International Journal of Electrical and Computer Engineering (IJECE)
Vol. 13, No. 6, December 2023, pp. 6964~6971
ISSN: 2088-8708, DOI: 10.11591/ijece.v13i6.pp6964-6971

Building a recommendation system based on the job offers extracted from the web and the skills of job seekers

Hanae Mgarbi, Mohamed Yassin Chkouri, Abderrahim Tahiri
Laboratory of Information System and Software Engineering, National School of Applied Sciences Tetouan,
Abdelmalek Essaadi University, Tetouan, Morocco

Article Info

Article history:

Received Dec 31, 2022
Revised Apr 9, 2023
Accepted Apr 14, 2023

Keywords:

Cross distances
E-portfolio
Job offers
Natural language processing
Recruitment
Skills analysis

ABSTRACT

Recruitment, or job search, is increasingly used throughout the world by a large population of users through various channels, such as websites, platforms, and professional networks. Given the large volume of information related to job descriptions and user profiles, it is complicated to appropriately match a user's profile with a job description, and vice versa. The job search approach has drawbacks since the job seeker needs to search a job offers in each recruitment platform, manage their accounts, and apply for the relevant job vacancies, which wastes considerable time and effort. The contribution of this research work is the construction of a recommendation system based on the job offers extracted from the web and on the e-portfolios of job seekers. After the extraction of the data, natural language processing is applied to structured data and is ready for filtering and analysis. The proposed system is a content-based system, it measures the degree of correspondence between the attributes of the e-portfolio with those of each job offer of the same list of competence specialties using the Euclidean distance, the result is classified with a decreasing way to display the most relevant to the least relevant job offers.

Corresponding Author:

Hanae Mgarbi
Laboratory of Information System and Software Engineering, National School of Applied Sciences Tetouan, Abdelmalek Essaadi University
Tetouan, Morocco
Email: hanae.mgarbi@etu.uae.ac.ma

1. INTRODUCTION

In recent years, job offers published on the Web have increased considerably with a wide variety of platforms and professional networks offering these offers. For the candidate, it becomes more and more difficult to find suitable job offers relevant to his profile. Indeed, candidates must register and create an account on each recruitment platform. This implies a considerable waste of time for candidates to manage their accounts, fill in all the information concerning the curriculum vitae and consult the job offers published on each platform. A recommendation system is a specific form of information filtering, and an application intended to offer a user item likely to interest him according to his profile. Recommendation systems are used in particular on online sales websites and are based on the comparison between the user and the items that are represented by elements of information (images, documents, and web pages). The conventional information retrieval or recommendation methods using vector modeling directly deduce the relevance based on the similarity measure between the vector representing the user and that representing the item. Following a study carried out on job offer recommendation systems (the word processor project [1], and the e-recruitment system analysis and design (SAJ) [2]). Most job recommender systems are company-oriented, not candidate-oriented, and highly dependent on the platform offering the jobs, i.e., each recommender system is linked with a platform. This increases the dependence between the platform and the recommendation system.

The contribution of this work is to propose a system for recommending job offers based on the e-portfolios [3], [4] of candidates according to their specialty of skills. Our research work is followed by a process that begins with the extraction of job offers from the web, according to a previously defined data model. Among the extracted data, there are data of textual formats that require natural language processing like text tokenization [5], PostTagging [6], and named entity recognition (NER) [7], to have well-structured data ready for filtering and analysis. The analysis was made on the processed data using knowledge bases to extract the list of skills specialties ordered according to the degree of expertise requested. In this step, the degree of correspondence between job offers and e-portfolios must be measured using the "Cross Distances" operator which returns the similarity calculation between each e-portfolio of the set of e-portfolios and each job offer of the set of job offers. As a result, the job seeker obtains the list of job offers ranked according to the degree of correspondence in a descending manner.

This document is organized as follows: section 2 provides an overview of data extraction and the principles of natural language processing levels and introduces the concept of the recommender system. We discuss the related work in section 3. In section 4, we explain the steps of our contribution to realizing a job offers recommender system based on the e-portfolios of candidates according to their skills. The conclusions close the article in section 5.

