IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 January 2023, accepted 15 February 2023, date of publication 22 February 2023, date of current version 28 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3247866

RESEARCH ARTICLE

Two Stage Job Title Identification System for Online Job Advertisements

IBRAHIM RAHHAL[1,2], KATHLEEN M. CARLEY[3], (Fellow, IEEE), ISMAIL KASSOU[1], AND MOUNIR GHOGHO[2], (Fellow, IEEE)

[1]ENSIAS, Mohammed V University, Rabat 10500, Morocco
[2]Research Laboratory (TICLab), College of Engineering and Architecture, International University of Rabat, Sale 11103, Morocco
[3]Institute for Software Research, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Corresponding author: Ibrahim Rahhal (brahim.rahhal@uir.ac.ma)

This work was supported in part by the Fulbright Program.

ABSTRACT Data science techniques are powerful tools for extracting knowledge from large datasets. Analyzing the job market by classifying online job advertisements (ads) has recently received much attention. Various approaches for multi-label classification (e.g., self-supervised learning and clustering) have been developed to identify the occupation from a job advertisement and have achieved a satisfying performance. However, these approaches require labeled datasets with hundreds of thousands of examples and focus on specific databases such as the Occupational Information Network (O*NET) that are more adapted to the US job market. In this paper, we present a two-stage job title identification methodology to address the case of small datasets. We use Bidirectional Encoder Representations from Transformers (BERT) to first classify the job ads according to their corresponding sector (e.g., Information Technology, Agriculture). Then, we use unsupervised machine learning algorithms and some similarity measures to find the closest matching job title from the list of occupations within the predicted sector. We also propose a novel document embedding strategy to address the issues of processing and classifying job ads. Our experimental results show that the proposed two-stage approach improves the job title identification accuracy by 14% to achieve more than 85% in some sectors. Moreover, we found that incorporating document embedding-based approaches such as weighting strategies and noise removal improves the classification accuracy by 23.5% compared to approaches based on the Bag of words model. Further evaluations verify that the proposed methodology either outperforms or performs at least as well as the state-of-the-art methods. Applying the proposed methodology to Moroccan job market data has helped identify emerging and high-demand occupations in Morocco.

INDEX TERMS BERT, job market analysis, job title classification, job title identification, machine learning, natural language processing.

NOMENCLATURE

d	Document containing n words.
S	The Number of documents in the document set.
$V(d)$	The vector representation of document d.
$\mathrm{w2v}(\mathrm{w}_i)$	The vector representation of the word w_i using w2v.
λ_i	The weight assigned to the word w_i.
$V(O)$	The vector representation of the occupation.
$V(J)$	The vector representation of the job advertisement.
α	The balance factor used for calculating the vector representation of the occupation indicating the degree of contribution of the title and the description.
β	The balance factor used for calculating the vector representation of the job advertisement indicating the degree of contribution of the title and the description.
N	The number of words selected from the description using the feature selection methods.

The associate editor coordinating the review of this manuscript and approving it for publication was Vlad Diaconita.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

w_i	A word within a document.
J	Job advertisement.
J_t	Job advertisement title.
J_d	Job advertisement description.
O	Occupation.
O_t	Occupation title.
O_d	Occupation description.
D	Set of documents.

I. INTRODUCTION

The widespread use of the Internet in many industries due to the digitization of processes and the development of social media has resulted in a large amount of data that needs to be processed and analyzed quickly and efficiently to extract valuable insights that can help in decision-making [1]. In this context, data science techniques can be powerful tools for extracting information from large datasets, facilitating the process of classifying different types of data (e.g., text, images, and video) [2], and can also solve many other tasks that are handled in a traditional manner, which is often time and resource consuming.

Similarly, the job market shifted from traditional channels to online websites and job portals. This is because employers and recruiters share various job advertisements across different platforms to expand their reach and target more job seekers. This shift represents an opportunity to understand the needs of the job market from the vast amount of data shared daily, which can benefit many stakeholders [3]. In particular, identifying the requirements in terms of skills and occupations can help labor market analysts and policymakers foster employment and also help job seekers and students find suitable jobs and the training needed to successfully transition to the job market [4].

Classifying online job advertisements (ads) is not a trivial task. Indeed, the information contained in a job ad is expressed in plain text in a non-structured or semi-structured format, and the lexicon used by employers in the text is often very different from the occupational classifiers and the databases developed by human resources experts. In addition, job ads often include overly generic information that is not relevant to the position. This adds noise to the process of matching the job advertisement to its corresponding occupation. For instance, a job advertisement may have a title that includes information about the city where the job is located or some salary information. Also, the description can contain information about the company and information about other tasks that do not necessarily relate to the desired occupation. It is therefore necessary to apply advanced techniques for word and document representation and to use novel feature extraction methods to address these challenges.

Most of the proposed methods for dealing with occupation normalization consider it to be a classification or clustering problem. In this context, several text classifiers, ranging from traditional machine learning (ML) models to deep learning models, have been proposed for this task, such as support vector machine (SVM) [1], naïve bayes [5], k-nearest-neighbor (KNN) [1], [5], artificial neural networks (ANNs) [6] and Bidirectional Encoder Representations from Transformers (BERT) [7]. While some studies used the combination of the title and the description to perform classification, the authors in [8] used only the title and found that 30% of the job offer titles did not contain enough information to identify the occupation. Similarly, the authors in [9] looked at the text of the job description only and found that each job description could correspond to more than one occupation. To the best of our knowledge, no previous study has examined the degree to which the title and description contribute when normalizing job ads. Classifying job ads using an occupational classifier or internal taxonomy has generally achieved satisfying results [6], [10]. However, these methodologies require human-labeled datasets with hundreds of thousands of examples, which is time-consuming and resource-intensive. In addition, updating the occupation description or including a newly created occupation in the occupational classifier is very difficult, as the entire training process must be repeated. Also, most prior work only focuses on English-language job ads and uses specific occupational classifiers such as the Occupational Information Network (O*NET); the extension of existing work to job ads written in other languages is challenging. which makes it extremely difficult to replicate their methodologies in other languages.

On the other hand, using unsupervised models to identify the occupation, such as clustering [11] and field similarity [12], avoids training the model with labeled data, which are not always available; this is particularly relevant since we are dealing with a large number of occupations. The majority of previous works have relied on simple techniques for word embedding such as Bag of Words (BOW) [1], [12] or Term Frequency Inverse Document Frequency (TFIDF) [11] to generate word embedding and have applied averaging methods to calculate the document embedding. However, these techniques are considered weak in capturing the semantic relationships between the words, especially when we are dealing with job ads written by multiple employers who are using different lexicons. Therefore, word embedding approaches and feature extraction techniques [13] need to be closely monitored to achieve the highest results, as state-of-the-art techniques do not perform well in all cases [11].

In this paper, we propose a job title identification methodology based on self-supervised and unsupervised machine learning algorithms with minimal labeling and high accuracy that can be replicated on data from other countries to overcome the limitations mentioned above. The proposed methodology consists of two steps: the classification of job ads by sector and the matching of job ads with occupations belonging to the predicted sector. The step of job ads classification is done using several text classifiers such as SVM, Naïve Bayes, Logistic Regression, and BERT to classify job ads into their corresponding sectors (e.g., Information Technology (IT), Agriculture) which will help us focus on the occupations of the predicted sector instead of using all the occupations from the occupational classifier. For the job title identification step, we compare different techniques for vector representation of texts and use several combinations and parameters to propose a customized document embedding strategy. We also test several feature selection methods to extract important keywords from the description and analyze the degree of contribution of the title and the description in improving the results. Finally, we calculate the similarity between the job ad representation and the occupation representations belonging to the predicted sector to choose the closest one. To do this, we collect the French occupational classifier "Pole Emploi" and about two hundred thousand job ads from job portals. When used to identify the occupation title on a random sample of job ads, our methodology achieves an overall accuracy of 76.5% and more than 85% for some sectors which is considered high accuracy compared to prior work. Furthermore, the effectiveness of our approach was validated with the help of a team of domain experts who manually labeled a sample of our dataset. Finally, we applied our methodology to a dataset of 248,059 job ads in the French language to get an overview of the Moroccan job market, especially the IT sector. This study allows us to shed light on key sectors and occupations in the Moroccan job market where there is a high demand for IT profiles and Telemarketers which was identified by a previous study on the offshore sector in Morocco [14]. Using this methodology, we can identify emerging occupations that can help decision-makers including universities to take appropriate measures to adapt their programs and curricula, and to also help job seekers and students in their orientation by taking a career path that leads to employment [4].

A. CONTRIBUTIONS

The main contributions of this paper can be summarized as follows:

- We propose a methodology for occupation identification in a scenario of a lack of labeled data so that it can be replicated for other languages and countries.
- We provide a comparison of document representation strategies for solving the problem of occupation identification and identify the degree of contribution of the title and description of the job ad in the matching process.
- We draw insights on the Moroccan IT job market needs in terms of occupation and construct a Moroccan job ads dataset in the French language, which can relieve the limitation in this field.

The remainder of this paper is organized as follows: Section II provides a comprehensive literature study of job title identification and related works in document embedding. Section III details the data collection and pre-processing. Section IV describes our proposed methodology and explains each component separately. In Section V, we discuss the data annotation process and experimental settings. We evaluate our methodology and present the results in Section VI, and compare our approach with existing methodologies on real datasets in Section VII. In Section VIII, we apply our methodology to get an overview of the Moroccan job market. The discussion and contributions are presented in Section IX. Finally, we conclude and discuss future work in Section X.

II. RELATED WORK

A. JOB TITLE IDENTIFICATION

Many studies have attempted to normalize job ads titles as a first step in structuring job ads before identifying the required skills based on job roles. An occupation is defined as a grouping of jobs that involve similar tasks and that require a similar skill set. It is important to note that occupations should not be confused with jobs or job titles. While a job is tied to a specific work context and executed by one person, occupations group jobs based on common characteristics. Identifying the required occupations in the job market can be considered as a top-down approach to discovering the required skills by inferring skills from structured skill bases that encompass full occupation descriptions such as the International Standard Classification of Occupations (ISCO) or O*NET.

There are two approaches to identifying job titles from job advertisements. The first approach uses supervised models to classify job titles, while the second approach uses unsupervised models to find the closest job title. In this section, we review the previous studies on job ad classification methods and summarize them in Table 1.

Many studies framed the task of job title identification as a text classification task where job ads were classified to their corresponding occupation based on the standard referential using SVM and KNN. In particular, in [1] and [18], CarrerBuilder.com used a multi-stage classifier to tackle a large number of classes which is almost similar to the application domain (online recruitment) used by LinkedIn's job title classification system [15], where they utilize a heavily manual phrase-based classification system dependent on short-text and a heavy reliance on crowd-sourced labeling of training samples. Moreover, in [16], they leveraged string similarity, where similar job titles were fed to the siamese network to learn to classify job titles. For this task, they used an in-house taxonomy to classify the job titles instead of using O*NET and ISCO bases. Also in [6], [7], [8], and [10] they used text classifiers, from traditional machine learning models to deep learning models respectively based on ISCO occupation classifiers or on customized lists of occupations. Text classifiers used in [6] and [10] showed good performance in extracting the needed skills of some occupations, while text classifiers of [8] achieved a less interesting accuracy because they used only the title of the job ad and didn't include the description. Finally, the authors identified that about 30% of the job offer titles do not carry enough information to identify the occupation. A similar study was described in [5] where the authors used a dataset from Kaggle to classify job titles based on the query description into 30 distinct classes corresponding to the top 30 occupations. They used several algorithms such as Bernoulli's Naïve Bayes, Multinomial Naïve Bayes, Random Forest, and Linear SVM and found that Linear SVM gives

TABLE 1. Literature comparison in terms of job title identification.

