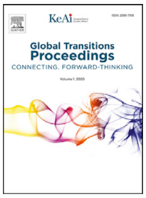
Global Transitions Proceedings 3 (2022) 371–375

Contents lists available at ScienceDirect

Global Transitions Proceedings

journal homepage: http://www.keaipublishing.com/en/journals/global-transitions-proceedings/



Design and development of machine learning based resume ranking system

Tejaswini K[a,*], Umadevi V[b], Shashank M Kadiwal[a], Sanjay Revanna[a]

[a] Department of Computer Science and Engineering, Dr. Ambedkar Institute of Technology, Bengaluru 560 056, Karnataka, India
[b] Department of Computer Science and Engineering, B.M.S. College of Engineering, Bengaluru 560 019, Karnataka, India

ARTICLE INFO

Keywords:
Resume ranking
Recommender systems
Hiring
Machine learning
Cosine similarity
KNN

ABSTRACT

Finding acceptable applicants for a vacant job might be a difficult process, especially when there are many prospects. The manual process of screening resumes could stymie the team's efforts to locate the right individual at the right moment. The laborious screening may be greatly aided by an automated technique for screening and ranking applicants. In our work, the top applicants might be rated using content-based suggestion, which uses cosine similarity to find the curriculum vitae that are the most comparable to the job description supplied and KNN algorithm is used to pick and rank Curriculum Vitaes (CV) based on job descriptions in huge quantities. Experimental results indicate the performance of the proposed system as an average text parsing accuracy of 85% and a ranking accuracy of 92%.

1. Introduction

In Human Resources (HR), talent acquisition is a big, complicated, and time-consuming activity. The size of the Indian market is mind-boggling [4]. Not only do one million people join the workforce every month, but the attrition rate is also significant. India has the largest percentage of workers who are "actively seeking a new job," according to LinkedIn [11]. Even though it is a vast and liquid market, there are a few unpleasant inefficiencies. The lack of a standard CV format and style is the most difficult component, which makes shortlisting of potential profiles for the desired positions tiresome and time-consuming [12]. Effective resume screening demands subject expert to assess the fit and applicability of a profile to the post. Shortlisting is difficult due to the wide range of career opportunities available today, as well as enormous number of applications received for the human resource department. The possibility of eliminating irrelevant profiles as quickly as possible in the process saves money and time [2].

1.1. Recommendation systems

The Resume Screening system is built using recommendation system mechanisms, specifically content-based filtering recommendation systems. A recommendation engine, often known as a recommender system, is a type of information filtering system that tries to predict a user's "rating" or "preference" for an item. Recommender systems in use include playlist generators for video and music services, product recommenders for online merchants, content recommenders for social media platforms, and open web content recommenders. Inside and across platforms, these systems can work with a single input, such as music, or multiple inputs, such as news, books, and search queries. There are other popular recommender systems for specific topics like restaurants and online dating [8]. Recommender systems have been used to investigate research papers, experts, collaborators, and financial services.

There are two different types of Recommendation Systems

1.1.1. Content-based recommender system

A content-based recommender takes use of data provided by the user, either directly (ratings) or indirectly (search results). Based on this information, a user profile is created, which is then utilized to provide recommendations to the user. The engine grows more accurate as the user offers more inputs or acts on the recommendations [9].

1.1.2. Collaborative filtering recommender system

Recommender systems, which advice things based on consumer collaboration, are the most widely used and well-established method of sending suggestions.

Two types of recommendation systems are pictorial represented as shown in figure 1.

In this paper, content based Filtering is used for recommendation purpose.

