PAKISTAN JOURNAL OF EMERGING SCIENCE AND TECHNOLOGIES (PJEST)

June, 2021, 1(1) ISSN 2788-8630 (On-Line)

AUTOMATIC CV RANKING USING DOCUMENT VECTOR AND WORD EMBEDDING

Ansa Abdul Noor[1], Maheen Bakhtyar[1], Bilal Ahmed Chandio[1], Rehana Gull[2], Junaid Baber[1]
[1] Department of CS and IT, University of Balochistan

[2]Department of CS, Sardar Bahadur Khan Women University, Quetta

*Correspondence: ansanoor8@gmail.com

ARTICLE INFORMATION

Citation: Ansa Abdul Noor, Maheen Bakhtyar, Bilal Ahmed Chandio, Rehana Gull, & Junaid Baber. (2021). AUTOMATIC CV RANKING USING DOCUMENT VECTOR AND WORD EMBEDDING (Version 2). Pakistan Journal of Emerging Science and Technologies, 2(1), 9. http://doi.org/10.5281/zenodo.5089434

Received: 09th June-2021

Revised and Accepted:

01-July-2021

Published On-Line

10-July-2021

***Corresponding Author:**

Ansa Noor:

ansanoor8@gmail.com

Original Research Article

ABSTRACT

This research is based on the practical facts related to the human resource department of any organization for the recruitment of personnel. As it is a challenging and crucial aspect for any organization to select the right talent for the right place. This paper helps in expertise finding in the different fields of Computer Science. Employers receive a bunch of resumes upon job openings. And the candidates are also interested are in sifting the best among the applicants. Screening the best candidate among the pool of resumes is a laborious task. This paper proposes an informational retrieval-based resume ranking scheme for screening and ranking the candidate's resumes. The primary purpose of this research study is to exploit the class NLP techniques to perform the information retrieval task for resume ranking based on job description similarity. In this proposed methodology, we compared document vectors with word embedding. Experiments show that word embedding method is more effective than the document vector.

Keywords: Word Embedding, Informational Retrieval, TFIDF.

Introduction:

Information retrieval (IR) models are composed of an indexed and a scoring or ranking function. An IR system's main purpose is to retrieve relevant documents or web pages in response to customer requests. The scoring mechanism is employed during the retrieval to order the recovered documents according to their relevance to the user query. The standard technique is used to overcome train word or paragraph embeddings on a corpus or use pre-trained embeddings. WE stand for word embeddings, which are distributed representations of phrases derived from a neural network model. In recent years, these continuous representations have been employed in a variety of natural language processing applications. This approach's effectiveness for classification and other NLP tasks is demonstrated by the fact that semantically similar terms are grouped near to

each other in the representation space. Transferring the success of word embeddings to the ad-hoc Information Retrieval (IR) problem, on the other hand, is a widely researched topic right now.

Hiring the right talent is a big challenge for all companies. A business that requires full effort developing and facing high rates of attrition. Today's competition is at a high peak.it has become a challenging task for the job provider as well as a job seeker to get the one as the expert. For job providers, the challenging task is the extraction of required data. In our case, we have a set of CVs, but job descriptions are unknown, therefore we need to come up with a solution based on unsupervised learning. As a result, word embeddings appear to be a suitable place to start our research. In a high-dimensional space, word embedding computes semantically meaningful vector representations of words. Recent word embedding approaches outperform older methods of representing a word as a vector, such as the bag-of-words representation or latent semantic indexing. Advanced word embedding techniques and the use of multidimensional transformation matrices to flexibly capture different semantics of a single word, leading to better representations for part-of-speech tagging tasks, have recently been actively studied since such word embedding techniques have shown their advantages in numerous tasks in natural language processing and information retrieval.

In the case of job seekers the challenge to get to the place where their talent could be utilized. The traditional approach of applying to the job is to look for the job and then apply by sending a resume or curriculum vitae (CV) to the place. As manually it is difficult. Large enterprise and head-searchers receive every day thousands of curriculum vitae CVs/resumes from applicants. Consequently, reviewing a large number of CVs/resumes manually is time-wasting. And also, with the growing need for research and development, many of the organizations are hunting for good researchable talent. Many Ph.D. scholars who complete their degrees mention their specialization by giving a short detail of a particular topic on their CVs. This can attract hiring companies. Efficient Automatic CV shortlisting will not only make the lives of employers easy but also make the recruitment process fast and efficient[1]. Reference [2], introduce high-level ontologies with domain-specific knowledge is introduced to meet the requirements of the job market.

