Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 167 (2020) 2318–2327

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Intelligence and Data Science (ICCIDS 2019)

A Machine Learning approach for automation of Resume Recommendation system

Pradeep Kumar Roy[a,*], Sarabjeet Singh Chowdhary[b], Rocky Bhatia[b]

[a]*Vellore Institute of Technology, Vellore, TN, India*
[b]*Adove, Noida, Uttar Pradesh, India*

Abstract

Finding suitable candidates for an open role could be a daunting task, especially when there are many applicants. It can impede team progress for getting the right person on the right time. An automated way of "Resume Classification and Matching" could really ease the tedious process of fair screening and shortlisting, it would certainly expedite the candidate selection and decision-making process. This system could work with a large number of resumes for first classifying the right categories using different classifier, once classification has been done then as per the job description, top candidates could be ranked using Content-based Recommendation, using cosine similarity and by using k-NN to identify the CVs that are nearest to the provided job description.

Keywords: Job Seekers; Resume Recommendation; Online Job Search; Resume Similarity;

1. Introduction

Talent acquisition is an important, complex, and time-consuming function within Human Resources (HR). The sheer scale of Indias market is overwhelming [2, 8, 14]. Not only is there a staggering one million people coming into the job market every month, but there is also huge turnover. As per LinkedIn, India has the highest percentage of the workforce that is "actively seeking a new job" [10]. Clearly, this is an extremely liquid, massive marketŁbut one that also has many frustrating inefficiencies. The most challenging part is the lack of a standard structure and format for resume which makes short listing of desired profiles for required roles very tedious and time-consuming [11, 24]. Effective screening of resumes requires domain knowledge, to be able to understand the relevance and applicability of a profile for the job role. With a huge number of different job roles existing today along with the typically large number of applications received, short-listing poses a challenge for the human resource department. Which is only further worsened by the lack of diverse skill and domain knowledge within the HR department, required for effective

* Corresponding author. Tel.: +91-8539035222
E-mail address: pkroynitp@gmail.com

1877-0509
10.1016/j.procs.2020.03.284

screening. Being able to weed out non-relevant profiles as early as possible in the pipeline results in cost savings, both in terms of time as well as money [29].

Today the industry face three major challenges:

- Separating right candidates from the pack - India being a huge job market and with millions seeking jobs; it is humanly impossible to screen the CVs and find the right match. This makes the whole hiring process slow and inefficient costing resources to the companies.
- Making sense of candidate CVs - Second challenges are posed by the fact that the CVs in the market are not standard practically every resume in the market has different structure and format. HR has to manually go through the CVs to find the right match to the job description. This is resource intensive and prone to error whereby a right candidate for the job might get missed in the process.
- Knowing that candidates can do the job before you hire them -The third and the major challenge is mapping the CV to the job description to understand if the candidate would be able to do the job for which she is being hired.

To overcome the mentioned issues in the resume short-listing process, in this paper we present an automated Machine Learning based model. The model takes the features extracted from the candidate's resume as input and finds their categories, further based on the required job description the categorised resume mapped and recommend the most suitable candidate's profile to HR. Our main contributions are listed below:

1. We developed an automated resume recommendation system.
2. Machine learning based classification techniques with similarity functions are used to find most relevant resume.
3. Linear SVM classifier performed best for our case compared to another ML classifiers.

Rest of the article organized as follows: section 2 describes the related works, the problem statement is stated in section 3. In section 4, we explain our proposed methodology, followed by results and conclusion in section 5 and 6 respectively.

2. Related Works

The number of job seekers are increasing with the time, every job receives a number of applications, and among them, many are relevant to the mentioned post. It creates a big problem for job recruiter as they have to shortlist the most eligible profile/resume from the pool of resume [2, 30, 23]. The process of matching the candidate resume with the job description is similar to a recommender system as the profile of the candidate recommended for a particular post. The recommendation system was introduced by Resnick and Varian [20]. The recommendation system is widely used in various domains nowadays including product recommendation on e-commerce portal [25, 27], book recommendation [16], news recommendation [6], movie recommendation [7], music recommendation [5], and many others [4, 13, 18, 29, 22].