2. Related work

Every job advertisement receives a significant number of applications, many of which are related to the listed position [1]. Because they must identify the most qualified profile/resume from a broad pool

[*] Corresponding author.
E-mail address: teju01.kc@gmail.com (T. K).

https://doi.org/10.1016/j.gltp.2021.10.002
Received 13 August 2021; Received in revised form 9 October 2021; Accepted 11 October 2021
Available online 14 October 2021

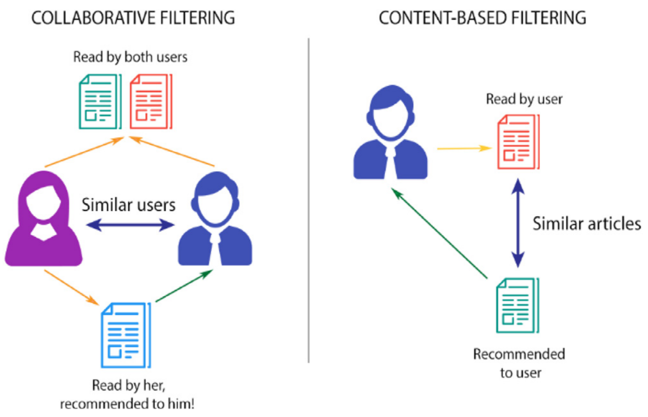


Fig. 1. Collaborative filtering vs content based filtering

Nomenclature

CV	Curriculum Vitae
HR	Human Resource
NLP	Natural Language Processing
AI	Artificial Intelligence
ML	Machine Learning
EM	Expectation Maximization
KNN	K Nearest Neighbours
MCQ	Multiple-Choice Questions
TF-IDF	Term Frequency- Inverse Document Frequency
JD	Job Description

of prospects, job recruiters encounter substantial challenges [5]. Because it is the profile of the applicant recommended for a specific role, the method of matching the candidate CV with the job description is similar to a recommender system [10]. Resnick et al. [18] developed the first recommendation system. The recommendation system is currently widely utilized in e-commerce products suggestions, book recommendations, news recommendations, movie recommendations, and music recommendations, to name a few examples. Authors discussed the widespread use of recommendation algorithms in real-time applications. The four primary forms of recommendation services are collaborative filtering, content-based filtering, knowledge-based filtering, and hybrid approaches. Wei et al. [13] discussed the many types of recommendation algorithms and how they function in great detail. Otaibi et al. [14] investigated the utilization of employment referral services in depth and discussed about the measures that must be taken during the hiring process for any organization and also described how the organization benefits from the e-recruitment portal, what candidate criteria might lead to the selection, and a variety of other essential recruiting approaches. To produce employment suggestions, Malinowski et al. [15] used an Expectation-Maximization (EM) algorithm that took into account both the candidate's resume and the employer's job description. Golec et al. [16] suggested a fuzzy-based method for determining candidate relevance to the job description.

3. Problem statement

The biggest issue facing the business today is how to discover the best people with minimum resources, through the internet, and in a short amount of time. To introduce efficiency to the whole process, two primary difficulties must be addressed.

- Deciphering their CVs
- Knowing that applicants can execute the job before the company employs them.

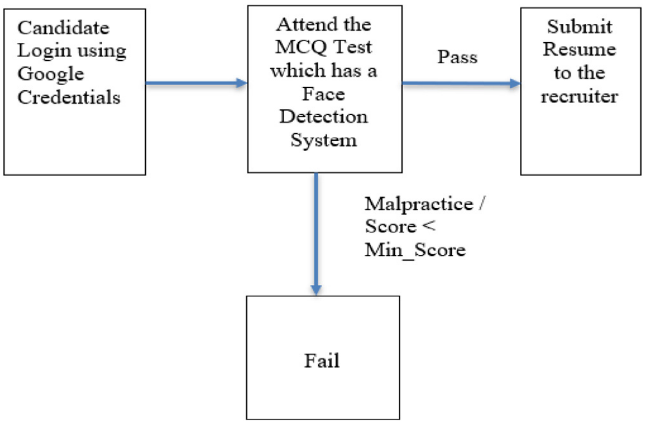

Candidate Login using Google Credentials
Attend the MCQ Test which has a Face Detection System
Pass
Submit Resume to the recruiter
Malpractice / Score < Min_Score
Fail

Fig. 2. Candidate screening system.