Furthermore, in this paper, a process is introduced which is used to update the existing ontology with the concept of formal analysis. For job recruitment matching of the semantic base, ontology is performed. Constructing quality ontology is crucial. Resume Resource Description Framework (RDF) [3], and Description of Career (DOAC) are frameworks based on ontologies. These studies are designed with the help of ontologies. Resume RDF is used for the description of the semantics of resumes and includes information such as skill, the experience of work, qualification, etc. Description of a career is another study that is the same as the Resume RDF. DOAC is the vocabulary used to describe the resumes if we talk about the performance of these two studies. The rate of generation of Resume RDF of queries as result is higher than DOAC. The main interest of this study was to find resumes that are appropriate matches for a required job description. Carrying out several experiments on CV/resume dataset as dataset consists of millions of resumes. In this research, candidates are short-listed by using different Natural Language Processing (NLP) techniques to find expertise in different fields of computer science. While word embeddings have been proven to improve in different NLP tasks, they have not yet been designed to improve text retrieval in software engineering to our knowledge.

Literature Review:

Automatic CV Shortlisting for Expert Finding is the highly recommended tool for the recruitment process as it can speed up the process of screening as compared to manual screening this research describes a ranking system that shortlists the candidates for a particular job. The experiments have been carried out on 169 CVs/resumes of the dataset against 25 to 30 job categories. The calculation of text similarity has been an essential approach of data analysis that may be used in a variety of NLP applications such as information retrieval and sentiment analysis. For retrieving our required results two NLP techniques like Term Frequency–Inverse Document Frequency (TFIDF), and word embedding have been used. The Jaccard coefficient, cosine similarity of TFIDF vectors, and cosine similarity of log TFIDF vectors were used in comparative research to assess the semantic similarity of academic articles and patents [4]. All of these approaches are corpus-based, and a case study was conducted for further research [5], which looked into the many uses of semantic similarity in the field of Social Network Analysis. Additionally, for ranking of CVs/resumes Google word embeddings are used as these are already trained. Much other research has been presented for expert findings. Each of the techniques has a different procedure for hunting the right talent for a particular post. Techniques such as collaborative filtering to the recruitment of candidate's job matching [6]. X.YI.j Allan and W.B craft expressed a method that uses a model of relevance to handle the vocabulary which is divided between a description of jobs and CV/resumes.

A semantic similarity computation method [7] that compared and analyzed multiple words using a huge corpus. The Word2Vec model was used to compute semantic information from the corpus, and the Li similarity measure was used to compute similarity. Automata collaboration in the system "CASPER" in this system the profiles of users are achieved from a user [8]. Such as data revisit, data read time, and data retrieving. These are factors that are viewed as a measure of relevance among resumes. This system recommends the job into these steps: first, this system searches a set of users who are related to the job. Second the user-related liked jobs are recommended to a large number of candidates. In this system, a strategy that is used is known as cluster-based collaboration filtering. It allows the jobseeker to find a job through a query that contains the same fields such as candidate location, candidate salary, and skills. Reference [9], has worked on a recommended system which is using a hybrid approach. This system is a combination of methods, such as content-based filtering and collaborative filtering. This system attempts to control the rating of data sparseness y grasping a model which is integrated, includes the synthetic resumes as a dataset.