Paper	Year	Occupational base reference	Method for job title identification
Bekkerman et al. [15]	2011	Customized base	ML text classifiers
Malherbe et al. [12]	2015	ROME	Compute field simiarity
Javed et al. [1]	2015	O*NET	Two stage classifiers
Amato et al. [8]	2015	ROME (ISCO)	ML text classifiers
Neculoiu et al. [16]	2016	Customized base	LSTM Siamese network
Karakastsanis et al. [17]	2017	O*NET	LSI
Zhu et al. [18]	2017	O*NET	Two stage classifiers
Boselli et al. [6]	2017	ISCO-08	ML text classifiers
Boseli et al. [10]	2018	ISCO-08	Deep learning text classifiers
Colace et al. [19]	2019	ISCO-08	ML classifier
Vinel et al. [11]	2019	-	Clustering methods
Mittal et al. [5]	2020	Customized list	ML text classifiers
Van et al. [7]	2020	Customized IT-related job	Deep neural network models
Mankolli et al. [20]	2021	-	Hybrid (classification, similarity)
Tran et al. [9]	2021	Customized list	Multi-label text classification

the best results for job title classification and that increasing the training set improves the accuracy. Finally, in [9] authors propose a multi-label classification approach for predicting relevant job titles from job description texts and consider that each job description may correspond to more than one occupation. They implement the algorithm presented in [7] using different pre-trained language models to apply it to the job titles prediction problem. They found that BERT with a multilingual pre-trained model obtained the highest result on their dataset and that the description alone is not enough for the prediction, so they need to reference extra information such as job name, job level, and job requirements.

The main disadvantage of text classifiers is the expense of data acquisition for training with many thousands of groups of occupations, often not too dissimilar from one another.

The second approach is the use of unsupervised models to identify job titles. The advantage of this approach is that there is no need to train the model. In [12], authors leverage field similarity between job ads and job occupation descriptions to infer the closest job title. By using field similarity, the author compares the job ad with only occupations from the same field, which reduces the number of comparisons and increases accuracy. In the same context, the authors of [20] used a hybrid machine learning system to identify similar job titles by generating a list of industries where each job title is relevant. They then used KNN to find the most similar k job titles. Additionally, clustering is also used in [11] to identify clusters of occupation. This methodology achieved the best performance using Additive Regularization on Topic Models (ARTM) when the number of clusters is predetermined.

Similar to works discussed above, topic modeling was used for job title identification [19], [21]. The patterns are inferred from the word that appears in the text. Authors of [19] used topic modeling with an explanation model in order to classify job titles and to identify the correlated word with each job title. Also, In [17], the author used Latent Semantic Indexing (LSI) and cosine similarity to match the job advertisement extracted from the web with occupation description data from the O*NET database.

The performance of such approaches was evaluated using different metrics depending on the used models. For text classifiers, accuracy, recall, and precision were used to compare the performance of the classifiers on a labeled dataset of job ads, while for the unsupervised approaches, the accuracy metric was mainly used. Moreover, in a recent study [1], [17], crowd-sourcing was used to evaluate their classifier, where users were invited to rank or select the most appropriate occupation for the job ad.

In this paper, due to the lack of labeled datasets that can be used for the training step, we opted to use a combination of the two approaches. We first classify the job ads only to their corresponding sector, which is the top-level of the occupational base reference by using the BERT for classification. This step only uses a few labeled data to fine-tune the model. In the second step, we use unsupervised models to compare different methods for representing the collected job ads and the occupations from the directory instead of relying on the bag of word representation. We also measure the contribution of both the job title and the description, instead of relying only on the title, as mentioned in most of the prior work. Finally, we investigate different methods to calculate the similarity measures between the job advertisement and the occupations from the same field to find the most similar one using document embedding. In addition, we use the French occupational directory "Pole Emploi" instead of using ISCO and O*NET which can relieve the limitation in this field by providing more comprehensive and up-to-date occupational information on job offers in French.

B. WORD AND DOCUMENT EMBEDDING

Word embeddings are distributed representations of text in a low-dimensional space. It allows words with similar meanings to have a similar representation by capturing the semantic and lexical properties of words, and also captures some relationships between the words [22] (e.g., V_{king} - $V_{man} + V_{women} = V_{Queen}$). These representations are generally obtained in three ways: (a) from word counts in a sentence such as BOW, CountVectorizer, and TFIDF. (b) from internal representations of words in neural networks such as FastText [23] and Word2Vec (W2V) [22], [24], [25]. (c) from word co-occurrences by incorporating global statistics such as Glove [26]. Previous work has found that neural networks perform better than Glove for a range of lexical semantic tasks [18], [27]. Neural networks such as W2V optimize the accuracy of prediction of surrounding words for each word based on the skip-gram model and preserve some semantic relationships of corresponding words. The word embedding presented above corresponds to a single word or short phrase, but we need vector representations of job ads that contain short text (e.g., the job title) and long text (e.g., the job description).

Embeddings for sentences or paragraphs are created by composing word embeddings. This can be achieved by averaging unweighted word vectors [22] or by doing some multiplicative operations [28]. In this context, more sophisticated architectures can also be used: for instance, recursive neural networks [29], long short-term memory networks (LSTMs) [30], convolutional neural networks [31] and also latent representation [32], where the authors proposed paragraph vectors to represent whole paragraphs or documents produced by neural networks whose objective is to predict words in a document. Other state-of-the-art approaches are based on transfer learning models in natural language processing (NLP) such as BERT [33]. BERT is a method of pre-training language representations that applies bidirectional training of the transformer. It can be used to extract quality features from text or can be fine-tuned with new data for specific tasks such as classification, entity recognition, or question-answering. Moreover, BERT offers an advantage over other models because it produces word representations that are dynamically informed by the words around them instead of having a fixed representation regardless of the context.

Regarding the task of job title identification, most prior works used a specific method for a specific task (word representation, feature extraction). For example, In [1] they used BOW for word representation and then improved their work in [18] to replace BOW with W2V and finally used different weighting strategies to calculate document embedding before applying the classifier. In [11] they applied a variety of methods including TFIDF, W2V, FastText and BERT to represent document embedding. Unfortunately, they found out that BERT did not perform as expected because of the difference in vocabulary between the training dataset and job vacancies text. In addition, they found out that W2V gives accurate results when the number of clusters is not specified. Finally, in [12] a work very similar to the second step of our methodology, they used only BOW for word representation, which does not capture synonyms or preserve semantic relationships.

Although all these approaches are relevant and effective, they differ from our approach. Indeed, we aim at dealing with job offers that contain long text and short text with a high degree of heterogeneity. The right choice of word representation can make a big contribution to the results. Here, the joint use of different techniques would be beneficial in evaluating the effectiveness of these approaches in our application domain.

III. DATA COLLECTION & PRE-PROCESSING

A. COLLECTION OF JOB ADS

In order to build our job title identification system, we collected job postings in French from different websites. These websites were identified using Optioncarriere.ma, which is a job aggregator that contains the most frequently used websites for job ads in Morocco [34], [35]. Since we are interested in all occupations, we did not have to add any filters to the URL; we just insert the domain name of the website and specify the section related to job offers. Since the language gap in prior work was French, we focused on French job ads.

For collecting the job ads, we developed a web crawler with Python for each website from the identified list (see table 2). The crawler starts by collecting URLs of the available job ads, then stores the content of each URL which is the content of the job ad in HTML format. The 10 websites were scrapped weekly from January 2019 to June 2021 to have a full representation of the majority of job ads to validate our work and also to use this dataset in our case study. Our dedicated server ran for more than 11 hours every time it was launched through a high-speed internet connection to perform this task.

After studying the structure of the identified websites, we found that most of the shared job advertisements are in a non-structured or semi-structured format, meaning that the description contains information on the company, the job description, hard skills, and soft skills in plain text without structure. Thus, we end up collecting only the job title and the description and sometimes other details, when available, that can help us in the labeling process. To collect this information, the crawler was also programmed to parse the HTML code and use tags to identify the content of each element (e.g. title, description, company name, sector, and other requirements). The total number of collected items before the de-duplication process (discussed later in the methodology section) was 248,059 job ad records which represent nearly all the published job ads online. The job ads were saved in a MySQL database with a table for each website, taking up 11.6 GB of storage space. This dataset will be used to evaluate our methodology and will also be used to analyze the needs in terms of occupations for the Moroccan job market.

TABLE 2. List of job portals used in the data collection process.

Website	Data type
www.avito.com	Non-structured
www.job-ma.com	Non-structured
www.emploi.ma	Semi-structured
www.callcenter.ma	Semi-structured
www.rekrute.com	Semi-structured
www.jobmaroc.com	Semi-structured
www.embauchemaroc.com	Semi-structured
www.m-job.ma	Semi-structured
www.marocemploi.net	Non-structured
www.rekruteonline.com	Semi-structured

B. OCCUPATIONS DIRECTORY - POLE EMPLOI

To identify the occupations corresponding to the collected job advertisements we need a standard occupation directory. In our study, we use Pole Emploi [36] a job directory equivalent to ISCO and O*NET developed by the French National Employment Agency where the categories are part of the Operational Directory of Professions and Jobs (Rome). Pole Emploi is a directory for organizing jobs into a clearly defined set of groups according to the tasks and duties performed in the job. It is categorized in a four-level hierarchy with 12 major top levels called domains or sectors in our case, 110 professional domains in the second level, 512 broad occupations in the third level, and 11,000 leaf-level occupations (see figure 1 for more details). The leaf-level occupations contain similar job titles to the parent nodes that contain the main occupation with the description. This means that similar job titles have the same information related to the top occupation. For example, a Java developer, web developer, or app developer needs the same skills and all of them are part of the same occupation called IT development. Each occupation contains information like occupation title, occupation description, tasks, requirements, work environments, general skills, and specific skills. In our context, we use the leaf-level title with the occupation description from the third level to estimate the distance between each specific job ad found online and the set of occupations existing in the occupation directory. The reason for using the leaf level is that recruiters tend to use specific keywords related to the job that do not exist at the third level. For example, the third level title is IT studies and development which is a general title, and the fourth level titles are web developer, app developer, and others.

Similarly, we developed a web crawler to collect each occupation from the job directory. Each occupation is identified by a unique ID. To access the occupation, we simply have to change the ID (codeRome) in the URL [37]. The crawler visits each URL and retrieves its content in HTML format. It then begins to parse the code line by line to identify the necessary content, namely the occupation title, a list of similar job titles, the occupation description, tasks, requirements, work environments, general skills, and specific skills (hard skills). Finally, each occupation is stored in a row of our MySQL database, and then the URL changes to process the next occupation until the required number of occupations is reached.

C. PRE-PROCESSING

Before using the collected textual data regarding both the job advertisements and Pole Emploi occupations, cleaning of the data is carried out. Following common text-mining techniques, we start with tokenization, lowercase reduction, punctuation, and special character removal including accents. Then we delete terms appearing only once as well as we reduced the produced vocabulary by deleting stop words and a specific set of words not conveying relevant information. After this, we perform lemmatization which consists in linking all the forms inflected from the same term to the same lemma (i.e. lemmatization reduces the words "studies", ''studying" to the word "study"). This step is important because it allows a better comparison of certain words and also reduces the dimension of the vocabulary.