Lu et al., [13] proposed a detailed survey which included the different protocols that were used by the researcher in the past few years for the recommendation system. They were discussed how the recommendation system widely used in real-time applications. A recommendation service is of mainly four types of Collaborative filtering, Content-based filtering, Knowledge-based and Hybrid approaches [28]. Wei et al., [28] discussed all different types of recommendation techniques with their working principle in detail. Al-Otaibi et al., [1] provided a detailed survey of job recommendation service. They discussed the steps involved in the recruiting process used by any organization. How the e-recruitment portal is helping to the organization, what factor of the candidate may lead to getting selected and many other relevant recruitment processes are explained. An Expectation Maximization (EM) algorithm was used by Malinowski et al., [15] for the job recommendation which considers both the applicant resume and the organizations job description. Whereas a fuzzy-based model used by Golec & Kahya [9] to evaluate the candidate relevancy with respect to the posted job description. Another model proposed by Paparrizos et al., [18] using a hybrid classifier. They used information retrieval, manual attributes and other for job recommending process.

Our work is different than that of earlier proposed systems, as in most of the existing system a job is recommended to the candidates based on their resume content, it leads a low classification accuracy. In order to improve it, we proposed a system which works in two phases: i) classifying the resume in their classes, and ii) ranking the candidate resume based on the job description and their resume content.

3. Problem Statements

Today the major problem being faced across the industry is how to acquire the right talent, using minimal resources over the internet and in minimal time. As described in section 1, there are three major challenges that are required to be overcome, to bring efficiencies to the complete process.

- Separating the right candidates from the pack
- Making sense of candidate CVs
- Knowing that candidates can do the job before you hire them

Our group intends to provide a solution to the above-mentioned challenges by automating the process. The solution would help to find the right CV from the large dumps of CVs; would be agnostic to the format in which CV has been created and would give with the list of CVs which are the best match to the job description provided by the recruiter [11]. The proposed solution involves supervised learning to classify the resumes into various categories corresponding to the various domains of expertise of the candidates. A multi-pronged approach to classification is proposed as follows:

- Perform NER, NLP, and Text classification using n-grams.
- Use distance-metric based classification.
- The solution shall provide a feedback loop closure to adjust/improve the accuracy by incorporating the feedback corresponding to the incorrectly screened profiles.

4. Methodology

The aim of this work is to find the right candidates resume from the pool of resumes. To achieve this objective, we have developed a machine learning based solution, The complete framework for the proposed model is shown in Figure 2. The proposed model worked in mainly in two steps: i) Prepare and ii) Deploy and Inference.

Dataset Description: The data was downloaded from the online portal(s) and from Kaggle. The data is in Excel format, with three column ID, Category, and Resume. **ID** - The sequence number of the resume, **Category** - Industry sector to which the resume belongs to, and **Resume** - The complete CV of the candidate. The number of instances for the different domain can be seen from Figure 1.

4.1. Preprocessing

In this process, the CVs being provided as input would be cleansed to remove special or any junk characters that are there in the CVs. In cleaning, all special characters, the numbers, and the single letter words are removed. We got the clean dataset after these steps having no special characters, numbers or single letter word. The dataset is split into the tokens using the NLTK tokenizes [12]. Further, the preprocessing steps are applied on tokenized dataset such as stop word removal, stemming, and lemmatization. The raw CV file was imported and the data in the resume field was cleansed to remove the numbers and the extra spaces in the date. Data Masking was done as:

- Mask string fragments like \x
- Mask string fragments for escape sequences like \a, \b, \t, \n
- Mask all numbers
- Replace all the single letter words with an empty string
- Mask email addresses