About [10], job recommender system, the user data is asked to input user's required data into a form that is a web-based interface. The data which is collected consists of demographic data, educational data, intermediate and university exams, and post-graduate degree, experience the job, skills in languages, and skills in IT, and users are also asked to upload their CVs/resumes. Furthermore, a statistical model is used for other aspects such as talent. This model requires to be trained before use. To train the recommender model the user's searched outcomes are used as training data. Therefore, training time will be long enough for each user. Reference [11], introduced a recruiting system "PROSPECT" to help in shortlisting candidates. A resume miner is used for extracting information from CVs/resumes. Then the algorithms such as BM25, Okapi, and KL were used to solve the same problem. This system is presented by [12]. If we talk about dictionaries, we know that every day new terms have appeared, so for that purpose, the dictionaries must be updated as these are going to be used in tagging the entries. The learning model which is

adopted used in this system used to gain two targets: at first, for enhancing the information extraction semantic data should be used, and second, new terms should be discovered. A system that has a hybrid recommender, which has two types of relationships one is content-based and the second is interaction-based [13]. In the first relation, the sequence of relations is as follows job-to-job, job-to-job seeker, and job seeker-to-job seeker are identified concerning their similarity. For calculating similarities two approaches are used, for example, the structured data such as for age and gender the value-weighted sum will be produced and other is unstructured such as the similarity between job and CVs/resumes and latent semantic analysis is also used in this hybrid recommender. In E-Gen, we evaluated and classified unstructured job descriptions to improve their relevance[1].

Reference [14], In contrast to the above-discussed works, said that for facilitating easy communication, they recommend using a common language. Because it would create potential automation in the phases of the recruiting process. Not only will this keyword from ruled vocabulary be used, which aid in combining the background knowledge to the domain of industry. Furthermore, these approaches are used to incorporate with ontology to determine that at which degree the match between position and applicant is performed. This is also very critical because it also occurs when the ontologies are reused. Reference [15], has discussed these issues and suggests that further manual work is required. Reference [16], proposed jointly working on Turkey's largest online recruitment website Kariyer.net. In this project, free-structured resumes are converted into label base Resume Parser (ORP) works on a large number of resumes written in Turkish and English. Semantic web and ontology are combined to form a web Ontology. A machine learning algorithm is employed to rate the candidates and conduct semantic matching techniques. The goal of this study is to assess job candidates in e-recruitment [17]. There is some detail of related works: Resume RDF, a resume ontology developed by [3], describes information from a resume by its classes and properties. A method that results in the semantic orientation of words in corpus corpora and their connection with other words [18]. An approach for retrieving information from resumes that uses the notion of sharing data to find new types of data [19]. Recent word embedding methods such as word2vec [20] and GloVe [21] have two notable advantages in terms of high-level semantics compared to traditional methods of representing a word as a vector, such as the bag-of-words representation [22] or latent semantic indexing: meaningful nearest neighbours and linear substructures [21].

In terms of the first, these approaches successfully capture semantically related words in a vector space as the n nearest neighbors of a certain word. When it comes to linear substructures, the vector created by removing two words in a vector space frequently gives semantics that contradicts the word. Since word embedding approaches have demonstrated their utility in a variety of natural language processing and information retrieval applications, sophisticated word embedding techniques have lately received a lot of attention. Word embeddings are vector representations of a word produced through the use of a large corpus to train a neural network. It has been frequently used to classify texts based on semantic similarity. One of the most popular types of word embeddings is Word2vec. The word2vec algorithm takes a text corpus as input and outputs word vectors, which can then be used to train any other word to get its vector value. Based on the distributional hypothesis, the Word2vec model employs a continuous skip-gram model [23]. Gensim [24], an open-source toolkit, includes several pre-trained word2vec models based on various datasets such as GloVe, google news, ConceptNet, Wikipedia, and Twitter [24]. To construct feature vectors for the dataset in this experiment, a pre-trained word embeddings model

called ConceptNet Number batch [25] was utilized as a word2vec model. There are several distributional semantic models (DSMs) that have been suggested. Each word in a DSM is represented as a d-dimensional vector of real values, with comparable vector representations for words that appear in similar contexts. Count-based models make up the majority of classic DSMs. Some new neural network models have recently been presented [20]. Deep neural networks are used in these techniques to learn from the context of the corpus and produce low-dimensional word vector representations, a process known as "word embedding." For different information retrieval tasks, word embedding models have been shown to perform much better than standard count-based models [26].

