Proceedings of the 5th International Conference on Smart Systems and Inventive Technology (ICSSIT 2023)
IEEE Xplore Part Number: CFP23P17-ART; ISBN: 978-1-6654-7467-2

Resume Classification using Elite Bag-of-Words Approach

Muskan Sharma
Dept. of Information Technology
Delhi Technological University
Delhi, India
muskan_2k19it082@dtu.ac.in

Gargi Choudhary
Dept. of Information Technology
Delhi Technological University
Delhi, India
gargi_2k19it048@dtu.ac.in

Seba Susan
Dept. of Information Technology
Delhi Technological University
Delhi, India
seba_406@yahoo.in

2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT) | 978-1-6654-7467-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICSSIT55814.2023.10061036

***Abstract*—As technology is advancing day by day, new trends are booming up, like automation, where traditional libraries are being automated to digital libraries. Therefore, instead of manually screening the resumes of all candidates, algorithms and models are being employed to screen resumes in job and career portals. This complete process of mapping resumes to their corresponding job profiles could be efficiently accomplished by making use of various machine learning and Natural Language Processing (NLP) tools. This article utilizes a recently introduced text vectorization technique called Elite bag-of-words for the vectorization of resumes. To implement this method, words in each class are ranked based on their occurring frequency, and then applied maximum entropy partitioning (MEP) to derive the top-ranked significant keywords in each class. These keywords, defined as the Elite keywords, were extracted from each class, and concatenated without redundancy, for predicting the resume type. This research study presents an experimental comparison of the proposed method with existing bag-of-words approaches. This paper implements four vectorization techniques and it is proved that the Elite bag-of-words approach outperforms the other methods for resume classification.**

Keywords – Bag-of-words, Elite keywords, Term frequency, Resume classification.

I. INTRODUCTION

In an increasingly competitive world, the strife to get selected and secure a job is day by day becoming even more difficult and complicated. With the onset of the surge in the number of job profiles versus the number of candidates in the current market scenarios, the sole motive or intention of every other organisation or corporate house is to find the best and most appropriate candidate with the required skill set as per the job profile description. The companies or organizations receive a humongous number of resumes on career portals. Categorizing them on the basis of their job title/posting as per their skills set is quite a tedious task if performed manually; hence automated resume screening is the need of the hour [1]. Since the problem of classification of resumes is a subset of the document classification problem, just like text or sentiment analysis, the tokenization methods of document analysis followed by machine learning can be used for resume classification as well [2].

The efficiency of document categorization depends on how well the machine learning algorithms are able to learn from the given data. Since these algorithms can be applied only on numerical representations, therefore, as a prerequisite, first the resume data having text, paragraphs, sentences should be transformed into feature vectors that can be further applied as input to various machine learning and deep learning models [3].

Now this problem can be solved by making use of popular text vectorization techniques [4] that transform text into numerical data representation called the bag-of-words (BoW) feature representation where the feature columns are the keywords that comprise the input vocabulary. Examples of BoW are the Term Frequency (TF) and Term Frequency - Inverse Document Frequency (TF-IDF) that have been amply used in literature for representing text in various domains and applications [5, 6, 7]. Another alternative to BoW is the use of word embeddings like Word2Vec [8]. Bag-of-words models like TF, TF-IDF in conjunction with machine learning classifiers have been used before for resume classification [9, 10]. Word embeddings have also been used along with convolutional neural networks [11, 12] and recurrent neural networks [13] for resume classification.

This paper aims to determine whether the Elite keywords which is a recently introduced bag-of words model for document representation [14] can serve out to be helpful in the categorization of curriculum vitae as per varied job roles. Elite keywords are defined by the authors in [14] to be the most significant keywords in each class, distinctive in terms of their frequencies of occurrences. The iterative Maximum Entropy Partitioning (MEP) algorithm is used to determine the threshold of the number of significant keywords in a class. The Elite keywords are then concatenated across classes after eliminating redundancy. The organization of the rest of this paper is as follows. Section II discusses some preliminaries on text pre-processing and vectorization, section III presents the methodology followed, section IV discusses the results, and section V concludes the paper.

