Talent Search at LinkedIn. In WWW. https://doi.org/10.1145/2872518.2890549

[19] Marios Kokkodis, Panagiotis Papadimitriou, and Panagiotis G. Ipeirotis. 2015. Hiring Behavior Models for Online Labor Markets. In WSDM. https://doi.org/10.1145/2684822.2685299

[20] Jia Li, Dhruv Arya, Viet Ha-Thuc, and Shakti Sinha. 2016. How to Get Them a Dream Job?: Entity-Aware Features for Personalized Job Search Ranking. In KDD. https://doi.org/10.1145/2939672.2939721

[21] Andrei Z. Broder, Adam Kirsch, Ravi Kumar, Michael Mitzenmacher, Eli Upfal, and Sergei Vassilvitskii. 2008. The

Hiring Problem and Lake Wobegon Strategies. In SODA. https://doi.org/10.1137/07070629X

[22] Ravi Kumar, Silvio Lattanzi, Sergei Vassilvitskii, and Andrea Vattani. 2011. Hiring a Secretary from a Poset. In EC. https://doi.org/10.1145/1993574.1993582

[23] Esteban Arcaute and Sergei Vassilvitskii. 2009. Social Networks and Stable Matchings in the Job Market. In WINE. https://doi.org/10.1007/978-3-642-10841-9_21

[24] Shuo Yang a, Mohammed Korayem b , Khalifeh AlJadda , Trey Grainger, Sriraam NatarajanCombining 2017 : content-based and collaborative filtering for job recommendation system: A cost-sensitive Statistical Relational Learning approach http://dx.doi.org/10.1016/j.knosys.2017.08.017