Many information retrieval tasks exist in software engineering, such as duplicate issue detection in open source projects and tag recommendation in Q&A websites [27]. The majority of them expand classic information retrieval techniques like cosine similarity and the TFIDF approach to obtain high performance on these tasks. Word embedding algorithms have recently been applied to numerous information retrieval problems in software engineering, with promising results [26]. Traditional information retrieval approaches such as the TFIDF method focus more on the relationship of various documents in the entire corpus, whereas word embedding techniques focus more on the relationship of words considering the context in which they appear.

Methodology

Our proposed methodology for resume ranking is based on two famous approaches in information retrieval. The classical and widely used approach is document vector in which document is represented by some predefined vocabulary. Figure 1 shows the document representation using the first approach. The vector is usually very sparse as the length of the vocabulary is kept very high. The document vector usually used a weighting schema known as TFIDF (Term frequency-inverse document frequency**)** which reduces the weight of very frequent words in corpus and gives high weights to the least frequent words. The main limitation of this approach is that it doesn't account for the context of the text. Two different documents with the same words are considered similar. To overcome the above-mentioned limitation, word2vec or word embedding-based approaches are used. In word embedding, the vector is obtained against every word in the predefined vocabulary and those vectors are concatenated to represent the document.

In our proposed scheme initially, the resume /CVs corpus is preprocessed and cleaned. The processed text is now ready to be fed in the TFIDF mechanism and word embedding. The term-frequency and inverse document frequency is used to create the feature vector R^n which allows giving more weight to the key terms. The job description object is also processed in the same manner as the resume corpus which is further used as a query in our information retrieval system. Eventually, the cosine similarity is computed to estimate the proximity of resumes with job description objects. The ranked resumes are finally sorted and filtered for a certain threshold. Our second method is based on Google word embedding which uses the state-of-the-art Word2vec. The study is aimed at empirically evaluating both NLP schemes to create a generic model for resume/CVs ranking. To assess the accuracy of our model we have intended to estimate the precision and recall using a human-annotated rank list as ground truth. Word2Vec creates a vector space with each unique word in the corpus, generally with several hundred dimensions, such that words with similar contexts in the corpus are close to one another in the space. This may be

achieved in two ways: starting with a single word and predicting its context, or starting with the context and predicting a word.

Cosine Similarity using TFIDF Vectors: A vectorized TFIDF model was used to transform the preprocessed documents into TFIDF vectors. The resulting vectors were a sparse matrix with TFIDF weights for each word in each document, with a size of [number of documents * number of features(unique words)]. These TFIDF weights from the matrix are now used as a feature for each document, and cosine similarity is used to compute document similarities.

Cosine Similarity using Word2Vec Vectors: The Google model was used to load the pre-trained word2vec model in this technique. The vector values or word embeddings of each word in all the preprocessed documents were computed using this word2vec model. The average of all the word vectors in a text was calculated, and the resulting vector was used as the document's feature word2vec vector. The cosine similarity inside the documents was computed using these word2vec vectors as feature vectors.

