University of Pisa

School of Engineering

Master of Science in Artificial Intelligence and Data Engineering

Project documentation

Human Resources Automatic CV Classification

Work group:
Massimo Valentino Caroti
Simone Landi
Jacopo Cecchetti

Academic year 2021-2022

Contents

```
['hr' 'designer' 'information-technology' 'teacher' 'advocate'
 'business-development' 'health and fitness' 'agriculture' 'bpo' 'sales'
 'consultant' 'digital-media' 'automobile' 'chef' 'finance' 'apparel'
 'accountant' 'construction' 'public-relations' 'banking' 'arts'
 'aviation' 'data science' 'web designing' 'mechanical engineer'
 'civil engineer' 'java developer' 'business analyst' 'sap developer'
 'automation testing' 'electrical engineering' 'operations manager'
 'python developer' 'devops engineer' 'network security engineer' 'pmo'
 'database' 'hadoop' 'etl developer' 'dotnet developer' 'blockchain'
 'testing']
```

Figure 1: Job Categories

1 Introduction

A resume is still an important document and decision maker in evaluating the path of candidates. Its main role is to detect the suitability of people applying for job offers. This project aims to develop a system that automates the pre-selection of suitability and assessment of candidates in the recruitment process using NLP techniques and machine learning algorithms. This system will replace manual CV processing activities and provide accurate and effective assessment results. To meet this requirement, the system will be implemented using a machine learning approach with classification algorithms. The promising results suggest that NLP and ML techniques employed in this study could be used for developing an efficient Resume Classification System.

1.1 Automatic CV classification

The selection of a suitable job applicant from the pool of thousands applications is often daunting job for an employer. The classification of job applications submitted in form of resumes against available vacancy takes significant time and efforts of an employer. Thus, Resume Classification System using the Natural Language Processing (NLP) and Machine Learning (ML) techniques could automate this tedious process. Moreover, the automation of this process can significantly expedite the applicants' screening process with mere human involvement. This project presents an automated NLP and ML-based that classifies the resumes according to job categories with performance guarantees.

1.2 Dataset

For the project, we used two datasets found online in which there are more than 4000 complete resumes, for each resume there are: personal information, job for which he/she wants to apply, skills, description of past jobs and passions. The code and dataset can be found in the GitHub repository. The main jobs in the merged dataset are shown in *figure 1*.

Figure 2: Word Cloud

Figure 3: Word Cloud for each category

1.3 Word Cloud

Word Cloud is a data visualization technique used for representing text data in which the size of each word indicates its frequency or importance. Significant textual data points can be highlighted using a word cloud. For the project, we used a word cloud associated with all resumes to visualize the most frequent words in the resumes. We also generated a word cloud for each job with the most frequent and important keywords to identify that job.

1.4 Python Notebook

The code used for the automatic CV classification can be viewed at the following GitHub link: `https://github.com/max423/Automatic-CV-classification.git`

2 Process

2.1 NLP

2.1.1 Preprocessing

The stages of our NLP preprocessing are as follows

1. *Data cleaning.* We used regular expression in order to remove punctuation and special characters and substitute words that were spelled wrong in the original data set. We also turned all the characters into lowercase characters and removed extra spaces.

2. *Tokenization.* We broke a stream of textual data into the smallest units of a sentence called tokens.

3. *Filtering.* We removed stop words, words which are filtered out because they are insignificant.

4. *Part-of-speech tagging.* We used part-of-speech tagging, which is the process of marking up a word in a text as corresponding to a particular part of speech, based on both its definiton and its context.

5. *Lemmatisation.* We grouped together the inflected forms of a word into its lemma, the canonical form of a set of word forms.

2.1.2 Document similarity

After performing NLP preprocessing we wanted to check if resumes belonging to the same categories were similar to each other. We chose *Cosine similarity* to measure the similarity between the resumes. Cosine similarity is a measure of similarity between two sequences of numbers (the vectorized resumes in our case). The sequences are viewed as vectors and the cosine similarity is defined as the cosine of the angle between them. Cosine similarity gives a useful measure of how similar two documents are likely to be.

For each resume we computed

- the mean of the cosine similarity between the resume and all the other resumes

- the mean of the cosine similarity between the resume and all the resumes belonging to the same category

Then, for each category, we computed the mean of the two values. We can see from *figure 4* that resumes belonging to the same category have a much higher similarity, so we decided to proceed with the classification.