2. THEORETICAL BACKGROUND

2.1. Data extraction

Data published on the web can be used for several purposes, and to achieve a goal such as being posted or structuring textual data, we can use the addressing element in the document tree (the Xpath) [8] that provides a powerful syntax for addressing specific elements of an extensible markup language (XML) document (or hypertext markup language (HTML) web pages) in a simple way. Web data that is in textual and semi-structured formats within web pages. They can be represented with ordered labeled trees, where the labels represent the hypertext markup language tags, and the tree represents the different levels of nesting of the elements that make up the web page. This representation is called document object model (DOM). And there is another method to do the data extraction is the procedure web wrapper [8] that extracts unstructured or semi-structured data and transforms it into a structured format, unifying it for further use fully automatic or semi-automatic.

2.2. Data preprocessing

Natural language processing (NLP) [9], [10] is defined as a branch of artificial intelligence that concerns the interaction between computers and humans using natural language, its applications process texts at different linguistic levels, they are 4 phases involved: The first one, morphological processing is the field of linguistics that analyzes the structure of words and performs morphological decomposition by analyzing the structure of the sentence. Tokenization is a necessary and non-trivial step in natural language processing. It is the task of cutting a string into identifiable linguistic units which constitute linguistic data. The second one, Syntactic analysis associates with each word different types of information such as its grammatical category, and its corresponding lemma to eliminate the ambiguity of several categories for a single word. Like the lemmatization that tries to reduce the word to its proper lemma which can be found in the dictionary. The third one, Semantic analysis studies the meaning of linguistic expressions for the disambiguation of the meaning of the word. The meaning of the remaining ambiguous words from the syntactic step is resolved by considering the interactions between the meanings of the individual words in the sentence. The last one, pragmatic analysis interprets the results of semantic processing from the point of view of a specific context. This processing makes it possible to remove the ambiguity of the sentences, which did not do so during the syntactic and semantic analysis phases.

2.3. Recommendation system

Recommender systems [11] are found in many current applications that expose the user to a huge collection of items. Such systems typically provide the user with a list of recommended items they might like or predict how much they might like each item. These systems help users choose appropriate items and make it easier to find their favorite items in the collection. The recommendation system [12] is based on the comparison between the user and the items, which are the information elements. Conventional information retrieval or recommendation methods using vector modeling directly deduce the relevance of the similarity measure between the vector representing the user and that representing the item.

The user of a recommendation system for more than exact anticipation of his tastes. He may be interested in discovering new objects, quickly exploring various objects, preserving their privacy, in the rapid responses of the system, and many other properties of interaction with the recommendation engine. It is therefore necessary to identify all the properties that can influence the success of a recommender system in the context of a specific application.

3. RELATED WORKS

In the literature there are only systems dedicated to the company to assign the able e-portfolios to the job offer proposed, there are no recommendation systems that are dedicated to the candidate, and which seek the job offers published on the internet. In this section, we conducted a study on the recommendation system based on job postings. Three research works were selected for this study.

The word processor project [1] for the classification of job offers uses the explicit-rules and machine learning. This project consists first of all of the text preprocessing such as tokenization, lower case reduction, HTML substitution of special characters, removing of stop words, elimination of numbers, and the recovery of the word stemming. After preprocessing, the project uses the rule-based approach, which can be defined by the user based on taxonomies modeling terminology in certain domains. A matrix was built with the titles of the online job offers and the columns represent the characteristics (i.e., the word counts and the different words stemming) two machine learning classifiers were used to perform the text classifications. The technique developed is based only on job titles. It offers a set of documents and is based on two different stages: extraction and classification of characteristics. The objective of the feature extraction task is to derive for each job category a particular data structure, called weighted word pairs (WWP) [13]–[15] and containing the most relevant pairs of elements (by exploiting an automatic processing of classical NLP as well as the probabilistic dependencies characterizing the titles of the job offers. On the other hand, the classification procedure is based on the matching between the terms taken from the job title job and the set of pairs linked to the different of Italian National Institute of Statistics (ISTAT) categories [16]. For each category, the number of matches is determined and each move is weighted by the probability dependence of the related pairs in the WWP structure. The category with the highest score is finally selected as the winner and selected for classification.