The foregoing issues are solved by automating the process of identifying the correct resume from enormous databases of resumes, regardless of how the resume was written, and generating a list of resumes that best suit the recruiter's job description. In the offered solution, algorithms are employed to rate resumes.

4. Objective and scope of the proposed work

The Objective of our work is, to identify the most qualified candidates for a certain vacancy. Recruiters must be able to thoroughly examine resumes in order to hire the right person at the right time. Every great recruitment strategy revolves around the requirement for efficient and effective resume screening.

5. Methodology

5.1. *Candidate screening system*

Fig. 2 depicts a system to screen the candidates using MCQ bases test to test their basic knowledge and understanding of the subject. The candidate screening system is used to screen the candidate based on their skills which has a face detection system to detect any malpractice during the screen test. Once the candidate achieves a minimum score the candidate can submit resumes which get uploaded on the Database. In case the candidate fails to attain a minimum score, the candidate is not allowed to submit resume.

5.2. *Resume screening and ranking system*

Framework of the proposed system is as shown in Fig. 3. Resumes from the Data set are parsed to remove white spaces, numbers, stop words like and, or, etc. TF-IDF vectorization is then applied to convert the words in the resumes to vectors. The text in the job description is also converted to vectors using TF-IDF vectorizer. Cosine distance is computed to measure the similarity between the resume and the job description provided and Then KNN algorithm is applied to identify the resumes which are closely matching with the JD provided by the recruiters.

5.2.1. *TF-IDF vectorizer*

The TF-IDF method is the most frequently used method for determining word frequencies. This is an abbreviation for "Term Frequency – Inverse Document" Frequency, one of the criteria used to determine the final score for each word [7]. TF-IDF are word frequency scores that aim to emphasize phrases that are more interesting, e.g., common in a text but not across texts, without delving into the arithmetic. The TF-IDF Vectorizer tokenizes texts, learns vocabulary, inverts frequency weightings, and allows encoding new ones.

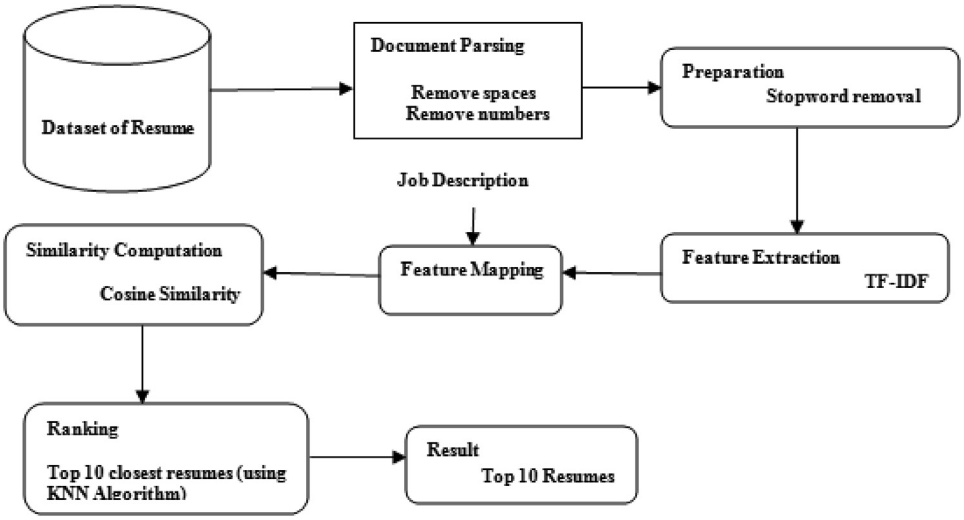


Fig. 3. Framework of the resume screening and ranking system.

Term Frequency: The term Frequency of a word refers to how many times it appears in a document.