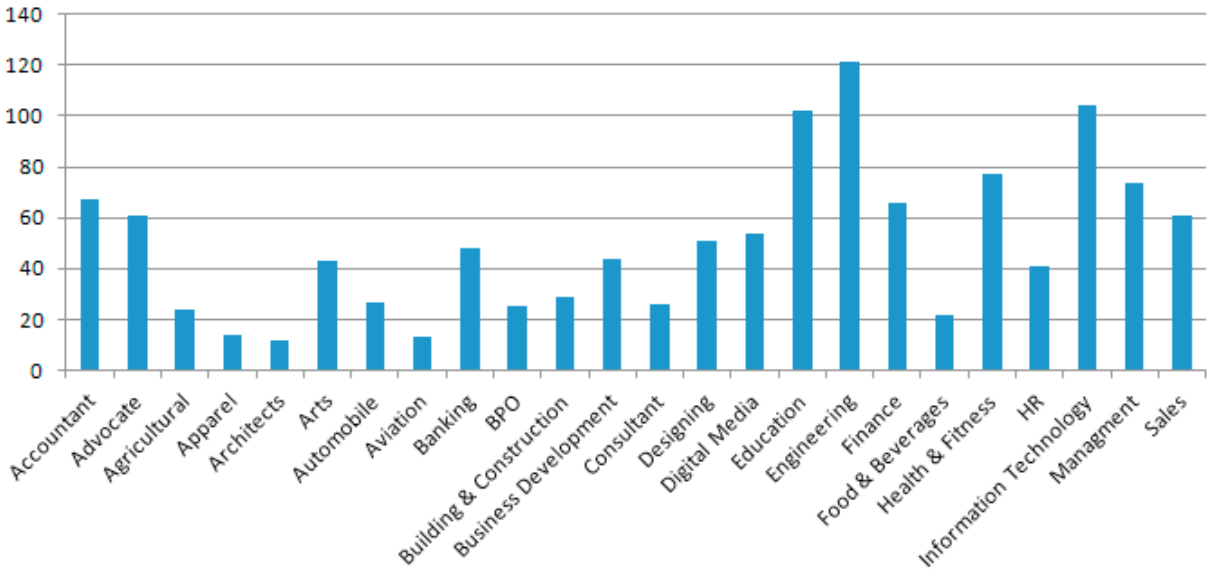

Fig. 1. Data Distribution over the different domains

- Stop words were masked from the dataset
- Lemmatization

Stop words removal: The stop words such as and, the, was, etc. are frequently appeared in the text and not helpful for prediction process, hence it is removed. Steps to filter the Stop Words:

1. We have tokenize the input words into individual tokens and stored it in an array
2. Now, each words matches with the list of Stop Words present in NLTK library:
   (a) `from nltk.corpus import stopwords` /*Imported Stop Word module from NLTK corpus*/
   (b) `StopWords[]= set(stopwords.words('english'))` /* Get set of English Stop Words*/
   (c) It returns total of 179 stop words, that can be verified using (`len(StopWords)`) and can be viewed by `print (StopWords)` function.
3. If the words present in the list of `StopWords[]`, filtered from the main sentence array.
4. The same process repeated until the last element of the tokenized array is not matched.
5. Resultant array does not have any stop words.

Stemming: Stemming is the method of decreasing word inflection to its root forms such as mapping a group of words to the same *stem* even though the stem itself is not a valid term in the language. Stem (root) is the part of the word to which you add inflectional (changing/deriving) affixes such as (-ed, -ize, -s, -de, -ing, mis). For example the words like: `Playing`, `Plays`, `Played` are mapped to their root word `Play`, the words like: `python`, `pythoner`, `pythoning`, `pythoned` mapped to their root word `python`:

$$\begin{pmatrix} Playing \\ Plays \\ Played \end{pmatrix} - - - - - - > \begin{pmatrix} Play \\ (root\ word) \end{pmatrix}$$

Lemmatization: Unlike Stemming, lemmatization decreases the inflected phrases to ensure that the root word belongs to the language correctly. Lemmatization comprises the following routine steps[1]:

[1] https://www.christianlehmann.eu/ling/ling_meth/ling_description/lexicography/lemmatization.html

- Transform the corpus of text into a list of words.
- Create a concordance of the corpus, i.e., of all the items of the word list as they occur in the corpus.
- Assign the word-forms to their lemmas based on the concordance.

The next step is feature extraction. On preprocessed dataset, we have extracted the features using the Tf-Idf [19]. The cleansed data was imported and feature extraction was carried out using Tf-Idf. The machine learning based classification model or learning algorithms need a fixed size numerical vector as input to process it. ML based classifiers did not process the raw text having variable size in length. Therefore, the texts are converted to a required equal length of vector form during the preprocessing steps. There are many approaches used to extract the features such as BoW(Bag of Words), tf-idf (Term Frequency, Inverse Document Frequency) etc. In BoW model, for each document, a complaint the narrative in our case, the presence (and often the frequency) of words is taken into consideration, but the order in which they occur is ignored. Specifically, we have calculated tf-idf (term frequency, and inverse document frequency) for each term present in our dataset using the *scikit learn* library function: `sklearn.feature_extraction.text.TfidfVectorizer` to calculate a tf-idf vector:

- to use a logarithmic form for frequency, sub-linear_df is set to True
- min_df is the minimum numbers of documents a word must be present in to be kept
- In order to ensure that the all feature vector have a euclidean norm of 1, norm is set to l2.
- gram_range is set to (n1, n2), where n1=1, and n2=2. It indicate that both uni-grams and bigrams considered.
- to reduce the non-informative features, stop_words is set to “english” to remove all common pronouns (“a”, “an”,“the”, “there”, ...)