The proposed e-recruitment system called (SAJ project) [2] presents the method of extracting data from the job offer description and from the candidate profile to be able to match them; they used the linked open data system, ontology job posting description domain, and domain specific dictionaries for data mining. SAJ enriches and builds the context between the extracted data to minimize the loss of information in the extraction process. Unstructured job description text extracted from any document format, such as Microsoft Word or PDF. Plain text is extracted using Apache Tika. Then the text is segmented into predefined categories using a self-generated dictionary. Automatic NLP and the dictionary help in data identification. Data entities are passed to two parallel processes, context building and entity enrichment. The result of these two processes is integrated and stored in the knowledge base. On the other hand, SAJ not only extracts job description entities but also enriches them, unlike existing e-recruiting systems. The entities extracted by SAJ and their connections can facilitate the search and recovery, scoring, and ranking of candidates against the job description. SAJ combines various processes to extract and enrich contextual information from the job description in online recruiting.

The development of a set of predictive systems [17] called Work4 Oracle can estimate the audience (number of clicks) that a job offer would get posted on Facebook, LinkedIn, or Twitter. These systems combine both recommender system techniques and machine learning methods. The results made it possible to quantify the factors that influence the audience (and therefore the attractiveness) of job offers posted on social networks. The authors used user data from Facebook (job), LinkedIn (position), and job posting (title). The reconciliation between the job postings 'title' to Facebook's 'job' and LinkedIn's 'position' is done using O*NET taxonomy [18] (online has detailed descriptions of the world of work for use by job seekers, workforce development and HR professionals, students, developers, and researchers) O*NET-SOC. The latter extracts the O*NET-SOC vectors and gives importance to the most significant data, then the vectors pass to a similarity function to help the system to predict the audience of job offers published on the networks. LinkedIn, Facebook, and Twitter thanks to Work4. Work4 collects data concerning the fields to which the O*NET-SOC taxonomy applies while respecting the privacy of users. Somvanshi *et al.* [19] used support vector machines (SVM) to train the system to better select the audience for job offers.

4. JOB OFFERS A RECOMMENDATION SYSTEM

In this section, we will present our research work starting with the extraction of data concerning job offers from websites and recruitment platforms based on our data model defined based on a set of criteria. Subsequently, the extracted data will be subject to natural language processing to prepare our dataset. Then, we will classify job offers and e-portfolios according to the skill specialty. Our recommendation system receives in the input the e-portfolios of candidates and the job offers, and it gives in the output the list of job offers to correspond to the e-portfolios, as shown in Figure 1.

4.1. Extraction of job offers data

The first step of our research work, started by extracting job offers in Morocco. We used recruitment sites like Rekrute, Emploi.ma, and professional networks like LinkedIn, to extract job offers directly from the site. We extracted all the information about the job posting such as the job title; the field of activity; the mission; level of education; years of experience; the city; the type of contract that makes up the data model. We have collected more than 4,000 job offers in a structured format in an Excel file using data miner [20].