II. PRELIMINARIES ON TEXT PRE-PROCESSING AND TEXT VECTORIZATION

Before applying the suitable machine learning models on the proposed dataset comprising of multiple resumes, a series of text pre-processing steps needs to be carried out [15]. The first step is to pre-process the resume data in order to remove insignificant words or noises so that the machine learning techniques could work more efficiently. Below are the steps used to perform text pre-processing.

1) Elimination of irrelevant punctuation marks - The presence of these delimiters can contribute a lot towards adding noise to our dataset which further might affect the accuracy.

2) Deletion of stop-words - Since stop-words do not carry much significance or add meaning to the classification task at hand, so they can be ignored and direct all the focus on the

words that are actually important in our predictions, and vectorize them and proceed further. The stop-words are language specific, that means there exists different sets of stop-words for – English, German, Spanish etc. In case of English, some of the stop-words include- "the", "a", "upon", "above", "is", "of", "below".

3) The resume cleaning also included removal of URL's, hashtags, mentions, extra whitespaces, numbers and non-English characters. This is followed by conversion of all uppercase letters into lowercase so that there is no ambiguity in case of two or more occurrences of the same word in both uppercase and lowercase.

4) Stemming and Lemmatization- Stemming refers to the phenomenon of reducing a given word to its stem or root word. The root of the word in this case generally is not a meaningful word. It is implemented by the class Porter Stemmer in the module: nltk.stem.porter present in the NLTK library [15]. Whereas, lemmatization refers to the phenomenon which is quite similar to the process of stemming. There exists only a slight difference; that in case of lemmatization, it reduces the given word to its corresponding root word which comes out to be a meaningful word. For the implementation purpose, an object is created of the class WordNet Lemmatizer inside the nltk.stem module present in the NLTK library. It comes with an added advantage over the previous method of stemming, that the final word representation after reductions is understandable and meaningful. For the resume text, first lemmatization has been performed and then stemming process has been carried out on the lemmatized words.

5) Feature extraction by text vectorization – This is the constitutional model which forms the basis of Natural Language Processing, that underlines the importance of transforming the tokens or words in a document into a numeric representation called the feature vector. In this work, we explore the bag-of-words model for text representation in which the words form feature columns and the rows represent resume samples. Bag-of-words (BoW) model has been successfully used before for the representation of large document classes like 20-Newsgroups [16], named entity recognition [17], sentiment analysis [18], and topic modelling from social media posts [19]. We therefore found the bag-of-words model an apt choice for the vectorization and representation of resume documents in our current study. Some popular bag-of-words models that we used in our experiments are described next.

a) One-hot encoding

The one-hot encoding is the most simplistic and conventionally used BoW model. The crux of this model lies in the fact that after we are done with all the pre-processing, we create a dictionary of all the keywords present in the corpus and based on their occurrences within a text document they are mapped with binary output i.e.- either 0 or 1, where 1 denotes that particular keyword is present in the document, and on the other hand 0 denotes the absence of the keyword. Most of the times while applying one-hot encoding, we are left with a sparse matrix having elements- 0 and 1.

b) Term frequency (TF)

TF stands for Term Frequency which is the count or frequency of keywords in a document. The underlying principle on which the TF works, is based on the fact that it takes into consideration the number of times a particular keyword k is present in the text document- in our case the document is the resume r.

TF can be represented mathematically as

$$TF(k, r) = count(k)|r \quad (1)$$

where, k stands for keyword and r represents the document.

c) Term frequency – inverse document frequency (TF-IDF)

The acronym TF-IDF stands for Term Frequency - Inverse Document Frequency, wherein we create a word frequency map or dictionary where each word is mapped to its corresponding frequency, and multiply this frequency by a weight that represents how rare this keyword is across all documents. TF-IDF is a modified version of the original Term Frequency (TF) wherein, in addition to the basic functionalities of the TF an added benefit is there - it aims to focus more on those frequently occurring keywords that do not occur commonly in all documents.