In our context, relying on the previous steps will not be enough because our collected job ads are unstructured. The title can contain other words not relevant to the job or to the profile that recruiters are looking for, such as "looking for" or "full time" (for example, the title "looking for a full-time Java programmer " can become "Java programmer"). Similarly, the job ad description can contain information about the company. To remedy this situation, we manually review a set of titles and create a delete list that will only be applied to the job ad titles. On the other hand, we split the job ad description into sentences that we classified into two categories, a company description or job description (skills, duties, and qualifications) using a naïve Bayes classifier that we trained on a manually labeled sample containing 300 job ads. Finally, our data will include a clean job title and a description that contains only information related to the job. This step can be skipped in the case of using structured job ads where the title is clear, and the company description is separate from the job description. This text pre-processing pipeline was used before applying any of the approaches described in the next section.

IV. METHODOLOGY

A. OVERVIEW

Job title identification remains a challenging task, even with advances in natural language processing (NLP) techniques,

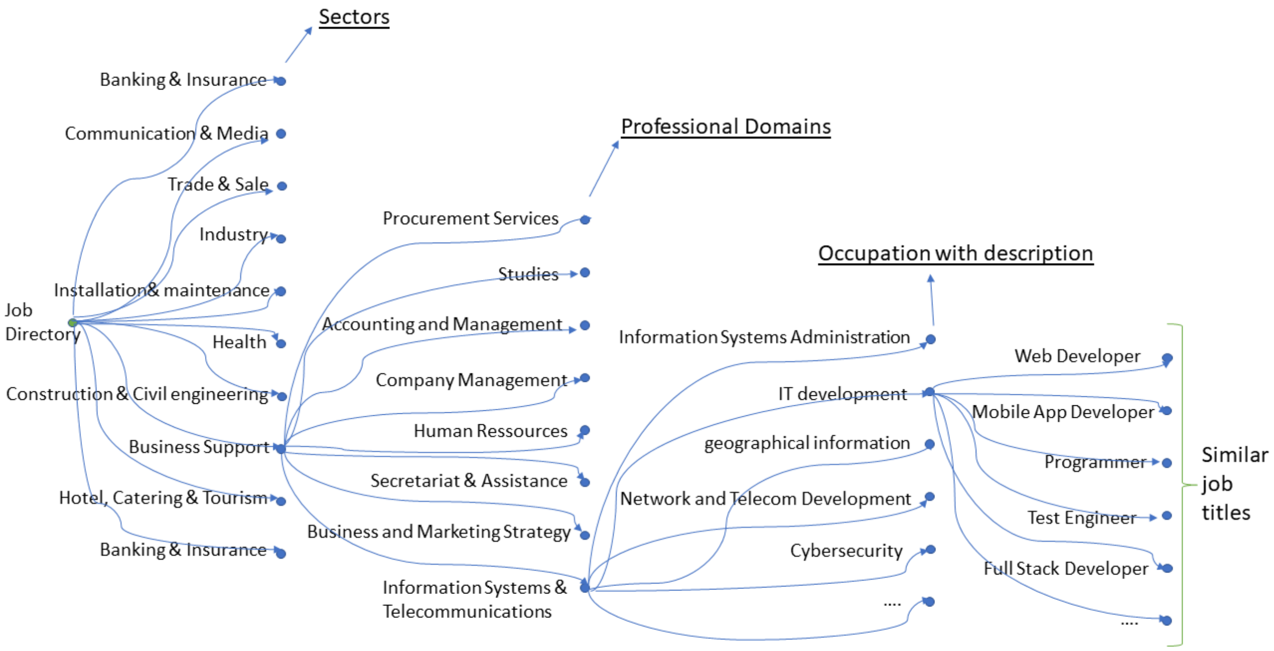

FIGURE 1. Overview on the occupational classifier scheme of "Pole Emploi" (the fourth level shows different job titles within the occupation of IT development that have the same description. This occupation belongs to the business support sector.).

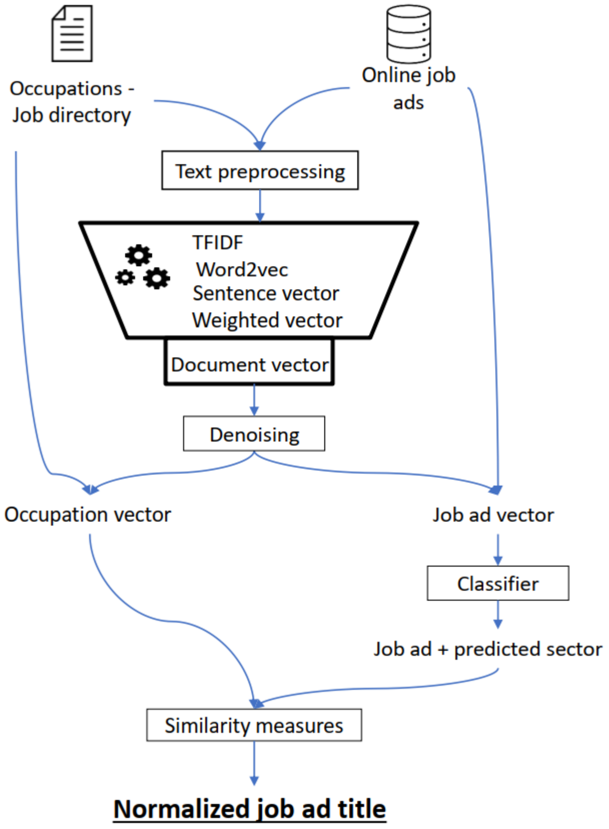

FIGURE 2. Overview of our proposed approach.

because the text appears in a range of different expressions written by different individuals. To tackle this problem, we propose using a combination of supervised and unsupervised machine learning algorithms. This combination will enable us to predict the sector of the job ad and then to identify its corresponding occupation. Figure 2 shows the proposed methodology that can be decomposed into 6 steps:

- *Step 1 (Data Collection):* We collect the national occupation directory and online job ads written in French.

- *Step 2 (Data Pre-Processing):* We apply some pre-processing techniques to structure and clean the text.

- *Step 3 (Document Representation):* we use several techniques to generate a representation for long text.

- *Step 4 (Denoising):* We extract the top words from the descriptions to have a better representation of the job ads and the occupations.

- *Step 5 (Sector Prediction):* We implement a classifier that will help us predict the sector of the job ad before identifying the job title by testing traditional machine learning models and deep learning models.

- *Step 6 (Job Title Identification):* We calculate similarity measures between the job ad and the occupations within the predicted sector and finally keep the one with the smallest similarity distance.

B. DEFINITIONS AND NOTATIONS

1) DOCUMENT

In this work, we refer to a document d as a sequence of consecutive words: d $=< w_1, w_2, \ldots, w_n >$, where w$_i$ is a word in the document d.

2) JOB AD

Most of the time, a job ad is composed of a title and a description. If we have more details we add them to the description. We represent a job ad in our context as follows: J $=< J_t, J_d >$, where J$_t$ is the job ad title and J$_d$ is the job ad description, which is usually longer than the title.

3) OCCUPATION

The occupation has the same format as the job ad where it contains a title and a description. We represent it as follows: O $=< O_t, O_d >$, where O$_t$ is the occupation title and O$_d$ is the description of the occupation.

C. DOCUMENT REPRESENTATION

The right choice of document representation can make a remarkable contribution to the results. In our related work section, we discussed different methods to represent a document ranging from BOW to neural networks. This section will focus on TFIDF and W2V representations.

1) TFIDF

TFIDF is one of the most popular term-weighting schemes. It reflects how important a word is to a document in a corpus. The TFIDF value increases proportionally to the number of times a word appears in the document. This value is offset by the number of documents in the corpus that contain the word. To put it in more formal mathematical terms, the TFIDF score for the word w in document d from the document set D is calculated as follows:

$$\begin{aligned} \mathrm{tfidf}(w,d,D) &= \mathrm{tf}(w,d)\cdot \mathrm{idf}(w,D) \\ \mathrm{tf}(w,D) &= \log(1{+}\mathrm{freq}(w,d)) \\ \mathrm{idf}(w,D) &= \log(\frac{\mathrm{S}}{\mathrm{count}(d\in D : w\in d)}) \end{aligned} \tag{1}$$

where S is the number of documents in the collection, tf is the term frequency and idf is the inverse document frequency.

2) W2V

Our job ads and occupations are long documents with many consecutive words. To represent them we use a weighted framework to generate the document frequency since W2V gives us a representation of a single word or a short phrase. Given a document d and the W2V model, the representation of the document vector is calculated as follows:

$$\mathrm{V(d)} = \frac{\sum_{w_i\in d}\lambda_i\cdot \mathrm{w2v}(w_i)}{\sum \lambda_i} \tag{2}$$

where w2v(w$_i$) is the vector of the word w$_i$ generated by W2V model and λ_i is the weight assigned to the word w$_i$.

We tried to build and use our own W2V model trained on data related to the collected job ads and occupations. However, the results of the model were not satisfying since our training data is small and W2V needs a large corpus. Instead, we used a pre-trained W2V model on Wikipedia in French [38]. We also tried FastText and ParagraphVectors for word representations but they did not improve our results and that was the same result for previous works [18], [39].

To determine the weight λ_i we used several weighting strategies. In this work, we only report the most interesting ones.

a: UNIFORM

We consider all words to have the same weight. So $\lambda_i = 1$. This strategy can be applied to all the documents (job ad title (J$_t$), Job ad description (J$_d$), occupation title (O$_t$) and occupation description (O$_d$)).

b: TFIDF FOR DESCRIPTION

In a document, some words are more important than others and need to be highlighted to keep their meaning in the document. TFIDF is a common weighting strategy used for text classification tasks that reflects how important a word is in a document. Hence,

$$\lambda_{id} = \mathrm{tfidf}(w_i, d, D)$$

by using equation 1, each word vector will be multiplied by its TFIDF score within the document. This weighting strategy can be tested on J$_d$ and O$_d$ because the title which is a short phrase is usually formed of unique words so the term frequency (TF) is equal to 1.

c: WORD FREQUENCY

Word frequency is a reliable way to get rid of irrelevant words within a document. Especially for short documents such as J$_t$ and O$_t$ that are supposed to be more informative. We used word frequency to reduce the contribution of words related to locations such as "Casablanca" or salary. So,

$$\lambda_i = \log(\mathrm{freq}(w_i))$$

where freq(w$_i$) is the frequency of the word w$_i$ in our dataset that contains the titles and the descriptions. This strategy can be applied to all the documents J$_t$, J$_d$, O$_t$ and O$_d$.

D. JOB ADS AND OCCUPATIONS REPRESENTATION

Most prior works consider that the title is more important than the description since it contains dense words that are relevant to the job [1], [10], [18]. The description adds more context to the title and helps improve prediction. In our context, recruiters publish their job ads in an unstructured format which means that we cannot rely on the title alone. So in order to produce the final vector representation of the job ad and the occupation, we used this formula:

$$\mathrm{V}(O) = \alpha\cdot \mathrm{V}(O_t) + (1-\alpha)\cdot \mathrm{V}(O_d) \tag{3}$$

$$\mathrm{V}(J) = \beta\cdot \mathrm{V}(J_t) + (1-\beta)\cdot \mathrm{V}(J_d) \tag{4}$$

where $\alpha, \beta \in [0, 1]$ are balance factors that define how much a document is contributing to the final vector. V(O) is the vector of the occupation - V(O$_t$) is the vector of the occupation title - V(O$_d$) is the vector of the words representing the occupation description - V(J) is the vector of the job ad - V(J_t) is the vector of the job ad title - V(J_d) is the vector of the words representing the job ad description.

E. DENOISING - KEYWORDS SELECTION

The job ad description and the occupation description are long text compared to the title of the job ad and the title of the occupation. These are usually between 2 and 5 words. The description usually contains words that are not informative. When we calculate the representation of the job advertisement V(J) using equation 4 we can lose the semantics of the text. Keeping informative words and removing the noise can improve our results and has a significant impact on reducing the training time and the required memory [2]. For this purpose, several feature selection methods are used such as Information Gain (IG) [40] Chi-Square (CHI) [41] and TFIDF [42]. In our evaluation, we tested both CHI and TFIDF but we are planning to report only the result of the CHI method since it's the one with the highest results [43]. Chi-square statistically sorts the words based on the calculated score, then we discard the words with low weights and keep the top N words [44].