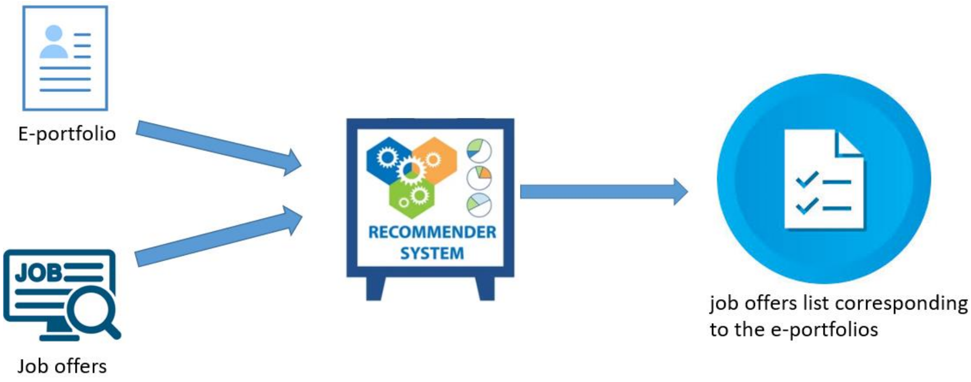

Figure 1. The input and the output of the recommender system proposed

4.2. Data pre-processing of job offers

To obtain high-quality data, the extracted job offers are subjected to automatic NLP [21]. We started by deleting rows with missing values so as not to block further processing. Then, we passed to the conversion of the nonnumeric values concerning the years of experience and the level of studies to have this data in the form of an integer. Then, we used the binarization of categorical data which concerns the qualitative data of our case such as the types of Moroccan contracts which are: The open-ended contract (CDI), a fixed-term contract (CDD), Freelance. In addition, the languages that can be especially in the Moroccan job market that we deal with: are Arabic, French, English, Spanish, Chinese, German, and Portuguese. These last three languages are requested more by call centers. We also have the city for our research project focused on Moroccan cities. This type of categorical data requires binarization to be able to apply algorithms that do not take non-numeric values as input, which requires transforming this data. The types of this data do not have a hierarchy between them, so we cannot assign a value or a score. This requires transforming them into a vector of the binary values 0 and 1, 1 being assigned as an indicator value for the location that occurs in the vector and the value 0 in the rest of the locations. We take the contract type example which turns into [CDI, CDD, Freelance], and the vector shows the location of each type in the vector, so to represent these types: we have $CDI = [1; 0; 0]$, $CDD = [0;\ 1;\ 0]$, $Freelance = [0; 0; 1]$, we applied the same principle to the other categorical data that we have for the job offer. Then, we performed these NLP operations sequentially on the profile title and the mission (the job description), as shown the Figure 2.

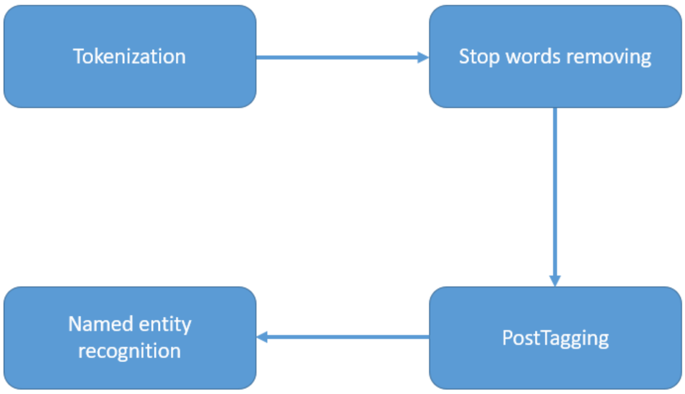

Figure 2. NLP steps applied to profile titles and job offers missions

Before starting the natural language processing, we removed all job offers with missing values so as not to block the steps that follow: foremost, the tokenization [22] is a necessary step in natural language processing that attempts to break text into identifiable linguistic units. It is applied to the job title and to the mission to recover all their data in a token format to apply the following processing. Then, stop words removing: this process removes all words that are unsymbolized and refers to particularly common words that may not convey meaningful information. Afterward, the part-of-speech tagging [23], [24] assigns each token its grammatical category such as verb, adjective, and noun. Ultimately, the named entity recognition technique (NER) allows for identifying tokens and assigning their categories, which can be proper names of people, places, organizations, currency, dates, and times. We used the Stanford Core NLP library [25] to do NLP processing such as text tokenization, PostTagging and NER.