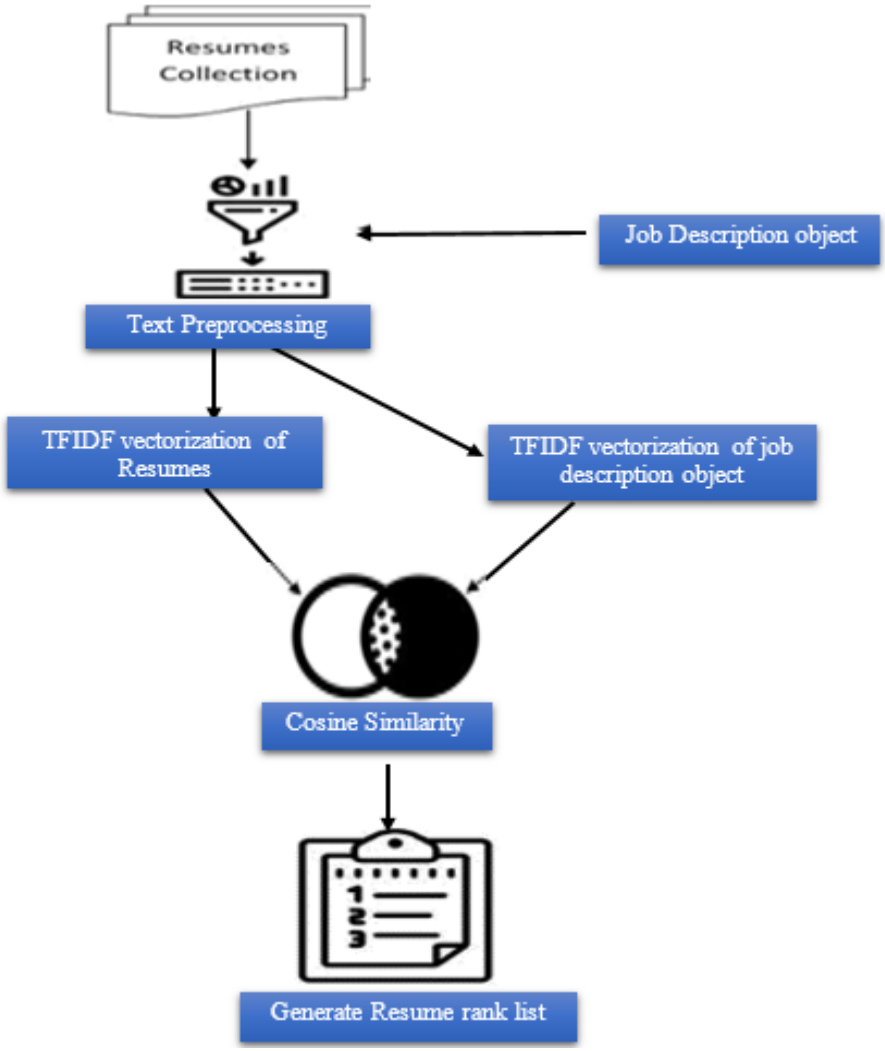

Fig. 1: Proposed methodology to compute document vector for the similarity.

In Fig. 1. The CVs corpus is preprocessed and cleaned. The processed text is now ready to be fed in the TFIDF mechanism and word embedding. The term-frequency and inverse document frequency is used to create the feature vector which allows giving more weight to the key terms. The job description object is processed in the same manner as the resume corpus. The cosine similarity is computed. The ranked resumes are finally sorted and filtered for a certain threshold.

Experiment:

Google word embeddings have been used to rank CVs; these word embeddings have already been trained by Google. Therefore, these embeddings were used to fit the model. We essentially used word embeddings to match our document. The average of each vector must be calculated separately to accomplish otherwise it will generate a three-dimensional vector. For this purpose, a word embedding function is created in which a specific word is selected, appended to embeddings, and the mean of that word is calculated. If no embedding for a given word exists, a random number is generated for that word. This random number will be trained later. After that, simply receiving the embeddings, stop words are removed and text cleaning will be used. Following that, the cosine similarity measure of the resulting vector is calculated. The mean word embedding of a query is also computed using Google word embedding concerning this cosine similarity.

A rank is generated after measuring the cosine similarity of a query with word embedding. Then according to this rank, the job descriptions are matched with the candidate's CV. A job description, for instance, is Hadoop. As a result, we've set some true positives that correspond to the job description for Hadoop. Hadoop itself, as well as some other related job titles such as Database Developer or Data Science Developer, will be tested for true positive. The total number of CVs in the dataset is 169. It displays the ranking of all CVs. Following that, ground truth is used to calculate the precision and recall of the rank list. In a nutshell, the entire operation comprises of the phases listed as: first and foremost, Google word embeddings were utilized.

