IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 37, No. 3, June, 2023, pp. 619-625
Journal homepage: http://iieta.org/journals/ria

Machine Learning-Based Recommendations and Classification System for Unstructured Resume Documents

Channabasamma[1*], Yeresime Suresh[2]

[1] Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad 500090, India
[2] Department of Computer Science and Engineering, Ballari Institute of Technology and Management, "Jnana Gangotri" Campus, Ballari 583104, India

Corresponding Author Email: channu.ar@gmail.com

https://doi.org/10.18280/ria.370311

Received: 2 February 2023
Accepted: 25 March 2023

Keywords:
categorization, classification, data extraction, recognition, recommendation, resume, screening, skills

ABSTRACT

With the burgeoning growth of the job market and a surge in applications, the processes of job recommendation and candidate selection have become complex and labor-intensive. The advent of new technologies such as machine learning has automated these processes, yet the unstructured nature of resumes, often in PDF format, necessitates laborious data extraction for efficient skill-based candidate screening and categorization. Ineffectual recruitment can result from mismatched skills. The system proposed in this study aims to address these challenges by automatically fetching and categorizing resumes, extracting critical information, and utilizing job descriptions for candidate selection and recommendations. Unstructured data from PDF documents is extracted using a PDF reader, and machine learning algorithms, specifically logistic regression and Gaussian Naïve Bayes, are employed for generating recommendations. In an innovative approach, this system not only classifies resumes but also recommends updates or rewrites. Performance of the proposed system is evaluated in terms of classification accuracy and the effectiveness of update recommendations, and results are compared with alternative models. This research represents a significant advancement in the application of machine learning to the automation of job recommendation and candidate selection processes.

1. INTRODUCTION

Resume screening represents a crucial step within a company's hiring process, permitting evaluation of potential employees prior to their recruitment. Furthermore, it facilitates the preparation of candidates for comprehensive examination [1]. The ultimate objective of resume screening is to identify suitable individuals for job vacancies, while concurrently preventing the company from overlooking exceptional candidates [1]. However, manual screening of resumes is a time-consuming and complex process, particularly given that each job vacancy typically attracts hundreds of resumes [2].

Although resumes generally adhere to a standard format, the specific skills and qualifications demanded by each role may necessitate varying levels of specificity in their creation [3]. The majority of contemporary job applications require electronic resumes. However, the development of resume screening approaches based on modern methodologies, knowledge discovery, and social networks remains in its infancy, with substantial research required prior to the implementation of commercial systems [4].

One notable limitation of resume screening is its reliance on candidate-provided content entered into job portals or electronic forms during the job application process. While job seekers may possess unique attributes that distinguish them, discrepancies may arise if the data entered does not match the content of the candidate's resume. At its most effective, resume screening could yield a list of individuals with similar resumes, and vice versa, but this could introduce significant bias and inaccuracies particularly in the case of numerous unsupervised searches [5]. Therefore, it is imperative that uploaded resume documents are screened for an effective job-matching profile. Consequently, the extraction of useful data from these documents is essential. In this context, Natural Language Processing (NLP) can be employed to extract job-matching attributes from the resume document [6]. The extracted attribute values can then be compared to the skill set requirements for recruitment, facilitating the identification of relevant jobs with commensurate salaries.

What is Natural Language Processing?

Natural Language Processing (NLP) is a process of breaking down the given input into component parts and interpreting their meaning. The process includes converting the input into a form that a computer can understand [7]. NLP can be employed to extract high-quality information from documents and queries by developing the computer's ability to understand and communicate with humans in the human language.