Inverse Document Frequency: Inverse document frequency refers to downscale terms that appear frequently in documents.

$$TF-IDF(t,d)=TF(t,d)*IDF(t,d) \tag{1}$$

$$\text{TF (t, d)} = \frac{\text{freq (t, d)}}{\sum \text{freq (ti, d)}} \tag{2}$$

$$\text{IDF(t)} = \log\left(\frac{N}{count(t)}\right) \tag{3}$$

Where freq (t, d) is the count of the instances of the term t in document d.

TF (t, d) is the proportion of the count of term t in document d

N is the number of distinct terms in document d.

5.2.2. Cosine similarity

A measure of similarity between two non-zero vectors in an inner product space is cosine similarity. It's equal to the cosine of the angle between them, which is the same as the inner product of the same vectors normalized to have the same length. The cosine of 0° is 1, and the cosine of any angle in the range (0,π] radians is less than 1. It is thus a judgment of orientation rather than magnitude: two vectors with the same orientation have a cosine similarity of 1, two vectors orientated at 90° to each other have a similarity of 0, and two vectors opposed have a similarity of -1, regardless of magnitude. In positive space, where the result is cleanly limited in display style [0,1] [0,1], the cosine similarity is very useful. The name comes from the phrase "direction cosine": unit vectors are maximum "similar" if they are parallel and maximally "dissimilar" if they are orthogonal in this case (perpendicular). The cosine, which is unity (highest value) when the segments subtend a zero angle and zero (uncorrelated) when the segments are perpendicular, is equivalent to this. The term frequency vectors of the documents are usually the attribute vectors A and B for text matching. Cosine similarity can be thought of as a way to normalize document length when comparing them using equation (4) [17].

$$\cos(\theta)=\frac{\vec{a}.\vec{b}}{\|\vec{a}\|\|\vec{b}\|}=\frac{\sum_{i=1}^{n}a_ib_i}{\sqrt{\sum_{i=1}^{n}a_i^2}\sqrt{\sum_{i=1}^{n}b_i^2}} \tag{4}$$

Where, a_i and b_i are the components of the vector A and B respectively. General form of Cosine Distance/Similarity is represented in Fig. 4. Here X_1 is $feature_1$ and X_2 $feature_2$. In this case X_1 is the resume and X_2 is the job description provided by the resume and the $Item_1$ are the words which is converted into vectors present in the resume and $Item_2$ are the words which is converted into vectors present in the job description.

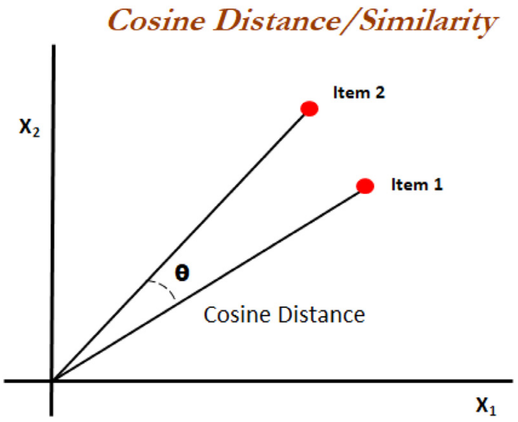


Fig. 4. Cosine distance/similarity.

5.2.3. KNN

KNN is a lazy learning approach that is non-parametric. It creates conclusions for fresh samples using a database containing data points split into distinct clusters. KNN focuses on item feature similarity rather than making assumptions about the underlying data distribution. When inferring about a resume, KNN calculates the "distance" between the target and every other resume in its database, ranks the distances, and delivers the top K most comparable resume choices [6].

5.2.4. Data set

Collecting the Resumes was the most important task in this process. Around fifty resumes were collected from Kaggle belonging to Java Developer and Project Manager Roles which were in doc and docx formats. The resumes collected were then converted into pdf format using bash script for easy handling of the data.