F. TEXT SIMILARITY

Equations 3 and 4 provide the document representation of all the occupations from the job directory and the collected job ads. To normalize our job ads and identify the appropriate occupation, we need to compare each job ad with all available occupations and select the occupation with the smallest distance. Previous studies have used the same method to identify the occupation directly or to determine the cluster containing the occupation [17], [45]. Several algorithms were tested to calculate the distance, including the Euclidean distance, Cosine similarity, and Word Mover Distance (WMD) [46]. After evaluating a sample of the scores produced, we decided to keep WMD. This method uses the word embedding of the words in two texts to measure the minimum distance that the words in one text need to travel in semantic space to reach the words in the other text.

To identify the occupation, we followed the algorithm described in Algorithm 1. The steps are as follows: First, we include the list of job ads to normalize and the list of occupations from the job directory. Second, we use the optimal document representation values, determined by testing different techniques, to produce a vector representation for each job ad and occupation. Third, we Calculate the distance between each job ad and all the occupations using WMD and record the distance values for each comparison (V(J) with all V(O)). Finally, we choose the occupation with the smallest distance, which indicates the highest similarity between the job ad and occupation.

G. TEXT CLASSIFICATION

Text classification is a supervised machine learning technique that automatically builds a classifier by learning from a set of pre-classified documents called a training dataset [47], [48].

Algorithm 1 Calculate Similarity Measures

Input: List of job ads representation, List of occupations representation Output: Distance value, Normalized job titles L

```
$Jobs$ ←ListOfJobAds
$Occupations$ ←ListOfOccupations
for Each job in Jobs do
    V($J$) ←JobAdRepresentation
    for Each occupation in Occupations do
        V($O$) ←OccupationRepresentation
        $distance$ ←distance(V($O$),V($J$))
        if $value > distance$ then
            $value \leftarrow distance$
            Title←TitleOf(V($O$))
        end if
    end for
    append Title to L
end for
Return L, distance
```

To find the closest occupation to a job ad, it is necessary to compare it with all the available occupations in the job directory, which can be time-consuming and memory-intensive, leading to suboptimal results. A solution to this problem is to include a text classification step that first classifies the job ad into its corresponding sector, and then compares its representation with the occupations within the predicted sector only.

We evaluated several algorithms for developing the text classifier using a dataset of 2028 job ads, which will be described in the next section. The algorithms evaluated were: Naïve Bayes [49], SVM [50], Logistic Regression and BERT. For BERT, we fine-tuned CamemBERT [51], which is a state-of-the-art language model for French based on the RoBERTa architecture [52], and pre-trained it on the French subcorpus using our training dataset.

H. COMPUTATIONAL COMPLEXITY ANALYSIS

We describe the computational complexity of our methodology as follows. Our methodology consists of three key steps. The first step is to calculate the document embedding for the job offer. Obtaining the word2vec representation of each word does not require any computation since this requires only fetching the corresponding vector from the memory. Computing the sum of the embedding vectors of r words representing the title (resp. k words representing the description) thus requires a computational cost of rn additions (resp. kn additions), where n is the number of dimensions in the embedding vectors. Computing the weighted sum of these resulting two sums requires $2n$ multiplications and n additions. For the second step, which is concerned with sector prediction using BERT, the computational complexity is, ignoring the number of additions, given by $O((r+k)^2m+(r+k)m^2)$ where m is the dimension of the word embedding generated by BERT (which is not necessarily equal to n) [53]. Finally, the last step is to compute the distance using the WMD algorithm between the job offer and all the occupations in the predicted sector, which represents a computational complexity of zn multiplications, where z is the number of occupations within the predicted sector.

The overall complexity for the processing of each job offer is therefore $O(n) + O((r+k)^2m + (r+k)m^2) + O(zn)$ multiplications. The number of words in the title and description, as well as the number of occupations in the predicted sector, have an impact on the computational complexity of our methodology. However, these values vary in a predefined range since the title of the job offer is a maximum of 6 words, and the words extracted from the description contain a maximum of 25 words. The maximum number of occupations in a sector, z, is 111. Hence, the complexity of processing a job offer is not high. Further, the task at hand does not require real-time processing so the time taken to process a job offer is not a concern.

V. EXPERIMENTS

In this section we give an overview about the data and experimental setup that is used to build our methodology.

A. DATASET & EVALUATION

To train our classifiers and evaluate the accuracy of our approach, we produced an annotated set of job ads selected from the corpus.

1) DATA ANNOTATION & PARTITIONING

In order to include all the sectors mentioned in the occupational directory, we performed a keyword search on our job ads dataset. This was done using words related to sectors such as industry, health, tourism, communication and others. This step allowed us to create a data set that was then annotated.

To build our classifier, we manually categorized each job ad into one of 12 categories. These categories are related to the sectors identified in the occupation directory (see table 3). The annotation process was not complicated and the labels were easily identified since we consider that each job offer belongs to only one sector and that the chosen sector reflects the job offer and not the company's sector (e.g. in the case where a company in the automotive sector is looking for a software engineer, the sector of the job will be the business support sector and not the industry sector since they are looking for an IT profile). Moreover, to validate the quality of our annotations, the dataset was verified by 3 experts in the field of human resources who also confirmed our categorization. We resulted in a dataset where the number of examples for each class (i.e. sector) is between 169 and 315. Then, we extracted a balanced dataset of 2028 job ads by making the number of job ads for each class be equal to that of the least represented class in the dataset. Table 4 lists the total number of job ads belonging to each class.

This dataset was split into three parts: training, validation, and testing. The training set is used to train the model and make it learn the hidden features/patterns in the data. The validation set is a set of data, separate from the training set, that is intended to validate the performance of our model during training. The testing set is a separate set of data used to test the model after completing training. In this work, we divided our dataset into 60% for training, 20% for validation, and 20% for testing. Finally, we used the three sets to train the classifiers described in the previous section to predict the sector of a job ad.

Additionally, to evaluate the accuracy of our unsupervised learning approach in identifying the correct occupation, we needed an occupation labeled dataset. To do this, a team of three experts in the field of human resources performed manual tagging of the 2028-element dataset using the occupational directory. The experts were provided with the occupational directory and the sector-labeled dataset and were asked to identify the appropriate occupation from the list of occupations in the same sector (by occupation, we mean the third level of our job directory that contains 512 broad occupations).

For each job ad, the experts read the title and the description. If they reached a consensus on how to label the occupation, the labeled job ad was added to the occupation labeled dataset. Otherwise, the considered job ad was discarded. Since the occupation labeling task is difficult and time-consuming, the experts have only processed 1276 job ads out of 2028, 1245 of which were labeled successfully (consensus was not reached for 31 job ads). This resulted in a dataset of 1245 labeled job ads where each sector is represented by at least 103 examples. While some occupations were represented by many job ads in the dataset, others were not represented by any job ads because such job ads are generally not published online; this is the case for example for high-responsibility positions for which recruiters prefer to seek recommendations or approach potential candidates directly. It is worth noting that occupation labels were used only to evaluate the results of our unsupervised learning method and not for building the sector classifiers.

2) EVALUATION

After selecting the algorithm and building the classifier, its performance must be evaluated to estimate the ability of the classification model to classify the unseen data into the correct classes. We used different metrics to evaluate the performance of the sector classifier such as Accuracy, Precision, Recall and F-score and they are defined as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} = \frac{\text{True Positives}}{\text{Positive Predictions}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} = \frac{\text{True Positives}}{\text{Positive Observations}}$$

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} = \frac{\text{CorrectPredictions}}{\text{All observations}}$$

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision}+\text{Recall}}$$

TABLE 3. Labels of our dataset with some examples.

Label	Example of job ad titles
Agriculture and fishing, natural and green spaces	Tractor driver, gardener, agricultural engineer, veterinarian
Banking, insurance & real estate	Bank account manager, real estate account manager, insurance customer advisor, financial analyst
Business support	Java developer, system administrator, payloader, accountant, human resources manager
Communication and media	Traffic manager, graphic designer, translator, copywriter
Construction & civil engineering	Plumber, site manager, civil engineering project manager, construction technician
Health	Nurse, ambulance driver, kinesitherapist, dental assistant, emergency physician
Hotel, catering & tourism	Head cook, housekeeper, waiter, receptionist, hotel manager
Human and community services	Teacher, lawyer, officer, senior trainer
Industry	welder, production manager, carpenter, electrician
Installation and maintenance	Quality control technician, mechanical maintenance engineer, maintenance project manager
Trade and sale	Telemarketer, sales engineer, salesman, cashier
Transport and logistics	Store manager, driver, logistics manager, warehouse manager

where TP, FP, FN and TN refer to True Positive, False Positive, False Negative and True Negative, respectively. These metrics have been used in prior work [8], [47]. Finally, to evaluate the performance of the title identification algorithm we rely on the overall accuracy which was used in previous work [1], [18], [45].

TABLE 4. The number of job ads in each class.

Class	Number of job ads
Agriculture and fishing, natural and green spaces	169
Banking, insurance & real estate	169
Business support	169
Communication and media	169
Construction & civil engineering	169
Health	169
Hotel, catering & tourism	169
Human and community services	169
Industry	169
Installation and maintenance	169
Trade and sale	169
Transport and logistics	169
Total	**2028**

B. EXPERIMENTAL SETTINGS

In our experiments, to create word embeddings for words and sentences we used the W2V model pre-trained on Wikipedia in French [38] where the dimensionality of the embedding is 500. So, our document representation (job title, job description, occupation title, and occupation description) will also have a 500 dimension.

For the classification task, we considered common supervised learning and self-supervised learning methods. We examined Multinomial Naïve Bayes, SVM, Logistic Regression and BERT. For Naïve Bayes, SVM, and Logistic Regression, we used the Scikit-Learn implementation [54] where pipelines were constructed for each model and TFIDF was used for feature representation. Then, a grid search with 5-fold cross-validation was performed over each classifier hyperparameter range to identify the hyperparameter combinations maximizing classification efficiency (using the training and validation sets). Table 5 represents the optimal hyperparameters obtained by the grid search.

TABLE 5. List of optimal hyperparameters that give the best results in terms of prediction accuracy and that were used to perform the final tests. Hyperparameters not mentioned have been set to their default values.

Algorithm	Parameters	Value
Naïve Bayes	Alpha	1.0
	Fit_prior	True
	Class_prior	None
Support Vector Machine	Kernel	linear
	C	10
	Penalty	l2
	Degree	3
	Max_iter	5
Logistic Regression	C	1
	Solver	lbfgs
	Penalty	l2
	Max_iter	100

For BERT, we utilized CamemBERT [51], the HuggingFace implementation of the pre-trained RoBERTa architecture [55]. In this paper, we used RoBERTa-base due to the GPU RAM constraint. Following common practice for fine-tuning pre-trained language models for classification [56], [57], we consider the $[CLS]$ token to be the final representation of the input data. During all the fine-tuning tests, we used the AdamW optimizer with a dropout rate of 0.1. We optimized the hyperparameters on the validation set by testing several values for learning rate (lr) and batch size. The values tested for $lr \in \{1e\text{-}3, 1e\text{-}4, 1e\text{-}5, 1e\text{-}6\}$ and batch size $\in \{8, 16, 32, 64\}$. The simulations show that the model with the highest test accuracy in all experimental settings relies on the hyperparameter combination $lr = 1e\text{-}5$, and batch size = 16. Finally, the model was trained for 15 epochs, and then we took the highest model checkpoint based on the validation set which is 8.