The rising use of social media applications such as Facebook, LinkedIn, Twitter etc. for job-seekers' screening activities is likely to propel the growth of the NLP market. NLP being a demanded technique is used by researchers in extracting the most relevant information, identifying negative character traits, and more. In the recruitment industry resumes

need to be screened, and the process requires to be automated, and NLP is one among the appropriate method to interpret the resume [8]. Even well-written resumes will never be 100% successful without proper interpretation or extraction. The research is motivated by several factors, including the exponential growth in the job market and the subsequent increase in the number of job applications, which has made the recruitment process more complex and time-consuming. Additionally, the unstructured data in the form of pdf resumes has made it even more challenging to extract structured information to automate the screening process [9].

The proposed system is motivated by the need to ease the recruitment process by providing an automated solution to screen candidates, categorize them based on their skills, and recommend job updates or rewrites. This will help recruiters to efficiently and accurately identify the most suitable candidates for a job.

Furthermore, the proposed system is also motivated by the need to reduce the mismatch between candidate skills and the skills demanded by industries, which can lead to ineffectiveness in all phases of the recruitment process. The system aims to ensure that the skills of the candidates are aligned with the skills required by the industry, resulting in more effective job candidate selections.

Overall, the research is motivated by the need to improve the recruitment process by providing an automated and accurate system to screen candidates, categorize them, and provide job recommendations. The proposed system is expected to improve the efficiency and effectiveness of the recruitment process, resulting in more successful job placements.

This paper utilizes NLP for effective data extraction from resume documents. NLP helps in analyzing every aspect of a resume and helps not only to screen the appropriate resume but also helps in recommending candidates to update or rewrite the resume if it is not on par with a standard resume. The satisfactory resumes are then classified according to the job description.

2. LITERATURE SURVEY

This section emphasizes mainly on the recent research carried out on resume classification and recommendations.

An automated resume ranking system, or screening system, is a type of bias identification mechanism that uses natural language processing (NLP) to produce human-readable ratings of an individual's educational, professional, and personal attributes that have been analyzed using a search query [10].

The resume ranking system relies on human contexts such as job titles, company names, and geography. This method measures individual attributes such as how the individual is perceived and how it is interpreted by others [11]. The Resume ranking system also known as job role assessment, is a method of matching users to positions using language patterns from resumes and job applications.

Manual assessment or screening is somewhat biased because it relies heavily on how a person perceives or interprets the resume, whereas an automated resume screening system with a machine learning approach will remove the bias in screening the resumes. NLP (Natural Language Processing) can be utilized to match job applicants with job openings based on a keyword [12].

Tejaswini et al. developed a resume ranking system [13], which used the KNN approach to rank and pick resumes from the available resume dataset. The best candidates can be identified using content-based suggestion, which selects and ranks a large number of Curriculum Vitae (CV) based on job descriptions and uses cosine similarity to identify the CVs that are most similar to the provided job description. According to experimental findings, the proposed system performs with an average text parsing accuracy of 85% and a ranking accuracy of 92%.

A job recommendation system was developed by Mishra and Rathi using the enhanced DSSM (Deep Semantic Structure and modelling) [14]. The Deep Semantic Structure Modelling (DSSM) system employs semantic modelling of sparse data to increase system effectiveness by representing job descriptions and skill entities as character trigrams. The results of the DSSM model using two distinct datasets (Naukari.com and CareerBuilder.com) are compared, and the results of the proposed system are found to be better. Xavier initializer and Adam optimizer were used in the proposed approach.

In the paper titled “Text Analysis for Job Matching Quality Improvement” [15], the authors proposed an approach for the recruitment or staffing agency to match the quality of candidates to the job. Key factors like commute time, job location, job type, hourly rates, and skill set are matched for improved quality in identifying the candidate. Using a text-mining technique, the authors carefully examined text data that had been written by recruiters at a hiring agency in order to investigate these qualities. The authors were able to extract both positive and negative keywords that affected the matching result.

A content-based recommendation engine is developed, it finds the best possibilities for a user by matching their interests and talents to the requirements of a job ad. To provide the required suggestion, the recommended engine makes use of various text filters and feature similarity algorithms. Similarity algorithms use topic models and the bag of n-grams as parts of feature vectors [16].