6. Implementation

In this model, KNN is used to find resumes that are the most similar to the job description supplied or resumes that are a close match to the job description. To begin, we used an open-source program called "genism" to scale the JD and resumes. This library creates a summary of the supplied information within the word limit. KNN was used to locate resumes that closely matched the JD. The recommended technique is best suited for the recruiter's first resume review. The recruiter would be able to assess resumes based on job criteria and quickly find those that best match the job description. The "Genism" module was used to summarize the collected text. Summarization is a textual technique that tries to emphasize significant information within a big corpus.

6.1. Text pre-processing

This includes removing stop words, punctuation, and stemming. This process will construct a graph with sentences as the vertices. At the vertices of the network are edges that indicate how similar the two phrases are. Select the vertices with the greatest scores and add them to the summary. The number of vertices to be chosen is determined by the ratio or word count.

6.2. TF-IDF calculation

Using equations (1), (2) and (3), TF-IDF is calculated. It provides information of a word frequency in the documents [3]

Higher the TF- IDF score of a term which is computed using above equations represents more relevance of in a document. In our system, we modelled the CVs and JD into a vector space. This is accomplished by compiling a glossary of terms found in the papers and converting them. Each phrase corresponds to a vector space dimension. Using the Count Vectorizer and the TF- IDF matrix, we generated the TF- IDF matrix for the CVs and the job query.

6.3. Resume recommendation is done using two approaches

6.3.1. Recommendation based on content using cosine similarity

In this approached, the employer's Job Description is matched against the content of resumes in the space, and the top n matching resumes are recommended to the employer. The model merges the cleansed resume data and JD into a single data set before computing the cosine similarity between the JD and resumes. Cosine similarity is calculated between each set of resume vectors and job descriptions which are represented in vector space containing 'n' using equation (4). According to similarity score obtained resumes are ranked with the job description. Based on assigned ranks, the top n ranked resumes are recommended to the recruiters.

6.3.2. Using KNN

KNN is used to find the CVs that are closest to the job description provided. To begin, we utilised an open source tool called "gensim" to scale the Job descriptions and resumes. This library generates a summary of the provided text within the word limit. To bring the Job descriptions and resumes to a similar word scale, this library was used to generate a summary of the Job descriptions and resumes, and then KNN was used to locate CVs that closely matched the provided Job descriptions.

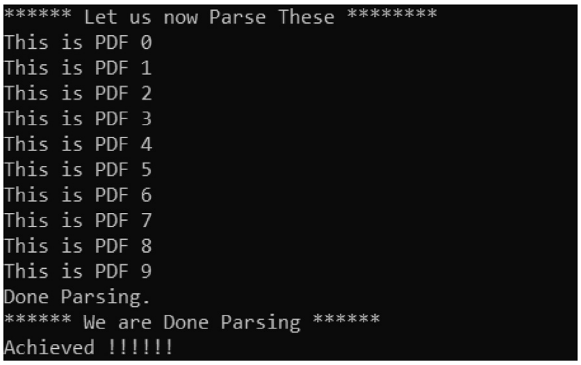
7. Results and discussions

We have used our own database (as mentioned in *5.2.3)* of resumes for the training as well as testing purpose. Resumes are parsed to remove the stop words, spaces etc. Fig. 5 represents the result of parsing resumes.

Fig. 6 represents cosine similarity between job description and content of the resumes computed using equation (4) considering job description as java developer.

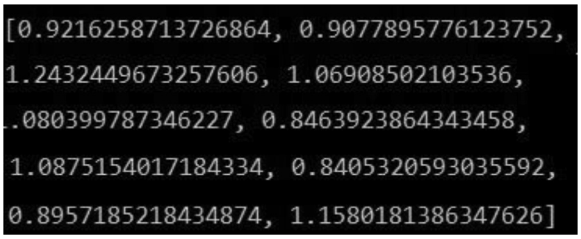
Fig. 7 represents the top-10 resumes predicted by the KNN algorithm using Cosine similarity value.

The System will be able to assess each candidate's resume and assign a relative rating and score. Table 1 and 2 shows the accuracy of parsing and ranking resumes. The system has an average parsing accuracy of 85% and a scoring accuracy 92%.