4.2. Deployment and Inference

In this process the tokenised CV data and the job descriptions (JD) would be compared and the model would provide CVs relevant to the job description as an output.

5. Results

Two models have been built on the cleansed data: i) Classification - Based on the resume and category the model has been designed to categories the resume in the right category and ii) Recommendation -The model would create a summary of the resume and job description provided by the recruiter and give the list of most relevant resume based on the similarity between resume and jobs description

5.1. Classification

The classification was done using four different models and their accuracy score was recorded.

1. **Random Forest** (RF) [3]: RF is an ensemble learning method for classification that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) of the individual trees.
2. **Multinomial Naive Bayes** (NB) [21]: NB classifiers are a family of simple “probabilistic classifiers” based on Bayes’ theorem with strong independence assumptions between the features
3. **Logistic Regression** (LR) [17]: LR uses a logistic function to model a binary dependent variable.
4. **Linear Support Vector Classifier** (Linear SVM) [26]: A SVM is a supervised machine learning classifier which defined by a separating hyperplane. In two-dimensional space, a hyperplane is a line which separates a plane into two separate planes, where each plane belongs to a Class. For example, if the training sample contained two class data such as male (Class 0) and female (Class 1), then the output of the SVM is a line which separates the complete data into two classes using a line. Each plane represents a class of the data either male (Class 0) or female (Class 1).