4.3. Filtering and data analysis

4.3.1. Data filtering

After the application of the natural language processing, we recovered the tokens well-cleaned and ready to analyze them, to have as result only the tokens that can help us to recover the requested skills from the job offer. As shown in Figure 3, we have performed analyses and filtering on the result obtained from the NLP treatments of the job title and mission. We have selected the tokens that are "names" identified by the technique of PostTagging, and eliminated other types like "verbs", and "adjectives". Then we applied the filter to the recovered tokens by deleting tokens that are identified by NER, and we recovered only the tokens that are not recognized. To concretize these operations, we give an example:

- Title: "Java Programmer (JSP) | Tetouan"
- Mission: "We are looking for a programmer who is looking to take their experience to the next level. Our programmer must have a solid background in the Java programming language (JSP)"

After performing the NLP steps and then filtering the data that we showed in Figure 3, we obtained the following tokens: "Programmer," "experience," "level," "language," "programming," "Java," and "JSP".

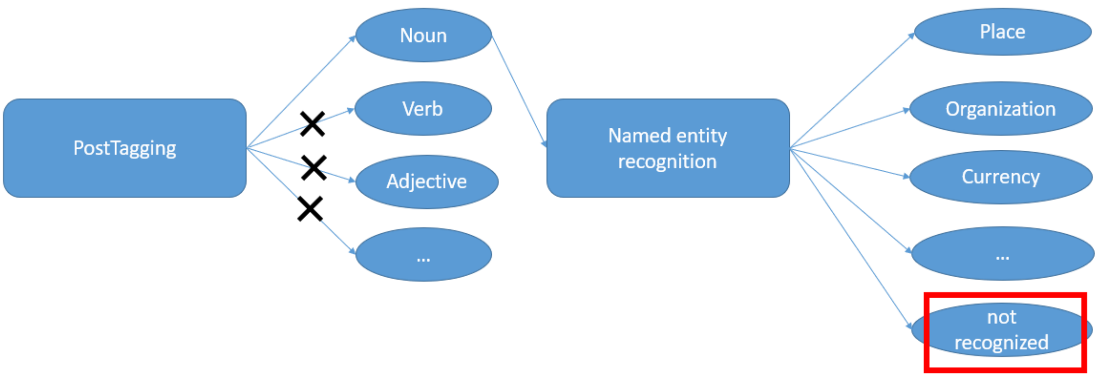

Figure 3. The filtering of tokens to retrieve skills

4.3.2. Data analysis

The goal of this phase is to create two ordered lists: one comprising the skills and the other listing the specialties associated with each skill. We proceed with data analysis after filtering the data and obtaining a set of tokens ready for analysis, focusing primarily on the skills and their corresponding technical word. This analysis involves three main steps: first, the weighting or scoring of the skills; second, the retrieval and grouping of the skill specialties; and third, the organization of the skills as shown below:

- Skills ponderation: In this phase, we sent the tokens recovered from the data filtering to the WordNet and the YAGO2 ontology [26]. We received for each token is it a skill or not as output. We have recovered only the tokens which are skills. Then, we applied the term frequency weighting on all tokens that are identified as a skill.
- Skills specialty recuperations: In this phase, we worked on the list of skills obtained from the previous phase. For each skill or skill alias in the list, we searched for its skill specialty. This search requires the use of the DICE competence center and a standardized hierarchy of O*NET professional specialties [27]. DICE covers the field of economics and information technology, O*NET classifies skills in the medical and artistic fields.

– Skills organization: After obtaining the skill weighting and the specialty to which each skill belongs, we move on to the organization of the skills. We have added the list of skills ordered by descending ranking according to their weightings to the data set of the job offer concerned. We have also added the list of skill specialties to the relevant job posting data, with a descending ranking according to the skill weight to which it belongs.

4.4. Similarity calculation

In this step, we manage to apply the similarity calculation between the e-portfolio and job offers with the same classification of skill specialties (this is the ideal case if it exists) then we apply the similarity calculation on job offers of the same skill specialty ranking except for the last ordered specialty on the list. Then on the same ranking except the last two, and so on we have called this approach the skill specialty ranking logic. We have defined the data model with five criteria (5 dimensions in the vector space model) which are the type of contract, the level of study, the years of experience, the city, and the sector of activity. To calculate the similarity between the e-portfolio and the job offers, we use the 5-dimensional vector space model to represent the e-portfolio vector in question and the job offers vectors that follow the classification logic of skills specialties. The result of each similarity calculation is ordered in descending order to have the most relevant to the least relevant job offers for each e-portfolio. The results for each candidate profile are displayed on an e-portfolio platform interface.