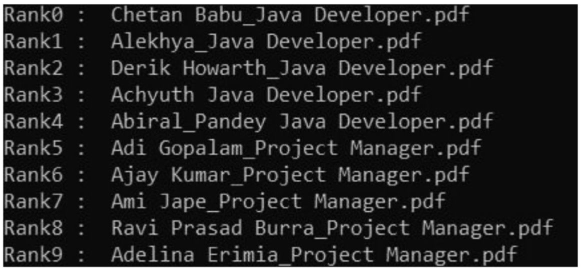
```
****** Let us now Parse These ********
This is PDF 0
This is PDF 1
This is PDF 2
This is PDF 3
This is PDF 4
This is PDF 5
This is PDF 6
This is PDF 7
This is PDF 8
This is PDF 9
Done Parsing.
****** We are Done Parsing ******
Achieved !!!!!!
```

Fig. 5. Resumes after parsing.



```
[0.9216258713726864, 0.9077895776123752,
1.2432449673257606, 1.06908502103536,
.080399787346227, 0.8463923864343458,
1.0875154017184334, 0.8405320593035592,
0.8957185218434874, 1.1580181386347626]
```

Fig. 6. Cosine similarity values computed.



```
Rank0 :  Chetan Babu_Java Developer.pdf
Rank1 :  Alekhya_Java Developer.pdf
Rank2 :  Derik Howarth_Java Developer.pdf
Rank3 :  Achyuth Java Developer.pdf
Rank4 :  Abiral_Pandey Java Developer.pdf
Rank5 :  Adi Gopalam_Project Manager.pdf
Rank6 :  Ajay Kumar_Project Manager.pdf
Rank7 :  Ami Jape_Project Manager.pdf
Rank8 :  Ravi Prasad Burra_Project Manager.pdf
Rank9 :  Adelina Erimia_Project Manager.pdf
```

Fig. 7. Top-10 resumes ranked by KNN algorithm.

Table 1
Parsing accuracy rate.

Number of Resumes	Correctly Parsed	Percentage
10	10	100
15	14	93.3
20	17	85

Table 2
Ranking accuracy rate.

Number of Resumes	Correctly Parsed	Percentage
10	10	100
15	12	80
20	15	75

8. Conclusion

The Resume Screening System replaces ineffective manual screening, ensuring that no candidate is overlooked. The need for efficient and effective resume screening is at the heart of every excellent recruitment strategy. The system will be able to accept or reject a job applicant based on two factors the company's requirements must match the skills listed in the applicant's resume, and the test evaluation will be based on the applicant's skills, ensuring that the resumes uploaded by the applicant are genuine and the applicant is truly knowledgeable about the skills. In our work, NLP methods and KNN algorithm are applied to rate resumes, assisting the firm in hiring the most qualified people.

9. Future scope

The method has a few limitations since it cannot be utilized as a sole criterion for choosing a candidate. In the future, a hybrid recommendation system that combines both Collaborative Filtering Recommendation Systems and Content-Based Filtering Systems can be built using the candidate's academic score and MCQ test results. The implicit compression of the text due to summarizing may have resulted in the loss of essential information while creating a summary using the "genism" package. This summarization technique may be fine-tuned to guarantee minimum information loss. The MCQ exam used to evaluate/screen the candidate is extremely generic, and there is potential to tailor the questions to the position in the future. During the interview, the image recorded during the exam may be used to compare the applicant to encourage fair employment.

Acknowledgement

The authors would like to thank the Management of Dr. Ambedkar Institute of Technology, Bengaluru – 560 056 and B.M.S. College of Engineering, Bengaluru – 560 019 for their support and encouragement for this research work.

References

[1] Pradeep Kumar Roy, Sarabjeet Singh Chowdhary, Rocky Bhatia, A Machine Learning approach for automation of Resume Recommendation system, Procedia Comp. Sci. 167 (2020) 2318–2327.