The ability of the TF-IDF to distinguish and emphasize on unique keywords is an added advantage over TF. The usual process for calculating it is divided into two parts. We individually create the Term Frequency (TF) matrix and the Inverse Document Frequency (IDF) matrix. And we then multiply the two matrices. Mathematically, TF-IDF can be represented as

$$TF-IDF(k,r) = TF(k, r) \times IDF(k) \quad (2)$$

Here $IDF(k)$ is the logarithm of the inverse fraction of documents that contain the keyword k. TF-IDF is one of the most popular and reliable BoW models used in NLP, noted for its advantages over TF and one-hot encoding.

III. RESUME CLASSIFICATION USING ELITE BAG-OF-WORDS

One disadvantage of TF and other BoW approaches is that there is no scheme of separating out redundant keywords that may affect the performance of the resume classification. Removing redundant and non-informative keywords or feature columns is the need of the hour. Usually feature selection schemes are additionally used for selecting important keywords [20]. However, feature selection by itself is an unstable method and the set of selected features depends heavily on the training samples [21].

In this paper we explore the use of Elite keywords [14], a recently proposed bag-of-words approach, for extracting significant keywords separately from each resume class. After shortlisting the significant keywords, they are concatenated across classes after removing the redundant keywords. This ensures that class-specific keywords are included in the feature columns. Since resume text is expected to contain keywords specific to a class that may not be as important for the other classes, therefore, the procedure of Elite keyword extraction from each class, and concatenation, will ensure that only significant keywords are selected. The method also returns stable results since the maximum entropy partitioning (MEP) method is used to separate the significant keywords from the non-significant keywords in each resume class.

The methodology followed in this paper is divided-majorly into five steps as shown in the process flow in Fig. 1: a) Data preparation- which involves collecting data from online resources b) Applying data pre-processing techniques c) Converting the text inside the documents into feature vectors d) Training the machine learning models e) Feeding resumes for testing purposes and predicting the resume category using the trained model.

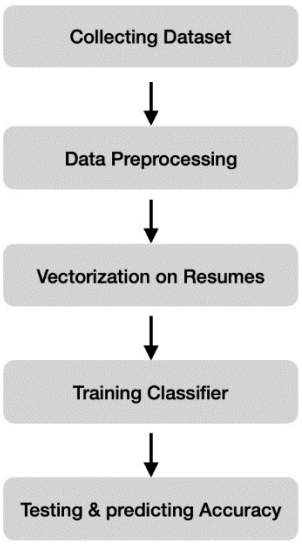

Fig. 1. Process flow

In order to adopt an optimal approach for the purpose of categorization of resumes, we first need to find a minimized subset of most relevant keywords for every category in the dataset.

The Elite keywords extraction was introduced in a recent work [14] as an automated technique for finding subsets of significant keywords from each class using maximum entropy partitioning (MEP). These significant keywords provide ample knowledge about each class and are based on relative term frequencies within each class. The detailed procedure for finding the Elite keywords for resume classification is described next.

1) Creating vectorizers

The first step is similar to that of BoW, in which we need to create a TF matrix for all the words occurring in the resumes of a specific class. This step would give us *C* matrices, each corresponding to the term frequencies of keywords in each category, where *C* is the number of resume classes.

2) Calculating cumulative Frequencies

From the TF matrices obtain the cumulative count of each word for an entire category; this can be achieved by simply adding all the frequencies corresponding to that word for a specific class.

3) Calculating Relative Frequency

For each class, first sum up all the cumulative frequencies and then divide each cumulative frequency with that sum in order to obtain a relative frequency value for each keyword.

4) Applying Maximum Entropy Partitioning

After obtaining the relative term frequencies in each class, we need to sort the frequencies in descending order and then apply the iterative MEP algorithm to partition the sorted relative frequency values into two parts - the upper part is defined as the Elite keywords and the lower part is to be discarded. MEP algorithm involves the computation of the Shannon entropy for the upper and lower parts, and summing up the two entropies. The sum of the probabilities in each of the upper and lower parts should sum up to one prior to the computation of entropy. The partition which gives the maximum sum of entropies is the optimal partition, since at this point, the probability distributions in the two sections approximate uniform distributions.