We chose to apply the Euclidean distance as the similarity calculation using the “Cross Distances” operator of the RapidMiner platform [28]. This operator has two inputs, one of them is the set of e-portfolios vectors and the other is the set of job offer vectors. The “Cross Distances” operator has three outputs, the first output is called “result set” which returns the similarity calculation between each e-portfolio of the set of e-portfolios and each job offer of the set of job offers, the second and the third return the two inputs with an ID for each record of the two sets if it does not exist. We implement the similarity calculator for our recommendation system using the process shown in Figure 4.

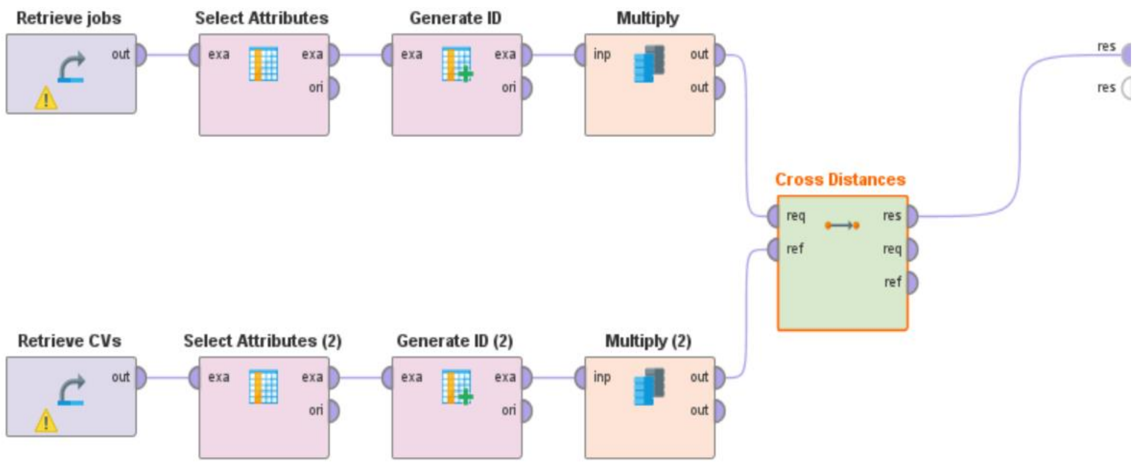

Figure 4. The processes proposed to calculate the similarity

We have designed a process to prepare the data and send it to the “Cross Distances” operator, this process uses four operators: *Retrieve Operator*, *Select Attributes*, *Generate ID*, and *Multiply*, as shown in the Figure 4: starting by the operator *Retrieve Operator* that can access stored information in the repository and load them into the process, we used this operator to load the job offers and e-portfolios dataset.

Afterwards, the operator *Select Attributes* selects a subset of attributes of an ExampleSet and removes the other attributes. This operator was used to retrieve only the selected attributes that correspond to the search criteria which are the same for the dataset of job offers and e-portfolios. Then the operator *Generate ID* that adds a new attribute with an id role in the input ExampleSet. Each example in the input ExampleSet is tagged with an incremented id. If an attribute with an id role already exists, it is overridden by the new id attribute. In our case, we only used this operator to generate an ID that it did not have. Subsequently, the operator *Multiply* takes the RapidMiner object from the input port and delivers copies of it to the output ports. Each connected port creates an independent copy. So, changing one copy does not affect other copies. This operator gave us another copy of the dataset to do other operations on the copy that we will need in the future. In the end, the operator Cross Distances calculates the distance between each example of a “request set” ExampleSet to each example of a “reference set” ExampleSet. This operator is also capable of calculating similarity instead of distance.

5. CONCLUSION

As part of our research work, we have set up a job offer recommendation system based on candidates e-portfolios. For the realization of this system, we started by extracting job offers from recruitment websites, then we applied natural language processing to the extracted data to have data set that is well-cleaned and ready to filter and analyze. We used the Euclidean distance using the vector space model to define the calculator of the similarity between the job offers and the e-portfolio concerned. Finally, we applied the similarity between the e-portfolio and the job offers that follow the logic of the list of skills specialties that we have already defined in this article.