[2] Jyothis Joseph, Jaimy Sunny, R Raveena, BlessyElzaByju, KC Laya, Resume Analyser: Automated Resume Ranking Software, Int. J. Res. Appl. Sci. Eng. Tech. (IJRASET) 8 (7) (2020) 896–899.

[3] Chirag Daryani, Gurneet Singh Chhabra, Harsh Patel, Indrajeet Kaur Chhabra, Ruchi Patel, An automated resume screening system using natural language processing and similarity, Top. Intellig. Comput. Indust. Des. (ICID) 2 (2) (2020) 99–103.

[4] Aseel B. Kmail, Mohammed Maree, Mohammed Belkhatir, Saadat M. Alhashmi, An automatic online recruitment system based on exploiting multiple semantic resources and concept-relatedness measures, in: 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, 2015, pp. 620–627.

[5] Ramjeet Singh Yadav, A.K. Soni, Saurabh Pal, A study of academic performance evaluation using Fuzzy Logic techniques, in: 2014 International Conference on Computing for Sustainable Global Development (INDIACom), IEEE, 2014, pp. 48–53.

[6] Evanthia Faliagka, Kostas Ramantas, Athanasios Tsakalidis, Giannis Tzimas, Application of machine learning algorithms to an online recruitment system, in: Proc. International Conference on Internet and Web Applications and Services, 2012, pp. 215–220.

[7] Siham Jabri, Azzeddine Dahbi, Taoufiq Gadi, Abdelhak Bassir, Ranking of text documents using TF-IDF weighting and association rules mining, in: 2018 4th international conference on optimization and applications (ICOA), IEEE, 2018, pp. 1–6.

[8] Evanthia Faliagka, Kostas Ramantas, Athanasios Tsakalidis, Giannis Tzimas, Application of machine learning algorithms to an online recruitment system, in: Proc. International Conference on Internet and Web Applications and Services, 2012, pp. 215–220.

[9] Xingsheng Guo, Houssem Jerbi, Michael P. O'Mahony, An analysis framework for content-based job recommendation, 22nd International Conference on Case-Based Reasoning (ICCBR), 2014 29 September-01 October.

[10] V. Senthil Kumaran, A. Sankar, Towards an automated system for intelligent screening of candidates for recruitment using ontology mapping (EXPERT), Int. J. Metadata Semant. Ontol. 8 (1) (2013) 56–64.

[11] Jack L. Howard, Gerald R. Ferris, The employment interview context: Social and situational influences on interviewer decisions 1, J. Appl. Soc. Psychol. 26 (2) (1996) 112–136.

[12] Pradeep Kumar Roy, Jyoti Prakash Singh, Amitava Nag, Finding active expert users for question routing in community question answering sites, in: International Conference on Machine Learning and Data Mining in Pattern Recognition, Cham, Springer, 2018, pp. 440–451.

[13] Kangning Wei, Jinghua Huang, Shaohong Fu, A survey of e-commerce recommender systems, in: 2007 international conference on service systems and service management, IEEE, 2007, pp. 1–5.

[14] Shaha T. Al-Otaibi, Mourad Ykhlef, A survey of job recommender systems, Int. J. Phys. Sci. 7 (29) (2012) 5127–5142.

[15] Jochen Malinowski, Tobias Keim, Oliver Wendt, Tim Weitzel, Matching people and jobs: A bilateral recommendation approach, in: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06), 6, IEEE, 2006 137c-137c.

[16] Adem Golec, Esra Kahya, A fuzzy model for competency-based employee evaluation and selection, Comput. Ind. Eng. 52 (1) (2007) 143–161.

[17] Grigori Sidorov, Alexander Gelbukh, Helena Gomez-Adorno, David Pinto, SoftSimilarity and Soft Cosine Measure: Similarity of Features in Vector Space Model, Computacion y Sistemas 18 (3) (2014) 491–504.

[18] Paul Resnick, Hal R. Varian, Recommender systems, Commun. ACM 40 (3) (1997) 56–58.