Mathematically, Shannon entropy is calculated by the equation shown below:

$$E = -\sum_{s} s \log s \tag{3}$$

where, E denotes the entropy and s stands for the probability values. MEP will be returning the optimal index at which we need to partition the sorted relative probabilities in order to get the subset of the most significant keywords. Therefore, to achieve the MEP index we need to run a loop from 2 to the length of the relative probabilities array, that would partition the array into two groups, and then we calculate the sum of the entropies of both the groups using the formula in (3). For the optimal partition we need to compare entropies at all the indices and hence return the index i at which the sum of the two entropies is maximum.

5) Concatenation of obtained Elite keywords

The Elite keywords obtained by maximum entropy partitioning of each class are concatenated across classes after removing the redundant keywords. This final array that we obtained after concatenation is the set of Elite keywords that will be used as tokens to calculate the TF matrix for the training set of the resume dataset.

6) Creating a TF matrix

The TF matrix is obtained by calculating the frequency of the concatenated Elite keywords in each resume document. Then we feed the matrix as input to the classifier and use the trained model for prediction.

IV. RESULTS AND DISCUSSIONS

All experiments are performed on the Kaggle resume dataset available online [1] that has 24 categories such as Accountant, Teacher etc. as shown in Table I. There are 1738 training samples and 744 testing samples in this dataset. The dataset consists of multiple attributes like ID, Resume_html, Resume_str. Resume_html which contains html tags corresponding to each resume, and ID which associates a unique numeric value to each resume, were dropped off during the pre-processing state because both of these hardly carried any significance to our experiments.

We compare the performance of the Elite bag-of-words for resume classification with different bag-of-words approaches found in literature, using one of the most effective classifiers used for text classification – random forest (with Grid search for hyperparameter optimization). Python 3.7 version software is used. All the BoW codes just took a few seconds to execute on a 2.6 GHz Intel PC. We

[1] https://www.kaggle.com/datasets/snehaanbhawal/resume-dataset

have made our Python code for extracting Elite keywords available online[2] for facilitating future research. We have shown the number of Elite keywords extracted from the 24 resume classes in Table I, along with the total number of keywords extracted from each class. It is observed that the application of MEP drastically reduces the number of features as verified from the Elite keywords' column in Table I.

TABLE I. RESUME CATEGORIES AND NUMBER OF KEYWORDS

Resume Categories	Elite keywords	Total Keywords
Accountant	2272	5084
Advocate	2573	5530
Agriculture	1994	4662
Arts	2198	5227
Apparel	2141	5384
Automobile	1750	3227
Aviation	2735	6222
Banking	2481	5396
BPO	1359	2481
Business-Development	2312	5424
Chef	2566	5674
Construction	2395	5514
Consultant	2687	6202
Designer	2305	5770
Digital-Media	1986	5150
Engineering	2692	6310
Finance	2189	5051
Fitness	2344	5575
Healthcare	2676	5948
HR	2027	4538
Information-Technology	2844	6275
Public-Relation	2505	6239
Sales	2226	5090
Teacher	2011	4636

For the visual representation of keywords, we have used the Word cloud in Python which primarily helps us in visualizing the text where the size of a specific word denotes the frequency or significance of the word in the resume. Fig. 2 shows the word cloud obtained using the vanilla BoW model for the "Accountant" class. The Elite keywords derived for the class "Accountant" are shown in the word cloud in Fig. 3. It is observed from the comparison of Fig. 2 and Fig. 3 that that class-specific significant words like *financial* and *company* are given more prominence in the list of Elite keywords for the class "Accountant". On the other hand, common words like *work*, *instruction*, *provide* etc. found in Fig. 2 have been removed in the Elite keyword subset as observed from Fig. 3.