In the future of our research work, we will study the performance of the recommender system by comparing the results due to the use of the Euclidean distance and the cosine similarity that we have already applied previously. We want to integrate more recruitment platforms that have job offers in different formats such as PDF documents, and images. We wish also to add an observatory based on big data techniques to have visibility for Moroccan job offers.

REFERENCES

[1] F. Amato *et al.*, "Challenge: Processing web texts for classifying job offers," in *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, Feb. 2015, pp. 460–463, doi: 10.1109/ICOSC.2015.7050852.

[2] M. N. Ahmed Awan, S. Khan, K. Latif, and A. M. Khattak, "A new approach to information extraction in user-centric E-recruitment systems," *Applied Sciences*, vol. 9, no. 14, Jul. 2019, doi: 10.3390/app9142852.

[3] H. Mgarbi, M. Y. Chkouri, and A. Tahiri, "Purpose and design of a digital environment for the professional insertion of students based on the e-portfolio approach," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 17, no. 01, pp. 36–48, Jan. 2022, doi: 10.3991/ijet.v17i01.26703.

[4] H. Mgarbi, M. Yassin Chkouri, and A. Tahiri, "Building student resume using the eportfolio aproach," *International Journal of Engineering Applied Sciences and Technology*, vol. 6, no. 5, Sep. 2021, doi: 10.33564/IJEAST.2021.v06i05.007.

[5] J. J. Webster and C. Kit, "Tokenization as the initial phase in NLP," in *Proceedings of the 14th conference on Computational linguistics* -, 1992, vol. 4, pp. 1106–1110, doi: 10.3115/992424.992434.

[6] M. Maree, A. B. Kmail, and M. Belkhatir, "Analysis and shortcomings of e-recruitment systems: Towards a semantics-based approach addressing knowledge incompleteness and limited domain coverage," *Journal of Information Science*, vol. 45, no. 6, pp. 713–735, Dec. 2019, doi: 10.1177/0165551518811449.

[7] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 50–70, Jan. 2022, doi: 10.1109/TKDE.2020.2981314.

[8] E. Ferrara, P. De Meo, G. Fiumara, and R. Baumgartner, "Web data extraction, applications and techniques: a survey," *Knowledge-Based Systems*, vol. 70, pp. 301–323, Nov. 2014, doi: 10.1016/j.knosys.2014.07.007.

[9] J. W. Luo and J. J. R. Chong, "Review of natural language processing in radiology," *Neuroimaging Clinics of North America*, vol. 30, no. 4, pp. 447–458, Nov. 2020, doi: 10.1016/j.nic.2020.08.001.

[10] Y. Zou, A. Kiviniemi, and S. W. Jones, "Retrieving similar cases for construction project risk management using Natural Language Processing techniques," *Automation in Construction*, vol. 80, pp. 66–76, Aug. 2017, doi: 10.1016/j.autcon.2017.04.003.

[11] G. Shani and A. Gunawardana, "Evaluating recommendation systems," in *Recommender Systems Handbook*, Boston, MA: Springer US, 2011, pp. 257–297.

[12] R. Mishra and S. Rathi, "Efficient and scalable job recommender system using collaborative filtering," in *ICDSMLA 2019*, 2020, pp. 842–856.

[13] F. Colace, L. Greco, P. Napoletano, and M. De Santo, "A query expansion method based on a weighted word pairs approach," in *Proceedings of the 4th Italian Inform. Retriev. Workshop*, 2013, pp. 17–28.

[14] F. Colace, M. De Santo, L. Greco, and P. Napoletano, "Improving text retrieval accuracy by using a minimal relevance feedback," in *IC3K 2011: Knowledge Discovery, Knowledge Engineering and Knowledge Management*, 2013, pp. 126–140.