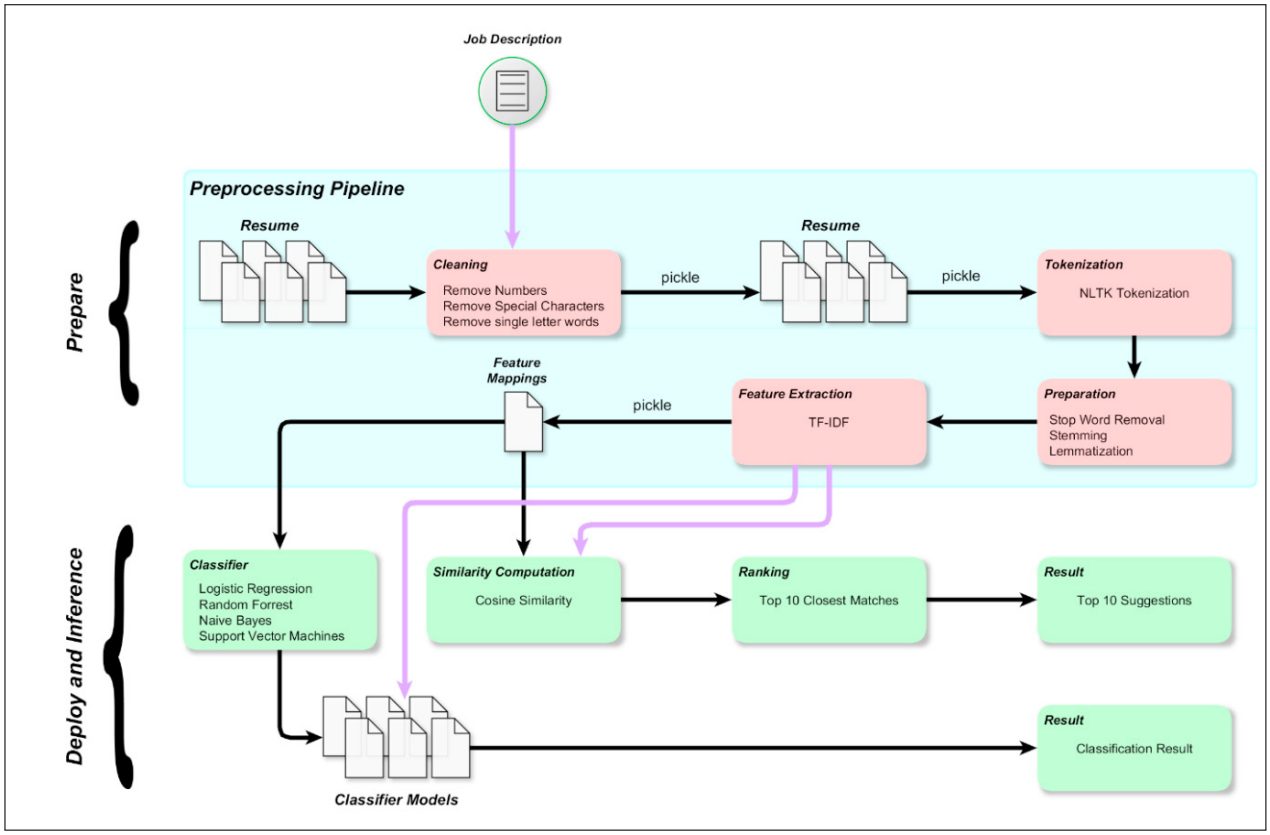

Fig. 2. A complete framework of the proposed model

Table 1. Results using the different classifiers

Classifier	**Accuracy**
Random Forest	0.3899
Multinomial Naive Bayes	0.4439
Logistic Regression	0.6240
Linear Support Vector Machine Classifier	**0.7853**

We have started our experiment with RF classifier, the extracted tf-idf features set is fed into the RF classifier to predict the resume category. The RF classifier yielded an accuracy of 38.99% on 10-fold cross-validation. The obtained results were not satisfactory and hence we used another popular classifier named "NB" for this task. NB classifier predicted the categories of resume with an accuracy of 44.39%, which was improved than the earlier classifier's accuracy (RF). However, 44.39% accuracy of NB classifier indicated that more than 50% of the resume was misclassified. We have used another classifier "Linear SVM" on the same data and achieved 78.53% accuracy. In order to improve the model accuracy, LR classifier was used and obtained 62.40% of accuracy which was lower than that of accuracy of "Linear SVM" classifier. The accuracy of all the models was calculated using 10 fold cross-validation, the average accuracy obtained from the classifiers was presented in Table 1.

The accuracy score of Linear Support Vector Classifier (Figure 3) higher compared to other models have we found this model to reliable and best fit for our objective. Continue with our best model (LinearSVC), we are going to look at the confusion matrix as shown in Figure 4, and show the discrepancies between predicted and actual labels. This is for the classification scope of our problem.

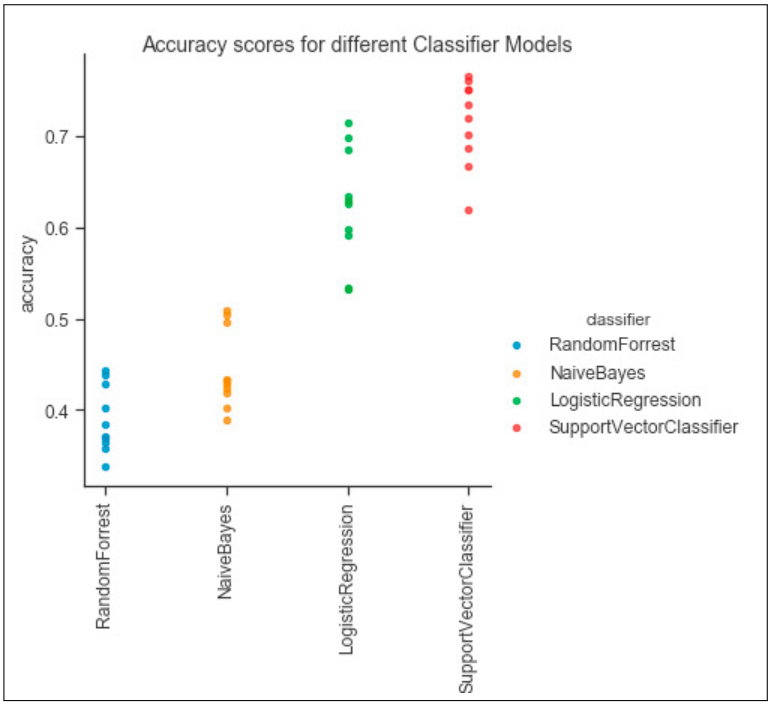

Fig. 3. A complete framework of the proposed model

5.2. CV Recommendation Model

The recommendation model is designed to take job description and CVs as input and provide the list of CVs which are closest to the provided job description. This is done using two approaches.