[2] https://github.com/Muskankalonia/Resume-Classification-Using-Elite-Bag-of-Words-Approach

Fig. 2. Keywords detected in the class "Accountant"

Fig. 3. Elite keywords detected in the class "Accountant"

We train all features on the random forest classifier. Further detailed performances of the BoW models can be analyzed from Table II which shows the results of the random forest classifier on the different bag-of-words representations like One-hot encoding, TF, TF-IDF and Elite keywords. Therefore, upon comparison the best performing model is observed to be the Elite keywords which gave the highest test accuracy of 62.60%.

TABLE II. PERFORMANCE COMPARISON OF DIFFERENT BAG-OF-WORDS APPROACHES FOR RESUME CLASSIFICATION

Method	Test accuracy
One-hot encoding	54.55%
TF-IDF	55.36%
TF	58.98%
Elite keywords	62.60%

The second-best performing model was TF with an accuracy of 58.98%, followed by TF-IDF with an accuracy score of 55.36%. One-hot encoding performed worst giving an accuracy score of 54.55%.

V. CONCLUSION

There is an increasing demand for automated resume screening from job and career portals. Natural language processing is regarded as the most popular means for understanding the content of resume, in order to classify the resume to different job profiles. We explore the most popular NLP tool in our work called the bag-of-words representation in which the text is transformed into a feature vector where the keywords constitute the feature columns. Removing redundant and non-informative keywords or feature columns is the need of the hour. Usually feature selection schemes are used for selecting important

keywords. However, feature selection is an unstable method and varies depending on the training samples. In this paper, we explore Elite keywords, a recently proposed bag-of-words approach, for shortlisting the significant keywords separately for each class. After shortlisting the keywords, they are concatenated across classes after removing the redundant keywords. This ensures that class-specific keywords are included in the feature columns. Since resume document contains keywords specific to a class that may not be as important for the other classes, therefore, the procedure of Elite keyword extraction from each class individually followed by concatenation will ensure that class-specific significant keywords are selected. The method is also stable since the maximum entropy partitioning method is used to automatically separate the significant keywords from the non-significant keywords in each class. We train the features on the random forest classifier using Grid search for hyperparameter optimization. After carefully observing the outcomes of each model, we noted that the Elite keyword subset was by far the most reliable and accurate bag-of-words model for classifying different categories of resumes in the benchmark dataset. Graphical methods for representing the significant keywords in resumes will be explored in our future work. Semantic representations such as the fuzzy bag-of-words approach will also be explored in future. Resume matching and retrieval based on input job profiles is also less explored research that will be the subject of our future study.

REFERENCES

[1] Zaroor, Abeer, Mohammed Maree, and Muath Sabha. "JRC: a job post and resume classification system for online recruitment." In *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 780-787. IEEE, 2017.

[2] Roy, Pradeep Kumar, Sarabjeet Singh Chowdhary, and Rocky Bhatia. "A Machine Learning approach for automation of Resume Recommendation system." *Procedia Computer Science* 167 (2020): 2318-2327.

[3] Swami, Pratibha, and Vibha Pratap. "Resume Classifier and Summarizer." In *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, vol. 1, pp. 220-224. IEEE, 2022.

[4] Wang, Yanzhe. "Basic Methodologies Used in NLP Area." In *2020 IEEE 3rd International Conference on Automation, Electronics and Electrical Engineering (AUTEEE)*, pp. 505-511. IEEE, 2020.

[5] Hamisu, Muhammad, and Ali Mansour. "Detecting advance fee fraud using nlp bag of word model." In *2020 IEEE 2nd International Conference on Cyberspac (CYBER NIGERIA)*, pp. 94-97. IEEE, 2021.

[6] Junior, Antonio P. Castro, Gabriel A. Wainer, and Wesley P. Calixto. "Weighting construction by bag-of-words with similarity-learning and supervised training for classification models in court text documents." *Applied Soft Computing* (2022): 108987.

[7] Helaskar, Mukund N., and Sheetal S. Sonawane. "Text Classification Using Word Embeddings." In *2019 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA)*, pp. 1-4. IEEE, 2019.