The experiments are performed using an Intel Core i7-10710 with 16 GB of memory installed, and a Windows 10 operating system. The algorithms are implemented by python 3.7.

VI. RESULTS

After conducting the experiments, we present in the following the results of our evaluations.

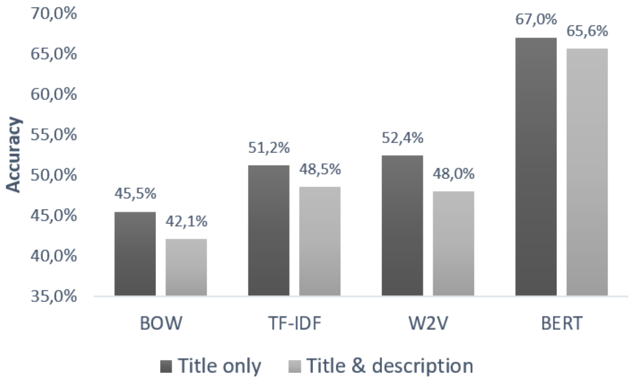

FIGURE 3. Performance outcomes related to accuracy. Here, the x-axis refers to the methods used for word embedding (with $\lambda_i = 1$ and $\alpha = \beta = 0.5$), and the y-axis indicates the accuracy.

A. COMPARAISON OF THE BOW, TFIDF, W2V AND BERT MODELS

To get a performance overview of document embedding methods, we first compare TFIDF, Word2vec and BERT with the baseline (Bag of word representation). More specifically, for the W2V approach, we use a uniform weighting strategy (Average $\lambda_i = 1$) for all the documents (the job ads and the occupations). Regarding document representation with BERT, we use raw vectors generated with the pre-trained model that already has weights. In addition, we consider that the titles (V(O$_t$) and V(J$_t$)) and the description (V(O$_d$) and V(J$_d$)) contribute equally to forming the final document representation which is equivalent to having $\alpha = \beta = 0.5$ in equation (2) and (3). The whole content of the description was used for both the job ad and the occupation description.

Figure 3 shows the accuracy of our methodology based on the different document embedding methods applied to both the title only and the combination of the title with the description. BOW (baseline), TFIDF, W2V and BERT scores are 42.1%, 48.5%, 48% and 65.6% respectively. Using BERT for document embedding significantly improved the accuracy of our methodology compared to W2V and TFIDF. The reason is that BERT gives a better representation of the document by generating the word embedding and the weights based on the context of the word, while our baseline model uses the average to calculate the document embedding.

In the meantime, using TFIDF and W2V for document representations gives almost the same accuracy. High hopes were placed on W2V since TFIDF doesn't support synonyms and doesn't preserve semantic relationships between words. After looking at our labeled data, we can explain this situation by the difference in vocabulary. This is because W2V was not able to give an embedding for some words that were not part of its vocabulary. However, TFIDF can deal with them since it relies on the word count. In the same context, applying lemmatization while processing the text generates words that are not in the W2V vocabulary (e.g., lem(inforgraphiste) = infographist) which explains the low accuracy of the W2V model. As mentioned in the methodology section (sec IV), we tried to apply our pre-trained W2V model but it did not improve the accuracy since we did not have enough data. Moreover, using lemmatization helped us increase accuracy by 2%. We also noticed that using the W2V model gives a more precise normalized title regarding the fourth level of our job directory. Thus, in future experiments, we will focus on improving accuracy by utilizing W2V and BERT and we will not include the BOW and TFIDF models in the comparison.

During our evaluation process, we noticed that including the occupation description in the document representation, helped us to identify the occupation for some job ads that are not normalized when using the title alone. This is done by adding more context to the title, which is similar to the results obtained in [8] where they identified that about 30% of the job ads' titles do not carry enough information to identify the occupation. For example, the word "electrician" can be found in both the industry and the installation and maintenance sectors. The same applies to the French word "Plongeur", which has two meanings: diver or dishwasher. However, Figure 3, shows that including the description negatively impacts the accuracy of our methodology. In fact, the description is a long text that usually contains words that are not informative which may lead to losing the meaning of the title. Therefore, in future experiments, we will try to optimize accuracy by finding ways to include the description with a minimal impact on the title.

B. THE EFFECT OF WEIGHTING STRATEGIES

Figure 4 summarizes the performance of the different weighting strategies (λ_i) used to calculate the document representation as discussed in the methodology section for equation (1) when applying W2V. We can see that when we use the title only, the highest accuracy was found with the uniform weighting strategy followed by the frequency weighting strategy. In contrast, when we included the description in the document representation, the top position was found by applying the TFIDF weighting strategy followed by the uniform weighting strategy. We explain this situation with the size of the document. The titles (V(O$_t$), V(J$_t$)) are short phrases that are intended to be highly informative, and usually consist of unique words, so their frequency is equal to 1, implying that the TFIDF weighting strategy is not relevant. On the other hand, the job description and the occupation description are long texts with a lot of noise. These texts need to be reduced by giving more importance to relevant words, which can be done by applying TFIDF. So in the next experiments, we will apply a uniform weighting strategy for the titles ($V(O_t)$, $V(J_t)$) and a TFIDF weighting strategy for descriptions ($V(O_d)$ and $V(J_d)$) before generating the document representation.

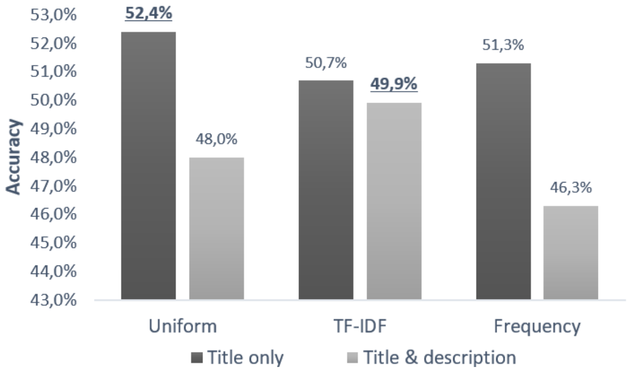

FIGURE 4. Performance outcomes related to accuracy. Here, the x-axis refers to the methods used for calculating the document embedding using different 3 weighting strategies (with W2V for word embedding and $\alpha = \beta = 0.5$), and the y-axis indicates the accuracy.

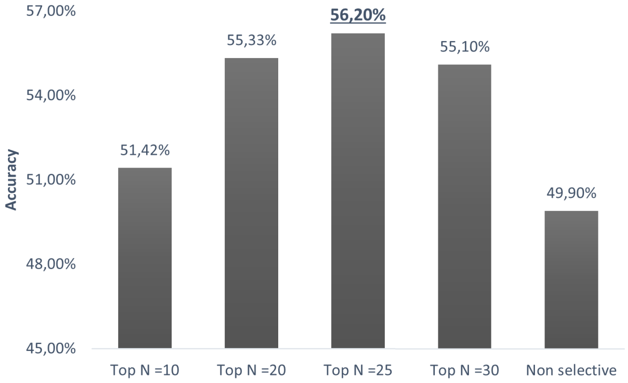

FIGURE 5. Performance outcomes related to accuracy. Here, the x-axis refers to N the number of relevant words extracted using the Chi-square method (with W2V for word embedding, $\lambda_i = 1$ for the titles, λ_i is calculated using TFIDF for the descriptions and $\alpha = \beta = 0.5$), and the y-axis indicates the accuracy.

C. INFLUENCE OF NOISE REMOVING

As discussed in the previous results, the description gives extra content to the job title by providing more information to improve the accuracy of the model. However, in many cases even after removing information related to the company, the description can still contain irrelevant general information such as marketing pitches and mission statements. This can negatively affect accuracy. To remedy this situation we applied the same concept used in weighting strategies giving the assumption that words with higher weight are more relevant to the job ad. We opted to get rid of the noise by ranking the words in the description based on their weight and discarding the least relevant. We applied the Chi-Square (CHI) technique to generate the weight of the words combined with the values of λ_i. Figure 5 shows the accuracy of top selective words with N = [10,20,25 and 30] compared to the whole description. We can see that the highest accuracy was achieved with 25 top words from the description with an improvement of 6.3% to achieve 56.2%. Values less than 25 also improved accuracy, but we expected to find a value of N between 5 and 15, as this range is sufficient to add context to the title. A brief explanation of this situation is that the list of top words contains some words that have been highly ranked even though they are not relevant to the matching process. This problem can be solved by cleaning the generated list.

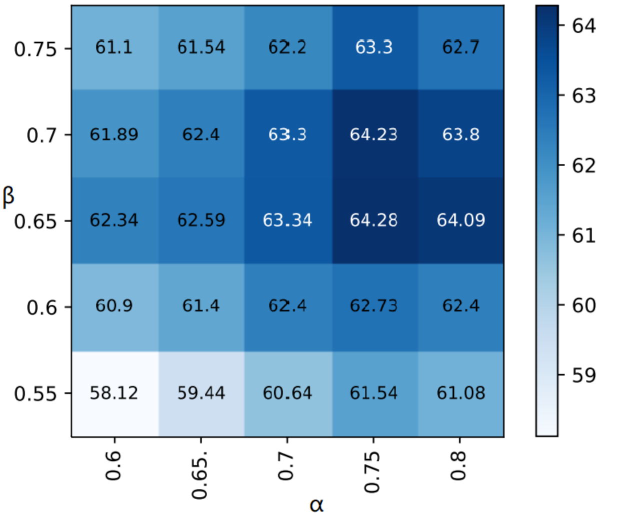

FIGURE 6. Accuracy of different combinations of α and β with W2V for word embedding, $\lambda_i = 1$ for the titles, λ_i is calculated using TFIDF for the descriptions and N = 25 (the darker spots indicate the best combinations, the accuracy is expressed in percentage).

D. INFLUENCE OF BALANCE FACTORS

As the title usually contains dense words related to the job, and the description provides more context to the title, their contribution to creating the document representation will differ. In equations (2) and (3), two balance factors α and β are involved in forming the vector of the occupation ($V(O)$) and the vector of the job ad ($V(J)$) respectively. To obtain the most accurate document representation, we experiment with different combinations of these values. We use uniform weighting for the titles and TFIDF weighting for the top 25 words selected from the description. Figure 6 shows the heat map related to the selected parameters. The blue squares indicate the accuracy improvement of our methodology. The darker the better. In these optimal conditions, we found that $\alpha = 0.75$ and $\beta = 0.65$ give the highest accuracy. This shows an improvement of 8% compared to the W2V, increasing from 56.2% to 64.28%. In other words, the occupation title is more relevant than the job ad title, and the job ad title relies more on the description to add more context and to achieve better accuracy. Furthermore, we can say that the occupation's title is more relevant than the occupation's description. In addition, the job ad title is more relevant than the job ad description. However, using both is more interesting than using the title alone.

After conducting several experiments with W2V for document representation, we found the optimal parameters that helped us improve accuracy by 16.28%. We tested our methodology using BERT for document representation on the title and on the 25 words from the description in addition to choosing $\alpha = 0.75$ and $\beta = 0.65$. The accuracy slightly increased by 2.1% to achieve 68.7%. We noticed that the performances were not as good as when we used W2V. We can explain that by the fact that BERT is powerful at generating weights for words and that the step of including the balance factors was the one that contributed the most to increasing the accuracy of BERT. At this step, we can say that the use of BERT for document representation outperforms the use of W2V in our methodology.