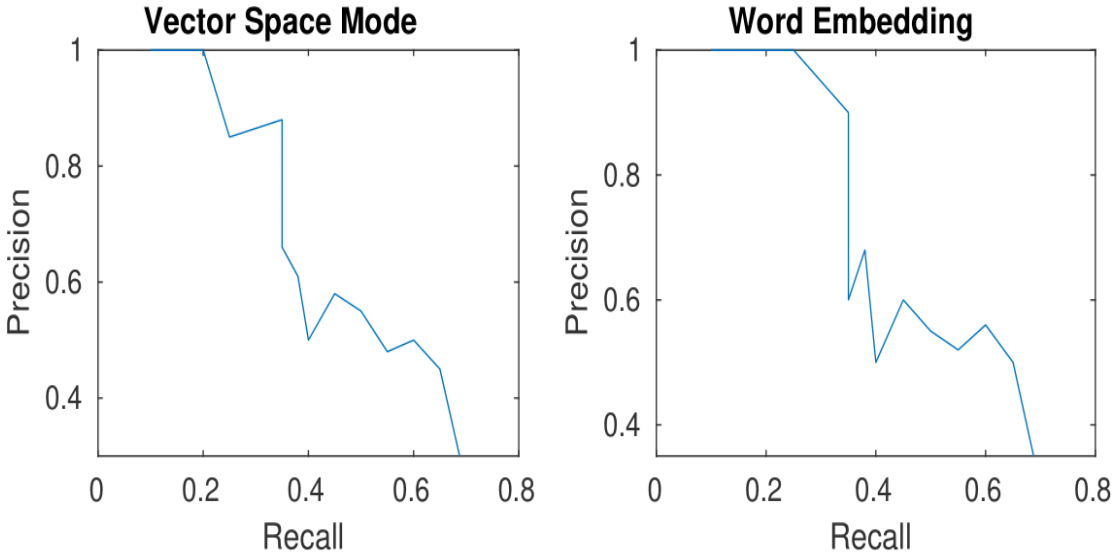

Fig 2: Precision and Recall of TFIDF and word embedding approaches

Figure 2 shows the precision and recall curves of document vector and word embedding. It can be seen that word embedding curve covers more area than the document vector. Which also that The word embedding-based approach is more effective than the document vector-based approach

The average word embedding of the entire document was then taken. Following that, cosine similarity was determined. As a result, a ranked list is created, and precision & recall are measured using this ranked list. There is a total of 169 CVs, with 25 to 30 job categories, such as Data Science, Hadoop, and Database Developer. Any of the job description categories can be queried to obtain a required ranked list of CVs.

Conclusion

This technology can speed up the recruiting process due to a combination of two aspects. First, rank the candidates according to how well they meet the job description. Furthermore, the usage of filters based on various information gathered from resumes helps screeners to look at fewer resumes to shortlist a set number of candidates. Second, showing snippets of resumes based on their match to job requirements and information extracted from resumes allows a screener to shortlist or reject candidates much more quickly than if they had to scan the entire resume. In this research, two famous approaches of information retrieval are evaluated for CV ranking. The word embedding-based approach is more effective than the document vector-based approach.

References

[1] R. Kessler, J. M. Torres-Moreno, and M. El-Bèze, "E-gen: Automatic job offer processing system for human resources," in MICAI 2007: Advances in Artificial Intelligence, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. https://doi.org/10.1007/978-3-540-76631-5_94

[2] D. Looser, H. Ma, and K.-D. Schewe, "Using formal concept analysis for ontology maintenance in human resource recruitment," in Proceedings of the Ninth Asia-Pacific Conference on Conceptual Modelling - Volume 143, 2013.

[3] Bojārs, U., & Breslin, J. G, . ResumeRDF: Expressing skill information on the Semantic Web. In 1 st International ExpertFinder Workshop.

[4] Shibata, N., Kajikawa, Y., & Sakata, "How to measure the semantic similarities between scientific papers and patents to discover uncommercialized research fronts: A case study of solar cells". In PICMET 2010 technology management for global economic growth . IEEE.

[5] Zhang, S., Zheng, X., & Hu, C. (2015, October). A survey of semantic similarity and its application to social network analysis. In 2015 IEEE International Conference on Big Data (Big Data) (pp. 2362-2367). IEEE. https://doi.org/10.1109/BigData.2015.7364028

[6] S. Laumer and A. Eckhardt, "Help to find the needle in a haystack: Integrating recommender systems in an IT supported staff recruitment system," in Proceedings of the special interest group on management information system's 47th annual conference on Computer personnel research - SIGMIS-CPR '09, 2009. https://doi.org/10.1145/1542130.1542133

[7] R. Rafter and B. Smyth, "Item selection strategies for collaborative filtering," in The 18th International Joint Conference on Artificial Intelligence (IJCAI 03), Acapulco, Mexico, 2003.

[8] F. Färber, T. Weitzel, and T. Keim, "An automated recommendation approach to selection in personnel recruitment," 2003.