[8] Sivakumar, Soubraylu, Lakshmi Sarvani Videla, T. Rajesh Kumar, J. Nagaraj, Shilpa Itnal, and D. Haritha. "Review on Word2Vec Word Embedding Neural Net." In *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, pp. 282-290. IEEE, 2020.

[9] Ali, Irfan, Nimra Mughal, Zahid Hussain Khand, Javed Ahmed, and Ghulam Mujtaba. "Resume classification system using natural language processing and machine learning techniques." *Mehran University Research Journal Of Engineering & Technology* 41, no. 1 (2022): 65-79.

[10] Duan, Liting, Xiaolin Gui, Mingan Wei, and You Wu. "A Resume Recommendation Algorithm Based on K-means++ and Part-of-speech TF-IDF." In *Proceedings of the 2019 International Conference on Artificial Intelligence and Advanced Manufacturing*, pp. 1-5. 2019.

[11] Mridha, M. F., Rabeya Basri, Muhammad Mostafa Monowar, and Md Abdul Hamid. "A Machine Learning Approach for Screening Individual's Job Profile Using Convolutional Neural Network." In *2021 International Conference on Science & Contemporary Technologies (ICSCT)*, pp. 1-6. IEEE, 2021.

[12] Nasser, Shabna, C. Sreejith, and M. Irshad. "Convolutional neural network with word embedding based approach for resume classification." In *2018 International Conference on Emerging Trends and Innovations In Engineering And Technological Research (ICETIETR)*, pp. 1-6. IEEE, 2018.

[13] Xu, Qiqiang, Ji Zhang, Youwen Zhu, Bohan Li, Donghai Guan, and Xin Wang. "A block-level RNN model for resume block classification." In *2020 IEEE International Conference on Big Data (Big Data)*, pp. 5855-5857. IEEE, 2020.

[14] Susan, Seba, and Juli Keshari. "Finding significant keywords for document databases by two-phase Maximum Entropy Partitioning." Pattern Recognition Letters 125 (2019): 195-205.

[15] Loper, Edward, and Steven Bird. "NLTK: the Natural Language Toolkit." In *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1*, pp. 63-70. 2002.

[16] Raj, Anshula, and Seba Susan. "Clustering Analysis for Newsgroup Classification." In *Data Engineering and Intelligent Computing*, pp. 271-279. Springer, Singapore, 2022.

[17] Antony, J. Betina, and G. S. Mahalakshmi. "Named entity recognition for Tamil biomedical documents." In *2014 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2014]*, pp. 1571-1577. IEEE, 2014.

[18] Haddi, Emma, Xiaohui Liu, and Yong Shi. "The role of text pre-processing in sentiment analysis." *Procedia computer science* 17 (2013): 26-32.

[19] Sharma, Anubhav, Seba Susan, Anmol Bansal, and Arjun Choudhry. "Dynamic Topic Modeling of Covid-19 Vaccine-Related Tweets." In *2022 the 5th International Conference on Data Storage and Data Engineering*, pp. 79-84. 2022.

[20] Lee, Lam Hong, Dino Isa, Wou Onn Choo, and Wen Yeen Chue. "High Relevance Keyword Extraction facility for Bayesian text classification on different domains of varying characteristic." *Expert Systems with Applications* 39, no. 1 (2012): 1147-1155.

[21] Singh, Yashpal, and Seba Susan. "SMOTE-LASSO-DeepNet Framework for Cancer Subtyping from Gene Expression Data." In *2022 15th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1-6. IEEE, 2022.

[22] Zhao, Rui, and Kezhi Mao. "Fuzzy bag-of-words model for document representation." *IEEE transactions on fuzzy systems* 26, no. 2 (2017): 794-804.

[23] Li, Changmao, Elaine Fisher, Rebecca Thomas, Steve Pittard, Vicki Hertzberg, and Jinho D. Choi. "Competence-Level Prediction and Resume & Job Description Matching Using Context-Aware Transformer Models." In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8456-8466. 2020.