E. THE EFFECT OF SECTOR PREDICTION

1) SECTOR PREDICTION

As we saw in the previous results, our methodology needed to include a description to add more context to the title. Nevertheless, even with this manipulation, our methodology sometimes had trouble selecting the right occupation for the job advertisement, since it was not able to search in the right field. The intent here is to improve the quality of prediction in close boundary decisions. Where comes the need to add the step of sector prediction to predict the sector of the job ad that will be used as a feature in our methodology. Table 6 shows the performance of different models used for classification. We can see that BERT outperforms the other models with an accuracy of 94%. More specifically, Table 7 shows the accuracy rate by sector.

TABLE 6. Performance evaluation of the different text classification techniques with the optimized hyperparameters. Accuracy, Recall, and F1-Score values are the average of the values computed for each sector.

Model	Accuracy	Recall	F1-score
SVM	90%	91%	90%
Naïve Bayes	70%	64%	67%
Logistic Regression	91%	91%	91%
BERT	**94%**	**94%**	**95%**

2) JOB TITLE IDENTIFICATION BY SECTOR

After applying the classifier, we grouped the job ads by sector to perform our methodology and compare each job ad with occupations from the same sector. To do that, we used the optimal parameters identified previously; W2V for word representation, uniform weighting for the titles, TFIDF weighting for the 25 top words selected from the descriptions, and α, $\beta \in [0.6,0.8]$. Figure 7 shows the heat map related to the selected parameters. We can see an enhancement of the accuracy by 14% from 64.28% to 76.5%. Again, we found the same values for α, β which are 0.75 and 0.65 respectively. However, this time we notice that the dark spots are on the left side of the optimal value rather than the right side as shown in figure 6. This means that adding the sector prediction slightly decreases the degree of importance of the description.

To explore and better understand our results, table 8 shows details about the accuracy of each sector separately. The values range from 63,6% to 88,7%. The highest accuracy

TABLE 7. Prediction accuracy for each sector using BERT.

Sector	Accuracy
Agriculture and fishing, natural and green spaces	90,3%
Banking, insurance and real estate	98,3%
Business support	94,6%
Communication and media	97,8%
Construction & civil engineering	96,9%
Health	98,1%
Hotel, catering and tourism	91,1%
Human and community services	90,1%
Industry	94,3%
Installation and maintenance	98,7%
Trade and sale	90,7%
Transport and logistics	93,2%

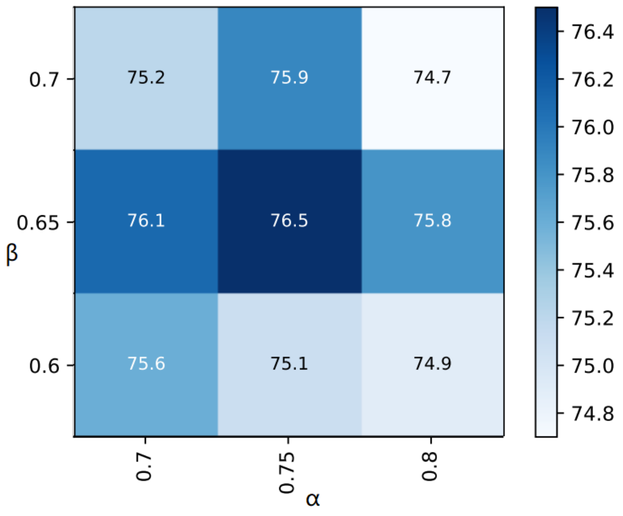

FIGURE 7. Accuracy of different combinations of α and β after using BERT for sector prediction and using W2V for word embedding, $\lambda_i = 1$ for the titles, λ_i is calculated using TFIDF for the descriptions and N = 25.

TABLE 8. Performance outcomes related to the accuracy of our methodology for job title identification for each sector using the optimal parameters (using BERT for sector classification, W2V for word embedding, $\lambda_i = 1$ for the titles, λ_i is calculated using TFIDF for the descriptions, $\alpha = 0.75$, $\beta = 0.65$ and $N = 25$).

Sector	Accuracy
Agriculture and fishing, natural and green spaces	78,8%
Banking, insurance and real estate	70,7%
Business support	75,2%
Communication and media	81,4%
Construction & civil engineering	83,8%
Health	88,7%
Hotel, catering and tourism	85,4%
Human and community services	69,7%
Industry	63,6%
Installation and maintenance	68,9%
Trade and sale	73,2%
Transport and logistics	76,74%

was achieved in the health sector, while the lowest accuracy was found in the industry sector. We can explain this difference in results by the number of occupations that exist at the third level of our occupational classifier. For example, the health sector and the hotel and tourism sector contain 29 and 38 occupations, respectively, while the industry sector contains 111 occupations. This makes the task of identifying occupations difficult, especially when the occupations are nearly similar.

Considering these results, we tested our methodology again after sector prediction using BERT for document representation. The accuracy also increased by 6.1% to achieve 74.8%. In this case, we can see that W2V outperforms BERT. Unfortunately, our expectations were not entirely met. After looking at the results of both models we explain this situation by the difference in vocabulary between the training dataset for BERT and the job ads. However, BERT performed well on job ads with a typo in the title or that contain an abbreviation such as human resources (HR).

VII. COMPARISON

To compare the results of our methodology with previous work, we implement and report the results of some methodologies in the related work section. We focused on unsupervised methods or methods with minimal interaction due to the lack of labeled datasets (most of the recent methods are based on classification and require hundreds of thousands of labeled data to train the models). We replicated the methodology described in [17] on a dataset related to the IT sector and the methodology described in [16] on a dataset related to the automotive sector. To do so, we created two small datasets. The first one was built from the annotated set of job ads by applying a filter to select IT-related job postings, resulting in 234 job ads. The second one was also built from the annotated set of job ads using a filter to select job ads related to the industry sector. Then, a manual classification was performed to keep only job ads related to the automotive sector. Regarding the occupation directory, we opted to use the job directory given by The Ministry of National Education, Vocational Training, Higher Education and Scientific Research [58] to show to what extent our methodology is easy to replicate on data from other countries.

The methodology described by Neculoiu et al. [16] applies siamese recurrent architectures to learn sentence similarity before classification by feeding similar and dissimilar job titles into the system. We use it only for titles. After preposessing the titles (As described in section IV.C), we build a word embedding vector supplemented with synonymic information for the LSTMs. Then we use the Manhattan distance [59] to force the model's representations learned from the sentences to form a structured space whose geometry reflects the semantic relations. Then we apply the model to classify job titles into predefined occupations. While the methodology described by Karakatsanis et al. [17] uses LSI to capture the hidden semantic relationships between documents in order to match the job ads with the occupation. Authors used TFIDF for word representation for both the job ads and the occupations and they included the job ad description in addition to the title.

Table 9 shows the results of the comparison of our methodology with the LSI method and the LSTM method on both datasets. For IT job ads, we can see that our methodology offers better accuracy since it achieved 85% as opposed to 81%. For automotive job ads, our methodology did not perform as the LSTM method. This result was expected since the findings of our methodology showed low accuracy in the industry sector due to the high number of occupations within this sector. However, our results are still satisfying because our methodology requires very few interactions, whereas the LSTM methodology needs hundreds of thousands of examples to train it, which is time and resource intensive. The comparison also shows that the LSTM method was weak at identifying the correct occupation. This was because the title was not very specific or very different from the training data, whereas our methodology showed a significant improvement since it included the description of the job ad.

VIII. CASE STUDY: APPLICATION ON THE MOROCCAN JOB MARKET

A. OVERVIEW

Our methodology aims to identify the right occupation for each job ad in order to better understand the needs of the job market. To get an overview of the Moroccan job market, we used 248,059 job ads collected from various websites between January 2019 and June 2021. Before applying our methodology, we processed our job ads by performing deduplication using the Simhash algorithm [60]. Then,

TABLE 9. Comparison of performance outcomes related to accuracy between our methodology and two other methodologies using two datasets.

Dataset	Methodology	Accuracy
IT job ads	Our proposed method	**85%**
	Karakatsanis et al. [17]	81%
Automotive job ads	Our proposed method	74%
	Neculoiu et al. [16]	**79%**

we followed the same steps described in the methodology section to classify the job ads into their corresponding sector. Figure 8 shows the number of job ads by sector during the defined period. We can clearly see that the business support and trade & sales sectors are the most recruiting sectors. These sectors contain occupations related to offshoring such as IT occupations and call center occupations that are highly demanded on the Moroccan job market [14]. Moreover, the industry sector has noticed an increase in the number of job offers due to the implementation of new factories related to the automotive sector [61], [62].

We can also see a remarkable decrease in demand for all sectors during two periods. The first took place during the second quarter of 2020. This is because Moroccan authorities announced a national lockdown from the end of February to the end of May 2020 as a result of Covid. The second one was during the fourth quarter of 2020 because there were rumors about a second lockdown due to the increase in Covid cases. In this context, we can notice that the only sector that was not impacted by Covid was the transport and logistics sector. This is due to the emergence of delivery companies who were looking for delivery agents.

TABLE 10. Ranking of top occupations within the IT occupation.

Rank	Occupation	# of job openings
1	Programmer	1643
2	Technical consultant	1108
3	Software engineer	801
4	Web integrator	707
5	Cybersecurity	615
6	Networking/Telecommunication	556
7	Information system consultant	378

B. RESULTS ON THE IT OCCUPATIONS

In this section, we chose to focus on the IT occupations included in the comparison subsection because they represent most job ads within the business support sector. We applied our methodology using the optimal parameters to get the list of the most in-demand occupations within IT occupations. By looking at table 10 we can see that occupations related to programming and software are highly in demand since programmers come in first place with a percentage of 11% from the total number of job ads followed by technical consultants and software engineers. The results of these findings were also found by the authors of [14], where they report that analyst&developer, project manager, and network manager are the most demanded occupations. According to these findings, the Moroccan job market has its own characteristics that set it apart from the world ranking, and universities should consider this when building curricula and training programs.

TABLE 11. Emerging occupations: comparison between the number of job ads by year.

Occupation	# of ads in 2019	# of ads in 2020	# of ads in 2021 (first 6 months)
Data scientist	37	81	47
Webmaster	101	206	120
Data center responsible	23	45	22
IT project director	180	257	175

Moreover, we can learn valuable insights from the previous results by filtering the number of job ads by year to identify emerging occupations. This allows us to detect the variation in the demand for some occupations as shown in table 11. For example, we have the data scientist occupation which has doubled the number of jobs opening in one year. The same is true of the webmaster position. This position has been in high demand during the Covid period as most companies are beginning to recognize the importance of having an online presence. Next is the data center responsible, which has increased by 1.95 times. Identifying emerging occupations enables us to inform students about future job market needs. In this way, they can choose a career path that leads to employment and become better prepared.

The application of job title identification is not only limited to finding cutting-edge or emerging occupations but it can also be used to study the degree of mismatch between labor market supply and demand [63]. In addition, we can conduct more in-depth studies by exploring the demand for hard and soft skills in all occupations [64] instead of focusing on specific fields (e.g., IT [21], [65], Automotive sector [62]).