[9] Sitikhu, P., Pahi, K., Thapa, P., & Shakya, S. (2019, November). A comparison of semantic similarity methods for maximum human interpretability. In 2019 artificial intelligence for transforming business and society (AITB) (Vol. 1, pp. 1-4). IEEE.https://doi.org/ 10.1109/AITB48515.2019.8947433

[10] J. Malinowski, T. Keim, O. Wendt, and T. Weitzel, "Matching people and jobs: A bilateral recommendation approach," in Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06), 2006. https://doi.org/ 10.1109/HICSS.2006.266
[11] A. Singh, C. Rose, K. Visweswariah, V. Chenthamarakshan, and N. Kambhatla, "PROSPECT: A system for screening candidates for recruitment," in Proceedings of the 19th ACM international conference on Information and knowledge management - CIKM '10, 2010. https://doi.org/ 10.1145/1871437.1871523.
[12] T. Gonzalez et al., "Adaptive employee profile classification for resource planning tool," in 2012 Annual SRII Global Conference, 2012. https://doi.org/ 10.1109/SRII.2012.67
[13] Y. Lu, S. El Helou, and D. Gillet, "A recommender system for job seeking and recruiting website," in Proceedings of the 22nd International Conference on World Wide Web - WWW '13 Companion, 2013. https://doi.org/10.1145/2487788.2488092
[14] C. Bizer, R. Heese, M. Mochol, R. Oldakowski, R. Tolksdorf, and R. Eckstein, "The impact of semantic web technologies on job recruitment processes," in Wirtschaftsinformatik 2005, Heidelberg: Physica-Verlag HD, 2005, pp. 1367–1381. https://doi.org/10.1007/3-7908-1624-8_72
[15] M. Mochol, A. Jentzsch, M. Mochol, and A. Jentzsch, "Applying an analytic method for matching approach selection ome Euzenat To cite this version : Applying an Analytic Method for Matching Approach Selection." 2013.
[16] D. Celik et al., "Towards an information extraction system based on ontology to match resumes and jobs," in 2013 IEEE 37th Annual Computer Software and Applications Conference Workshops, 2013. https://doi.org/10.1109/COMPSACW.2013.60
[17] "An integrated e-recruitment system for CV ranking based on ahp," in Proceedings of the 7th International Conference on Web Information Systems and Technologies, 2011. https://doi.org/ 10.5220/0003337901470150.
[18] P. Turney and M. L. Littman, "Unsupervised learning of semantic orientation from a hundred-billion-word corpus," National Research Council of Canada, 2002.
[19] U. Marjit, "Discovering Resume Information using linked data," Int. j. web semant. technol., vol. 3, no. 2, pp. 51–62, 2012. https://doi.org/ 10.5121/ijwest.2012.3204
[20] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J." Distributed representations of words and phrases and their compositionality",2013.
[21] Pennington, J., Socher, R., & Manning, C. D,. Glove "Global vectors for word representation" In Proceedings of the conference on empirical methods in natural language processing, 2014.
[22] Manning, C. D., Raghavan, P., & Schütze, H," Xml retrieval. In Introduction to Information Retrieval" Cambridze University Press 2008. https://doi.org/ 10.1109/INFRKM.2016.7806343
[23] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).
[24] Rehurek, R., & Sojka, P. (2010). Software framework for topic modelling with large corpora. In In Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks.
[25] Speer, R., Chin, J., & Havasi, C. (2017, February). Conceptnet 5.5: An open multilingual graph of general knowledge. In Thirty-first AAAI conference on artificial intelligence.
[26] Ye, X., Shen, H., Ma, X., Bunescu, R., & Liu, C. (2016, May). From word embeddings to document similarities for improved information retrieval in software engineering. In Proceedings of the 38th international conference on software engineering (pp. 404-415). https://doi.org/10.1145/2884781.2884862
[27] Xia, X., Lo, D., Wang, X., & Zhou, B. (2013, May). Tag recommendation in software information sites. In 2013 10th Working Conference on Mining Software Repositories (MSR) (pp. 287-296). IEEE. https://doi.org/10.1109/MSR.2013.6624040