2.2 Classification

The goal of this project is to develop a classifier that is able to perform automatic CV classification. The classifier will be used by employment offices and postgraduate orientation centers so that the classifier will suggest a list of suitable jobs to the user based on his/her resume.

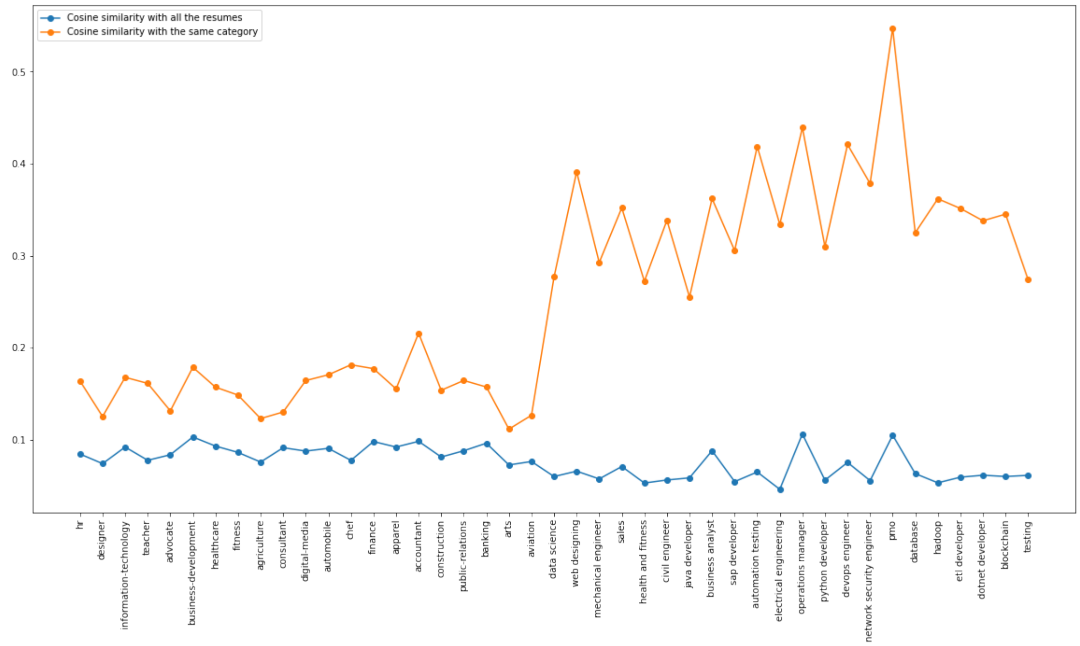

Figure 4: Cosine similarity

2.2.1 Text Vectorization: Term Frequency - Inverse Document Frequency (TFIDF)

TFIDF is based on the Bag of Words (BoW) model, which converts the text into a feature vector by counting the occurrence of words in a document. The **Term Frequency (TF)** is a measure of the frequency of a word in a document. TF is defined as the ratio of a word's occurrence in a document to the total number of words in a document.

$$\frac{\text{occurences of w in document d}}{\text{total number of words in document d}}$$

The **Inverse Document Frequency (IDF)** is the measure of the importance of a word. IDF provides weightage to each word based on its frequency in the corpus D.

$$\ln \frac{\text{total number of documents}}{\text{number of documents containing w}}$$

After applying TFIDF, text in resumes can be represented as a TFIDF sparse matrix of dimensions (number of documents) x (vocabulary words). The value corresponding to each word represents the importance of that word in a particular document.

2.2.2 Classifiers comparison

We evaluated different classifiers using the accuracy as evaluation metric and a K-Fold Cross Validation with K = 10 for a better accuracy result.

Considering this metric the decided to use the *Linear Support Vector* classifier

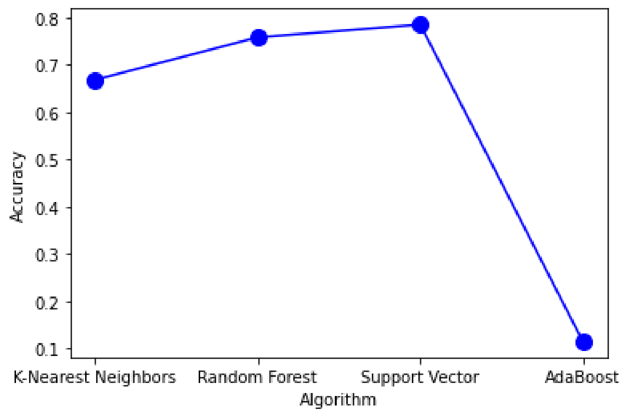

Figure 5: Cosine similarity

2.2.3 Classification remarks

Looking at *figure 6* we can see that the categories with the higher misclassified resumes are the ones that have the lowest similarity according to *figure 4*.