IX. DISCUSSIONS

The paper investigates a new methodology for occupation identification in the case of a lack of labeled data. While most studies consider this problem as a classification problem, where the model is trained on hundreds of thousands of labeled data. This paper computes similarity measures to identify the most appropriate occupation. In fact, using classification algorithms is not easy, especially when we have

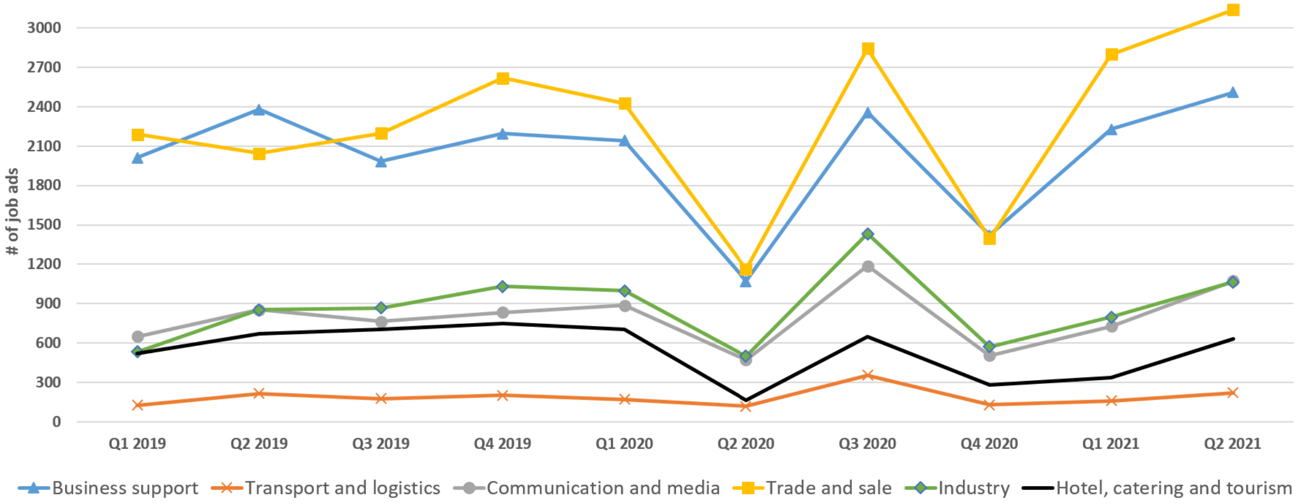

FIGURE 8. Job market trends by sector between January 2019 and June 2021 (only the 6 sectors that experienced major changes during the COVID period are shown). Here, the x-axis refers to the semesters, and the y-axis indicates the number of job ads.

a large number of classes (512 occupations). These papers usually use two-stage classifiers to simplify the problem and to reduce the number of classes [1], [18]. From them, we have adapted our model to first perform classification by fine-tuning BERT to predict the sector, and then to rely on similarity for the rest.

When tested, our proposed methodology either outperforms or performs at least as good as the state-of-the-art methods. Specifically, our approach allows the model to focus on the job title while also including the job description. Other methods from previous studies worked well when using the title only. However, as discussed in the comparison section (section VII), when the title is not complete or too general (which represents more than 40% of our dataset since we are working on unstructured data), they end up making errors. This study shows that including both the title and the description in the matching process with a defined ratio can help us improve the accuracy of the model. Furthermore, including all the information from the description can bring negative gains to the model, because the title is a dense sentence while the description is a long text. Thus, a step of feature selection is necessary to find and select the optimal number of words from the description. In this context, recent approaches can be used to improve our methodology such as Dual Regularized Unsupervised Feature Selection Based on Matrix Factorization and Minimum Redundancy (DR-FS-MFMR), which would allow us to discard redundant features from the set of original features [13].

While on the one hand, many of the papers we included in our study did not give sufficient attention to document representation as they relied on BOW or TFIDF representation which are not suitable for capturing synonyms or preserving semantic relationships. However, the results of the experiments show that using appropriate embedding can help us improve accuracy by 23%. The experiments also show that applying state-of-the-art algorithms for word embedding, in our case BERT, is not sufficient and that we need to use a model that is pre-trained on data related to job ads.

Finally, our methodology demonstrates its effectiveness in being easy to replicate on data from other countries or by choosing a different occupational directory since we used the French directory ''Pole Emploi'' to analyze the results of the experiments and used the Moroccan job directory to make the comparison with other methodologies. This situation is not trivial when considering the job title identification as a classification problem since the model is already pre-trained on a specific dataset using a predefined occupation directory.

X. CONCLUSION AND FUTURE WORK

In this paper, we present a two-stage job tile identification methodology based on semi-supervised and unsupervised machine learning algorithms with minimal labeling. In particular, for each job ad based on similarity measures, we find the most appropriate occupation using a standard occupational classifier. During the conducted experiments and after pre-processing the collected job ads, we tested several word and document representation methods such as TFIDF, neural language models that rely on distributional semantics (Word2Vec, FastText), and deep contextualized word representation (BERT). They were all subjected to several weighting strategies in order to reduce the impact of irrelevant words, especially in the description. Then, we tested various balance factors to identify the degree of contribution of both the title and the description to the process.

According to the experiment results, classifying the job ads by sector improved the accuracy of our methodology by 14% since the similarity measures between the job ad and the occupations will be applied only within the predicted sector instead of using all the occupations from the referential. For document representation, we found that results using W2V outperformed BERT since there is a difference in vocabulary between the training dataset and job vacancies. However, in the case where the sector is not specified, we found that BERT provides the most accurate results. When it comes to weighting strategies, results show that uniform and frequency word weighting work best for short text (job ad titles, occupation titles), as these are not sensitive to word weighting, while the TFIDF weighting strategy for long text (job ad descriptions, occupation descriptions) significantly improves performance. In addition, we found that document embedding using only the top N selective words from the description using weighting scores gives the most accurate results among all the configurations we tested since we add relevant context to the title. Finally, experiments also verify the effectiveness of using both the title and the description in the matching process. They also verify that we should not give them equal weights because the title is more relevant since it contains more dense words related to the job.

These findings helped us improve the accuracy of our methodology by 34% over the baseline. Our results - in terms of performance – are comparable to those obtained by the classification approach. Specifically, we obtained an overall accuracy of 76.5%, which can sometimes exceed 85% depending on the sector, such as the health sector and hotel & tourism sector. Furthermore, these findings can also be applied to improve the accuracy of the classifier when considering the task of job title identification as a classification problem.

Finally, this methodology can be replicated in other languages using other occupation classifiers with minimal interaction to normalize the job ads and get insights from them. The proposed technique has been tested in a real-life setting framed within the project called "Data science for improved education and employment in Morocco" supported by USAID which aims at analyzing the job market needs and extracting skills from them [4]. It can also be applied in the process of defining training courses by universities based on job market needs. At the same time, youth and job seekers looking for employment can benefit from the results of studies using this methodology to analyze the labor market.

In the future, we intend to add a step of job enrichment with skills terms based on the occupation description so that the job ad and occupation description are as similar as possible because recruiters do not follow a specific format when writing job advertisements. We also intend to do more cleaning of the list of top N words generated by weighting strategies to keep only relevant words. Furthermore, we plan to train our own Word2Vec model on sentences related to jobs in French, which may increase the accuracy of our methodology.

ACKNOWLEDGMENT

The authors would like to thank Mehdi Zakroum, Aymen Hafid, Imane Khaouja, and Ibtissam Makdoun for their useful comments and discussions.

REFERENCES

[1] F. Javed, Q. Luo, M. McNair, F. Jacob, M. Zhao, and T. S. Kang, "Carotene: A job title classification system for the online recruitment domain," in *Proc. IEEE 1st Int. Conf. Big Data Comput. Service Appl.*, Mar. 2015, pp. 286–293.

[2] M. S. Pera, R. Qumsiyeh, and Y.-K. Ng, "Web-based closed-domain data extraction on online advertisements," *Inf. Syst.*, vol. 38, no. 2, pp. 183–197, Apr. 2013.

[3] R. Kessler, N. Béchet, M. Roche, J.-M. Torres-Moreno, and M. El-Bèze, "A hybrid approach to managing job offers and candidates," *Inf. Process. Manage.*, vol. 48, no. 6, pp. 1124–1135, Nov. 2012.

[4] I. Rahhal, K. Carley, K. Ismail, and N. Sbihi, "Education path: Student orientation based on the job market needs," in *Proc. IEEE Global Eng. Educ. Conf. (EDUCON)*, Mar. 2022, pp. 1365–1373.

[5] S. Mittal, S. Gupta, K. Sagar, A. Shamma, I. Sahni, and N. Thakur, "A performance comparisons of machine learning classification techniques for job titles using job descriptions," *SSRN Electron. J.*, 2020. Accessed: Feb. 22, 2023. [Online]. Available: https://www.ssrn.com/abstract=3589962, doi: 10.2139/ssrn.3589962.

[6] R. Boselli, M. Cesarini, F. Mercorio, and M. Mezzanzanica, "Using machine learning for labour market intelligence," in *Machine Learning and Knowledge Discovery in Databases* (Lecture Notes in Computer Science), Y. Altun, K. Das, T. Mielikäinen, D. Malerba, J. Stefanowski, J. Read, M. Zitnik, M. Ceci, and S. Dzeroski, Eds. Cham, Switzerland: Springer, 2017, pp. 330–342.

[7] T. Van Huynh, K. Van Nguyen, N. L.-T. Nguyen, and A. G.-T. Nguyen, "Job prediction: From deep neural network models to applications," in *Proc. RIVF Int. Conf. Comput. Commun. Technol. (RIVF)*, Oct. 2020, pp. 1–6.

[8] F. Amato, R. Boselli, M. Cesarini, F. Mercorio, M. Mezzanzanica, V. Moscato, F. Persia, and A. Picariello, "Challenge: Processing web texts for classifying job offers," in *Proc. IEEE 9th Int. Conf. Semantic Comput. (IEEE ICSC)*, Feb. 2015, pp. 460–463.

[9] H. T. Tran, H. H. P. Vo, and S. T. Luu, "Predicting job titles from job descriptions with multi-label text classification," in *Proc. 8th NAFOSTED Conf. Inf. Comput. Sci. (NICS)*, Dec. 2021, pp. 513–518.

[10] R. Boselli, M. Cesarini, F. Mercorio, and M. Mezzanzanica, "Classifying online job advertisements through machine learning," *Future Gener. Comput. Syst.*, vol. 86, pp. 319–328, Sep. 2018.

[11] M. Vinel, I. Ryazanov, D. Botov, and I. Nikolaev, "Experimental comparison of unslupervised approaches in the task of separating specializations within professions in job vacancies," in *Proc. Conf. Artif. Intell. Natural Lang.*, Cham, Switzerland: Springer, 2019, pp. 99–112.

[12] E. Malherbe, M. Cataldi, and A. Ballatore, "Bringing order to the job market: Efficient job offer categorization in E-recruitment," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2015, pp. 1101–1104.

[13] F. Saberi-Movahed, M. Rostami, K. Berahmand, S. Karami, P. Tiwari, M. Oussalah, and S. S. Band, "Dual regularized unsupervised feature selection based on matrix factorization and minimum redundancy with application in gene selection," *Knowl.-Based Syst.*, vol. 256, Nov. 2022, Art. no. 109884.

[14] I. Khaouja, I. Rahhal, M. Elouali, G. Mezzour, I. Kassou, and K. M. Carley, "Analyzing the needs of the offshore sector in Morocco by mining job ads," in *Proc. IEEE Global Eng. Educ. Conf. (EDUCON)*, Apr. 2018, pp. 1380–1388.

[15] R. Bekkerman and M. Gavish, "High-precision phrase-based document classification on a modern scale," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2011, pp. 231–239.

[16] P. Neculoiu, M. Versteegh, and M. Rotaru, "Learning text similarity with siamese recurrent networks," in *Proc. 1st Workshop Represent. Learn. (NLP)*. Berlin, Germany: Association for Computational Linguistics, 2016, pp. 148–157. Accessed: Feb. 22, 2023. [Online]. Available: http://aclweb.org/anthology/W16-1617, doi: 10.18653/v1/W16-1617.

[17] I. Karakatsanis, W. AlKhader, F. MacCrory, A. Alibasic, M. A. Omar, Z. Aung, and W. L. Woon, "Data mining approach to monitoring the requirements of the job market: A case study," *Inf. Syst.*, vol. 65, pp. 1–6, Apr. 2017.