i) **Content Based Recommendation using Cosine Similarity**
ii) **k-Nearest Neighbours**

5.2.1. Content-Based Recommender

Considering this is the case of document similarity identification, we have gone with the Content-based recommender where Job Description provided by the employer is matched with the content of resumes in the space and the top n (n being configurable) matching resumes are recommended to the recruiter. The model takes the cleansed resume data and job description and combines the two into a single data set, and then computes the cosine similarity between the job description and CVs.

5.2.2. k-Nearest Neighbours

In this model, k-NN is used to identify the CVs that are nearest to the provided job description, in other words, the CVs that are close match to the provided job description. First, to get the JD and CVs to a similar scale, we have used an open source library called “gensim”, this library generates the summary of the provided text in the provided word limit. So to get the JD and CVs to similar word scale this library was used to generate a summary of JD and CVs and then k-NN was applied to find the CVs which are closely matching the provided JD.

5.3. Implications

The model designed is best suited for the first level of screening of the resumes by the recruiter. This would help the recruiter to classify the resumes as per the requirements and easily identify the CVs that are the best match to the job description. The model would assist the recruiter in hastening the profile shortlisting, at the same time ensuring

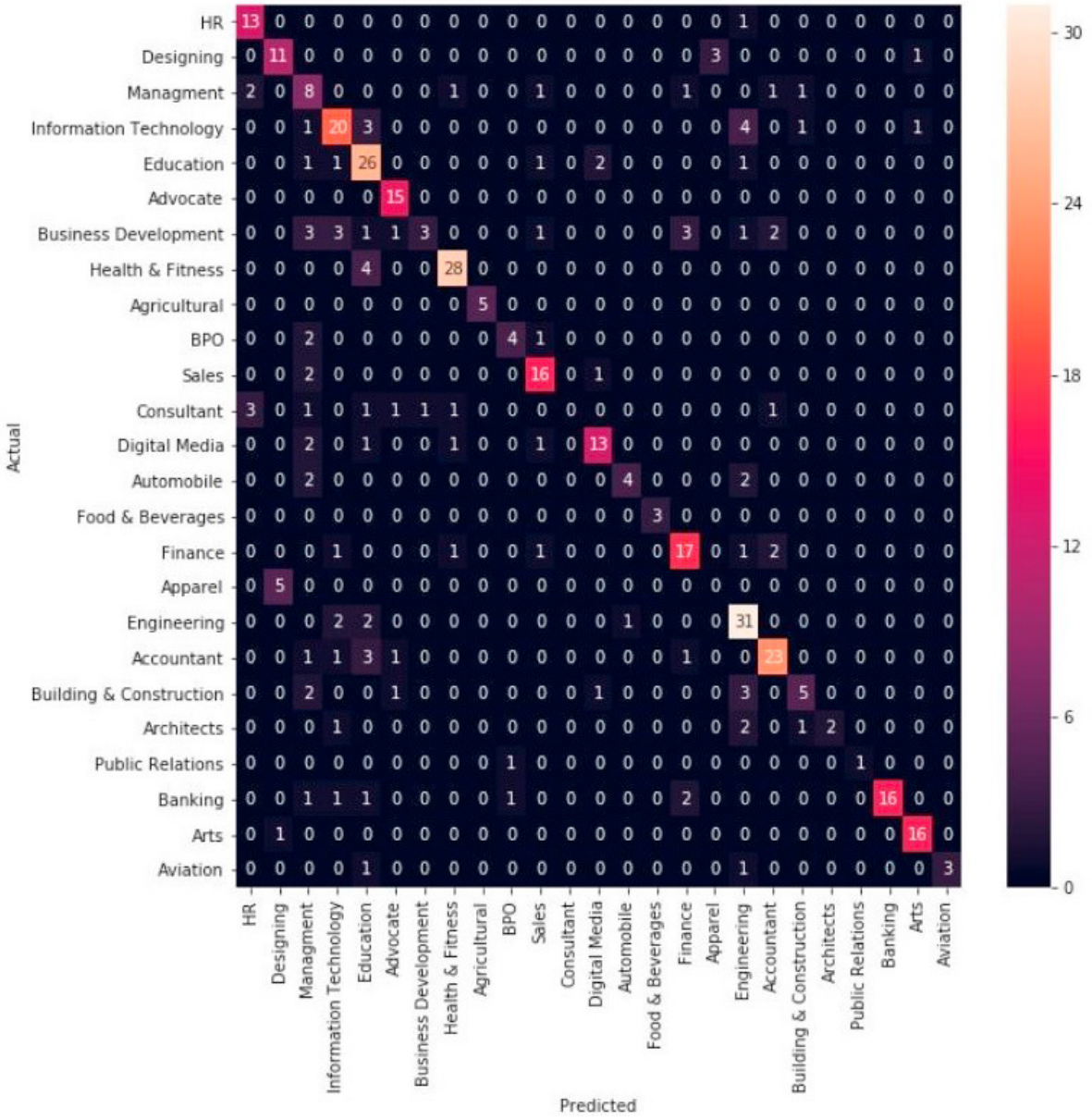

Fig. 4. Confusion matrix using Linear SVC classifier

credibility of the shortlisting process, as they would be able to screen thousands of resumes very quickly, and with the right fit, which would not have been possible for a human to do in near real time. This would aid in making the recruitment process efficient and very effective in identifying the right talent. Also, this would help the recruiter to reduce the resources spent in identifying the right talent making the process cost-effective. On the second level, the model provides the ranking to the CVs as per their fit vis-a-vis the job description, making it easier for the recruiter by giving the resume list in order of their relevance to the job. The recommendation made by the model are currently for the varied industry but the model and be further enhanced to target specific industry which would make it more effective, and give better recommendations.