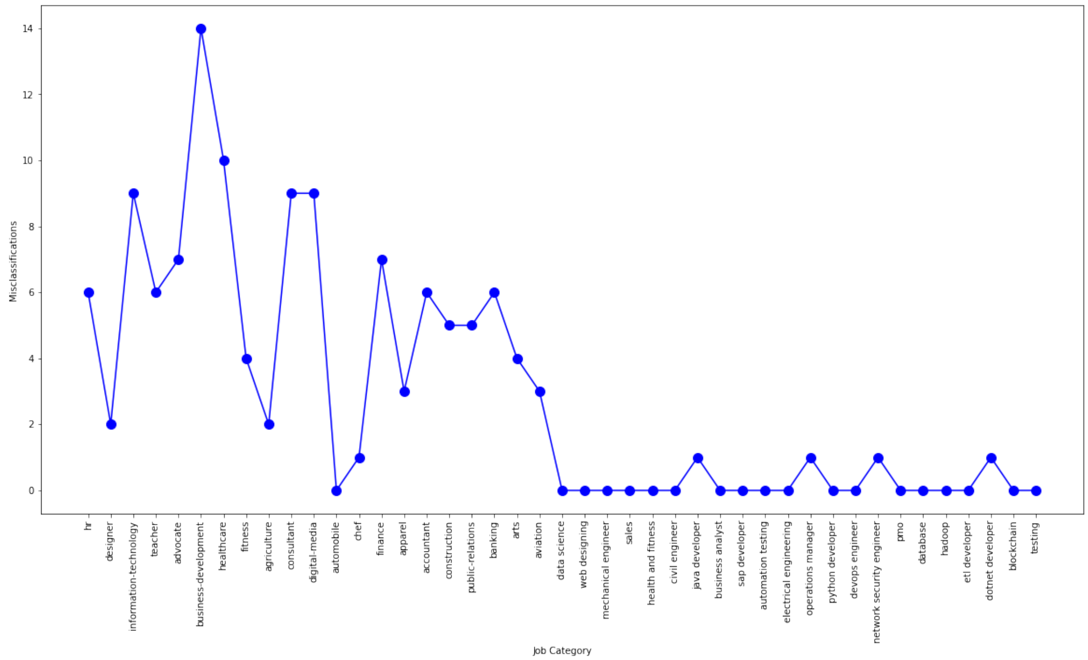

Figure 6: Misclassifications

3 Results

On the cleaned data, a model was built based on Classification. Based on the curriculum and category, the model was designed to classify the curriculum into the right category.

3.1 Classification

The classification was done using Linear Support Vector model: a supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. The Linear Support Vector classifier yielded an accuracy of 0.78 on 10-fold cross-validation. The confusion matrix obtained with our model is shown below, highlighting the discrepancies between predicted and actual labels.