[15] F. Colace, M. De Santo, L. Greco, and P. Napoletano, "Text classification using a few labeled examples," *Computers in Human Behavior*, vol. 30, pp. 689–697, Jan. 2014, doi: 10.1016/j.chb.2013.07.043.

[16] G. Barcaroli, A. Nurra, S. Salamone, M. Scannapieco, M. Scarnò, and D. Summa, "Internet as data source in the Istat survey on ICT in enterprises," *Austrian Journal of Statistics*, vol. 44, no. 2, pp. 31–43, Apr. 2015, doi: 10.17713/ajs.v44i2.53.

[17] M. Diaby, E. Viennet, and T. Launay, "Toward the next generation of recruitment tools," in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, Aug. 2013, pp. 821–828, doi: 10.1145/2492517.2500266.

[18] O*NET, "O*NET-SOC 2010 occupations-occupational listings," *O*NET resource center, Onetcenter.org*. https://www.onetcenter.org/taxonomy/2010/list.html (accessed Dec. 02, 2021).

[19] M. Somvanshi, P. Chavan, S. Tambade, and S. V Shinde, "A review of machine learning techniques using decision tree and support vector machine," in *2016 International Conference on Computing Communication Control and automation (ICCUBEA)*, Aug. 2016, pp. 1–7, doi: 10.1109/ICCUBEA.2016.7860040.

[20] "Data miner," *dataminer.io*. https://dataminer.io/ (accessed Dec. 10, 2021).

[21] Y. Su and C.-C. J. Kuo, "Recurrent neural networks and their memory behavior: a survey," *APSIPA Transactions on Signal and Information Processing*, vol. 11, no. 1, 2022, doi: 10.1561/116.00000123.

[22] P. Singh, *Machine learning with PySpark*. Berkeley, CA: Apress, 2019.

[23] E. Brill, "A simple rule-based part of speech tagger," in *Proceedings of the workshop on Speech and Natural Language - HLT '91*, 1992, p. 112, doi: 10.3115/1075527.1075553.

[24] R. Navigli, "Word sense disambiguation," *ACM Computing Surveys*, vol. 41, no. 2, pp. 1–69, 2009, doi: 10.1145/1459352.1459355.

[25] X. Cheng, X. Kong, L. Liao, and B. Li, "A combined method for usage of NLP libraries towards analyzing software documents," in *Advanced Information Systems Engineering*, 2020, pp. 515–529.

[26] A. B. Kmail, M. Maree, M. Belkhatir, and S. 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)*, Nov. 2015, pp. 620–627, doi: 10.1109/ICTAI.2015.95.

[27] A. Zaroor, M. Maree, and M. Sabha, "A hybrid approach to conceptual classification and ranking of resumes and their corresponding job posts," in *IDT 2017: Intelligent Decision Technologies 2017*, 2018, pp. 107–119.
[28] I. Mierswa and R. Klinkenberg, "Rapid Miner Studio," *rapidminer.com.* https://rapidminer.com/ (accessed Dec. 20, 2021).

BIOGRAPHIES OF AUTHORS

Hanae Mgarbi is a Ph.D. student at the research laboratory of the Information System and Software Engineering (SIGL) at the National School of Applied Sciences, and computer engineering at the Abdelmalek Essaadi University, Tetouan, Morocco. She can be contacted at email: hanae.mgarbi@etu.uae.ac.ma.

Mohamed Yassin Chkouri is the leader and founder of the accredited research laboratory of the Information System and Software Engineering (SIGL) and the Head of the Department of Computer Engineering since 10/2016 at the National School of Applied Sciences, Abdelmalek Essaadi University, Tetouan, Morocco. He can be contacted at email: mychkouri@uae.ac.ma.

Abderrahim Tahiri received his engineering degree and Ph.D. in Computer Science from the University of Abdelmalek Essaadi-Morocco. Since 2010, he has been an Associate Professor in the Department of Informatics at the National School of Applied Sciences of Tetouan, School of Engineering at Abdelmalek Essaadi University. He has been a member of the school Council and was appointed Coordinator of the Informatics branch and then head of the department. He can be contacted at email: t.abderrahim@uae.ac.ma.