5.4. Limitations

There are few limitations to the model design as of now, but these can be overcome by having more data to train the model. The current limitation of the model are i) Model takes CVs in CSV format, but in the real world, the CVs are either in .doc, .pdf, etc format. Due to the limitation of the data set, the model could not be enhanced to take .doc or .pdf as input, but using a library "textract" this can be achieved. The library can read varied file format and convert them into a single format which can be used as input to the model, ii) Generation of a summary using "genism" library might have caused loss of important information due to implicit compression of the text due to summarization. There is the scope of fine-tuning this summarization process to ensure minimal information loss, for example, important features of data like candidate skill and experience are not lost.

6. Conclusion

Huge number of applications received by the organization for every job post. Finding the relevant candidate's application from the pool of resumes is a tedious task for any organization nowadays. The process of classifying the candidate's resume is manual, time consuming, and waste of resources. To overcome this issue, we have proposed an automated machine learning based model which recommends suitable candidate's resume to the HR based on given job description. The proposed model worked in two phases: first, classify the resume into different categories. Second, recommends resume based on the similarity index with the given job description. The proposed approach effectively captures the resume insights, their semantics and yielded an accuracy of 78.53% with LinearSVM classifier. The performance of the model may enhance by utilizing the deep learning models like: Convolutional Neural Network, Recurrent Neural Network, or Long-Short Term Memory and others. If an Industry provides a large number of resume, then Industry specific model can be developed by utilizing the proposed approach. By involving the domain experts like HR professional would help to build a more accurate model, feedback of the HR professional helps to improve the model iteratively.