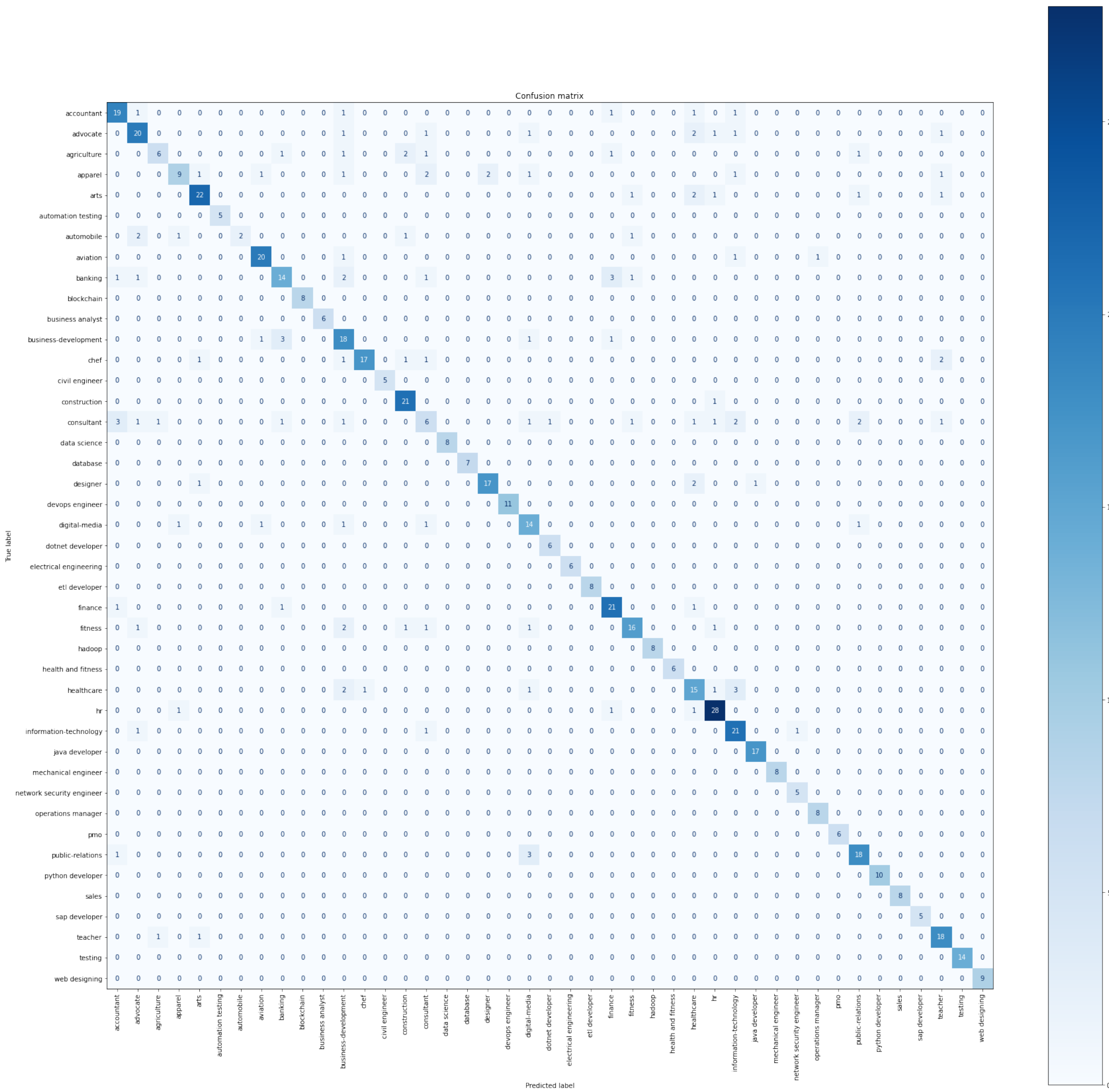

Figure 7: Confusion Matrix

3.2 Potential implications

The model would help recruiters speed up the selection of profiles, classifying CVs according to requirements and easily identifying the CVs that best match the job description. Furthermore, it would help the recruiter to reduce the resources used in identifying the right talent, making the process cost-effective. The recommendations made by the model are currently for various industries, but the model can be further improved to target specific industries, which would make it more effective and provide better recommendations.

3.3 Limitations

The model design has some limitations, which can be overcame by having more data for training the model. One limitation is the fact that the model accepts CVs in CSV format, but in the real world CVs are in .doc, .pdf, etc. A possible solution may be to use libraries that can read different file formats and convert them into a single format that can be used as input for the model. One possible upgrade could be to give an evaluation of resumes. In order to do this, it is necessary to avoid losing important information due to text compression caused by summarisation. It will be necessary to refine the summarisation process to ensure a minimum loss of information, such as candidates' years of experience.

3.4 Conclusion

The process of classifying resumes is manual, time consuming and wastes resources. The proposed model classifies the curriculum into different categories automatically. The proposed approach effectively captures the insights about the curriculum, their semantics and produced 0.78 accuracy with the Linear Support Vector classifier. Involving domain experts such as HR professionals would help to build a more accurate model; feedback from HR professionals helps to improve the model iteratively. This study has been conducted on the general topic of Resume Screening. More detailed results may be obtained from a company by conducting an analysis on their business area.