[18] Y. Zhu, F. Javed, and O. Ozturk, "Document embedding strategies for job title classification," in *Proc. 30th Int. Flairs Conf.*, 2017, pp. 55–65. Accessed: Oct. 4, 2022. [Online]. Available: https://www.aaai.org/ocs/index.php/FLAIRS/FLAIRS17/paper/view/15470

[19] F. Colace, M. D. Santo, M. Lombardi, F. Mercorio, M. Mezzanzanica, and F. Pascale, "Towards labour market intelligence through topic modelling," in *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, 2019, pp. 1–10.

[20] E. Mankolli and V. Guliashki, "A hybrid machine learning method for text analysis to determine job titles similarity," in *Proc. 15th Int. Conf. Adv. Technol., Syst. Services Telecommun. (TELSIKS)*, Oct. 2021, pp. 380–385.

[21] A. De Mauro, M. Greco, M. Grimaldi, and P. Ritala, "Human resources for Big Data professions: A systematic classification of job roles and required skill sets," *Inf. Process. Manage.*, vol. 54, no. 5, pp. 807–817, 2018.

[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 3111–3119.

[23] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," in *Proc. 11th Int. Conf. Lang. Resour. Eval. (LREC)*, Miyazaki, Japan, May 2018, pp. 1–4.

[24] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1155–1237, 2003.

[25] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proc. 25th Int. Conf. Mach. Learn.*, Helsinki, Finland, Jul. 2008, pp. 160–167.

[26] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[27] M. Baroni, G. Dinu, and G. Kruszewski, "Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2014, pp. 238–247.

[28] J. Mitchell and M. Lapata, "Vector-based models of semantic composition," in *Proc. ACL HLT*, vol. 56, 2008, pp. 236–244.

[29] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 1631–1642.

[30] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process. (Long Papers)*, vol. 1, 2015, pp. 1556–1566.

[31] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2014, pp. 655–665.

[32] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. 31st Int. Conf. Mach. Learn.*, Beijing, China, vol. 32, Jan. 2014, pp. 1188–1196.

[33] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1, Jun. 2019, pp. 4171–4186.

[34] P. Hoang, T. Mahoney, F. Javed, and M. McNair, "Large-scale occupational skills normalization for online recruitment," *AI Mag.*, vol. 39, no. 1, pp. 5–14, Mar. 2018.

[35] I. Rahhal, I. Makdoun, G. Mezzour, I. Khaouja, K. Carley, and I. Kassou, "Analyzing cybersecurity job market needs in Morocco by mining job ads," in *Proc. IEEE Global Eng. Educ. Conf. (EDUCON)*, Apr. 2019, pp. 535–543.

[36] *Pole Emploi Occupational Classifier*. Accessed: Feb. 11, 2021. [Online]. Available: https://www.poleemploi.fr/employeur/vos-recrutements/le-rome-et-les-fiches-metiers.html

[37] *URL Example From Pole Emploi That Contains the Details of the Occupation With the Code m1805*. Retrieved From [Online]. Available: https://candidat.pole-emploi.fr/marche-du-travail/fichemetierrome?codeRome=M1805

[38] C. Schöch, "A Word2Vec model file built from the French Wikipedia Xml dump using Gensim," Tech. Rep., Oct. 2016.

[39] M. Vinel, I. Ryazanov, D. Botov, and I. Nikolaev, "Experimental comparison of unsupervised approaches in the task of separating specializations within professions in job vacancies," in *Artificial Intelligence and Natural Language*, D. Ustalov, A. Filchenkov, and L. Pivovarova, Eds. Cham, Switzerland: Springer, 2019, pp. 99–112.

[40] E. O. Omuya, G. O. Okeyo, and M. W. Kimwele, "Feature selection for classification using principal component analysis and information gain," *Expert Syst. Appl.*, vol. 174, Jul. 2021, Art. no. 114765.

[41] A. Satorra and P. M. Bentler, "A scaled difference chi-square test statistic for moment structure analysis," *Psychometrika*, vol. 66, no. 4, pp. 507–514, Dec. 2001.

[42] O. Al-Harbi, "A comparative study of feature selection methods for dialectal Arabic sentiment classification using support vector machine," *Int. J. Comput. Sci. Netw. Secur.*, vol. 19, pp. 167–176, Jan. 2019.

[43] M. Y. Javed and H. Mohsin, "An automated approach for software bug classification," in *Proc. 6th Int. Conf. Complex, Intell., Softw. Intensive Syst.*, Jul. 2012, pp. 414–419.

[44] B. Al-Salemi, M. Ayob, G. Kendall, and S. A. M. Noah, "Multi-label Arabic text categorization: A benchmark and baseline comparison of multi-label learning algorithms," *Inf. Process. Manage.*, vol. 56, no. 1, pp. 212–227, Jan. 2019.

[45] P. Andrey and M. Bergeat, "Analyse des offres d'emploi en ligne: Comment coder Le métier," 13 eme Journées de méthodologie statistique de l'Insee (JMS), Jun. 2018. [Online]. Available: http://www.jms-insee.fr/2018/S25_3_ACTE_ANDREY_JMS2018.pdf and

[46] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2015, pp. 957–966.

[47] A. Omar, T. M. Mahmoud, T. Abd-El-Hafeez, and A. Mahfouz, "Multi-label Arabic text classification in online social networks," *Inf. Syst.*, vol. 100, Sep. 2021, Art. no. 101785.

[48] S. Stieglitz, M. Mirbabaie, B. Ross, and C. Neuberger, "Social media analytics—Challenges in topic discovery, data collection, and data preparation," *Int. J. Inf. Manage.*, vol. 39, pp. 156–168, Apr. 2018.

[49] K.-M. Schneider, "Techniques for improving the performance of naive Bayes for text classification," in *Computational Linguistics and Intelligent Text Processing* (Lecture Notes in Computer Science), A. Gelbukh, Ed. Berlin, Germany: Springer, 2005, pp. 682–693.

[50] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Machine Learning: ECML-98* (Lecture Notes in Computer Science), C. Nédellec and C. Rouveirol, Eds. Berlin, Germany: Springer, 1998, pp. 137–142.

[51] L. Martin, B. Müller, P. J. O. Suárez, Y. Dupont, L. Romary, É. de la Clergerie, D. Seddah, and B. Sagot, "CamemBERT: A tasty French language model," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 7203–7219.

[52] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*.

[53] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. U. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2017, pp. 1–15.

[54] *Scikit-Learn Machine Learning in Python*. Accessed: May 15, 2021. [Online]. Available: https://scikitlearn.org/

[55] *Huggingface Implementation*. Accessed: June. 21, 2021. [Online]. Available: https://huggingface.co/

[56] Y. Moukafih, M. Ghogho, and K. Smaili, "Supervised contrastive learning as multi-objective optimization for fine-tuning large pre-trained language models," 2022, *arXiv:2209.14161*.

[57] B. Gunel, J. Du, A. Conneau, and V. Stoyanov, "Supervised contrastive learning for pre-trained language model fine-tuning," 2020, *arXiv:2011.01403*.

[58] *List of Professions and Jobs by Sector Published by the Ministry of National Education, Vocational Training, Higher Education and Scientific Research*. Accessed: Dec. 20, 2021. [Online]. Available: https://dfp.gov.ma/component/content/article/307.html

[59] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed. San Mateo, CA, USA: Morgan Kaufmann, Univ. Illinois at Urbana Champaign, 2006.

[60] G. S. Manku, A. Jain, and A. D. Sarma, "Detecting near-duplicates for web crawling," in *Proc. 16th Int. Conf. World Wide Web*, May 2007, p. 141.

[61] A. Haddach, L. B. Allal, A. Laglaoui, and M. Ammari, "Moroccan automotive industry: Opportunities and perspectives," *Amer. J. Eng. Res.*, vol. 6, pp. 75–82, 2017.

[62] I. Makdoun, G. Mezzour, K. M. Carley, and I. Kassou, "Analyzing the needs of the automotive job market in Morocco," in *Proc. 13th Int. Conf. Comput. Sci. Educ. (ICCSE)*, Aug. 2018, pp. 1–6.

[63] I. Makdoun, I. Rahhal, G. Mezzour, I. Kassou, and K. M. Carley, "Skill mismatch evidence for cybersecurity skills in Morocco," *Proc. Comput. Sci.*, vol. 184, pp. 941–946, Jan. 2021.

[64] I. Khaouja, I. Kassou, and M. Ghogho, "A survey on skill identification from online job ads," *IEEE Access*, vol. 9, pp. 118134–118153, 2021.

[65] T. Janotik, "Current labor market trends in the area of information technology in Slovakia," in *Proc. 13th Int. Conf. Emerg. eLearning Technol. Appl. (ICETA)*, Nov. 2015, pp. 1–8.

IBRAHIM RAHHAL received the engineering degree in computer science from the Mohammadia School of Engineering (EMI), Rabat, Morocco, in 2016, with a major in information systems. He is currently pursuing the Ph.D. degree with Mohammed V University, Rabat, with co-direction of the International University of Rabat. His research interests include machine learning and natural language processing. He was a recipient of the Fulbright Fellowship, in 2019.

KATHLEEN M. CARLEY (Fellow, IEEE) received the S.B. degree from the Massachusetts Institute of Technology (MIT), the H.D. degree from the University of Zurich, and the Ph.D. degree from Harvard University. She is currently a Professor of societal computing with the Institute for Software Research, Carnegie Mellon University, the Director of the Center for Computational Analysis of Social and Organizational Systems (CASOS), the Director of the Center for Informed Democracy and Social Cybersecurity (IDeaS), and the CEO of Netanomics. Her research interests include computer science and social science for addressing complex real world issues, such as social cybersecurity, disinformation, disease contagion, disaster response, and terrorism, from a high dimensional network analytic, machine learning, and natural language processing perspective. She and her group have developed network and simulation tools, such as ORA, that can assess network and social media data.

ISMAIL KASSOU received the master's degree, in 1988, and the Ph.D. degree in computer science from the University of Rouen, in 1992. He was the Deputy Director in charge of research and cooperation of ENSIAS and the Director of the ST2I Doctoral Studies Center, Mohammed V University, Rabat, Morocco. He has a wide experience in project management and international cooperation. He has been a Professor with Mohammed V University, since 1993, where he is currently the Vice President of research and cooperation. He is the coauthor of two patents and about 63 scientific articles. His research interests include knowledge management, text mining, web mining, and information retrieval.

MOUNIR GHOGHO (Fellow, IEEE) received the M.Sc. and Ph.D. degrees from the National Polytechnic Institute of Toulouse, France, in 1993 and 1997, respectively. He was an EPSRC Research Fellow with the University of Strathclyde, Scotland, from September 1997 to November 2001. In December 2001, he joined the School of Electronic and Electrical Engineering, University of Leeds, England, where he was promoted to a Full Professor, in 2008. While still affiliated with the University of Leeds, in 2010, he joined the International University of Rabat, where he is currently the Dean of the College of Doctoral Studies and the Director of the ICT Research Laboratory (TICLab). He is also the Co-Founder and the Co-Director of the CNRS-Associated International Research Laboratory, DataNet, in the field of big data and artificial intelligence. He has coordinated around 20 research projects and supervised more than 30 Ph.D. students in the U.K. and Morocco. His research interests include machine learning, signal processing, and wireless communication. He is a fellow of Asia-Pacific AI Association (AAIA). He was a recipient of the 2013 IBM Faculty Award and the 2000 U.K. Royal Academy of Engineering Research Fellowship. He has served as an Associate Editor for many journals, including *IEEE Signal Processing Magazine* and IEEE Transactions on Signal Processing.