References

[1] Al-Otaibi, S.T., Ykhlef, M., 2012. A survey of job recommender systems. International Journal of Physical Sciences 7, 5127–5142.
[2] Breaugh, J.A., 2009. The use of biodata for employee selection: Past research and future directions. Human Resource Management Review 19, 219–231.
[3] Breiman, L., 2001. Random forests. Machine learning 45, 5–32.
[4] Carrer-Neto, W., Hernández-Alcaraz, M.L., Valencia-García, R., García-Sánchez, F., 2012. Social knowledge-based recommender system. application to the movies domain. Expert Systems with applications 39, 10990–11000.
[5] Celma, O., 2010. Music recommendation, in: Music recommendation and discovery. Springer, pp. 43–85.
[6] Das, A.S., Datar, M., Garg, A., Rajaram, S., 2007. Google news personalization: scalable online collaborative filtering, in: Proceedings of the 16th international conference on World Wide Web, ACM. pp. 271–280.
[7] Diao, Q., Qiu, M., Wu, C.Y., Smola, A.J., Jiang, J., Wang, C., 2014. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars), in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM. pp. 193–202.
[8] Färber, F., Weitzel, T., Keim, T., 2003. An automated recommendation approach to selection in personnel recruitment. AMCIS 2003 proceedings , 302.
[9] Golec, A., Kahya, E., 2007. A fuzzy model for competency-based employee evaluation and selection. Computers & Industrial Engineering 52, 143–161.
[10] Howard, J.L., Ferris, G.R., 1996. The employment interview context: Social and situational influences on interviewer decisions 1. Journal of applied social psychology 26, 112–136.
[11] Lin, Y., Lei, H., Addo, P.C., Li, X., 2016. Machine learned resume-job matching solution. arXiv preprint arXiv:1607.07657 , 1–8.
[12] Loper, E., Bird, S., 2002. Nltk: the natural language toolkit. arXiv preprint cs/0205028 .
[13] Lu, J., Wu, D., Mao, M., Wang, W., Zhang, G., 2015. Recommender system application developments: a survey. Decision Support Systems 74, 12–32.
[14] Maheshwary, S., Misra, H., 2018. Matching resumes to jobs via deep siamese network, in: Companion Proceedings of the The Web Conference 2018, International World Wide Web Conferences Steering Committee. pp. 87–88.
[15] Malinowski, J., Keim, T., Wendt, O., Weitzel, T., 2006. Matching people and jobs: A bilateral recommendation approach, in: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06), IEEE. pp. 137c–137c.
[16] Mooney, R.J., Roy, L., 2000. Content-based book recommending using learning for text categorization, in: Proceedings of the fifth ACM conference on Digital libraries, ACM. pp. 195–204.

[17] Nasrabadi, N.M., 2007. Pattern recognition and machine learning. Journal of electronic imaging 16, 049901.
[18] Paparrizos, I., Cambazoglu, B.B., Gionis, A., 2011. Machine learned job recommendation, in: Proceedings of the fifth ACM Conference on Recommender Systems, ACM. pp. 325–328.
[19] Ramos, J., et al., 2003. Using tf-idf to determine word relevance in document queries, in: Proceedings of the first instructional conference on machine learning, Piscataway, NJ. pp. 133–142.
[20] Resnick, P., Varian, H.R., 1997. Recommender systems. Communications of the ACM 40, 56–59.
[21] Rish, I., et al., 2001. An empirical study of the naive bayes classifier, in: IJCAI 2001 workshop on empirical methods in artificial intelligence, pp. 41–46.
[22] Roy, P.K., Singh, J.P., 2018. A tag2vec approach for questions tag suggestion on community question answering sites, in: International Conference on Machine Learning and Data Mining in Pattern Recognition, Springer. pp. 168–182.
[23] Roy, P.K., Singh, J.P., Baabdullah, A.M., Kizgin, H., Rana, N.P., 2018a. Identifying reputation collectors in community question answering (cqa) sites: Exploring the dark side of social media. International Journal of Information Management 42, 25–35.
[24] Roy, P.K., Singh, J.P., Nag, A., 2018b. Finding active expert users for question routing in community question answering sites, in: International Conference on Machine Learning and Data Mining in Pattern Recognition, Springer. pp. 440–451.
[25] Schafer, J.B., Konstan, J., Riedl, J., 1999. Recommender systems in e-commerce, in: Proceedings of the 1st ACM conference on Electronic commerce, ACM. pp. 158–166.
[26] Schölkopf, B., Smola, A.J., Bach, F., et al., 2002. Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press.
[27] Singh, J.P., Irani, S., Rana, N.P., Dwivedi, Y.K., Saumya, S., Roy, P.K., 2017. Predicting the helpfulness of online consumer reviews. Journal of Business Research 70, 346–355.
[28] Wei, K., Huang, J., Fu, S., 2007. A survey of e-commerce recommender systems, in: 2007 international conference on service systems and service management, IEEE. pp. 1–5.
[29] Yi, X., Allan, J., Croft, W.B., 2007. Matching resumes and jobs based on relevance models, in: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, ACM. pp. 809–810.
[30] Zhang, L., Fei, W., Wang, L., 2015. Pj matching model of knowledge workers. Procedia computer science 60, 1128–1137.