Conventional job recommendation systems work on classifying the resume based on the data stored as tables or CSV or Excel files, collected through the job application portal or through web forms in the resume [17]. The proposed system extracts or scrapes the data from the resumes that are pdf files and the proposed system along with classification also recommends the resume for updates or rewriting by classifying the resume as unsatisfactory resumes. All the resumes are verified for their quality, and if the resume does not convey any clear information, then the resume is classified as unsatisfactory and other resumes that clearly states the skill set of the candidate are classified as satisfactory.

3. METHODOLOGY

The aim of the proposed system is to extract the data from the pdf files and to classify the resume as satisfactory and unsatisfactory resumes, then recommend rewriting to further strengthen the unsatisfactory resume and classify the satisfactory resumes based on the skill set.

3.1 Overview of the proposed system

The overview of the proposed system is depicted in Figure 1. The system can be fragmented into three major modules.

i) To Extract or scrape the essential data from the resume, that is in pdf format:

For this task 'PyPDF' package is used. PyPDF is a Python library that allows for the manipulation of PDF files. PyPDF provides functionality for extracting information from PDF documents by analyzing their internal structure.

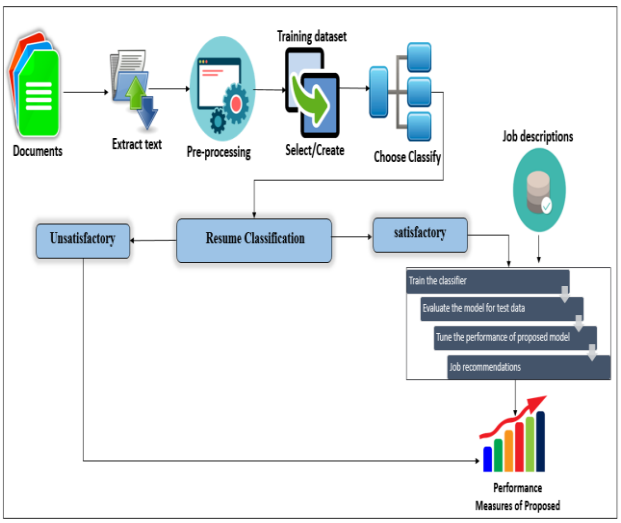

Figure 1. Resume recommendation and classification system overview

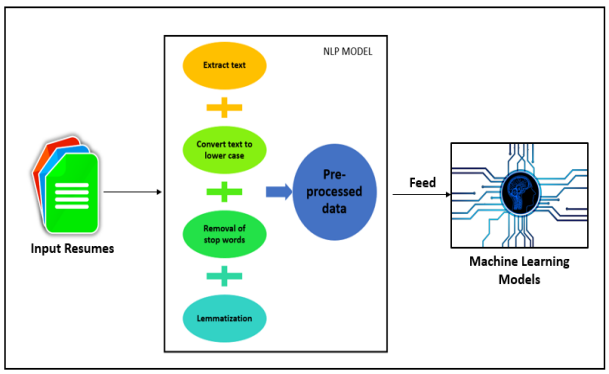

Figure 2. NLP model

To extract information from a PDF document using PyPDF, one would typically follow the following steps:

1. Open the PDF file using PyPDF.
2. Access the desired page or pages within the document.
3. Extract text or other content from the page(s).
4. Close the PDF file.

To accomplish these steps, PyPDF provides a variety of operational codes that can be used to interact with PDF documents. Some of the key operational codes for extracting information from PDF documents using PyPDF include:

- **PdfFileReader**: A class that represents a PDF file and provides functionality for reading and parsing the file.
- **getPage**: A method of the **PdfFileReader** class that allows for the selection of a specific page within the document.
- **extractText**: A method of the **PdfFileReader** class that extracts text from a selected page.
- **numPages**: A property of the **PdfFileReader** class that provides the number of pages in the PDF document.

By using these operational codes, PyPDF enables users to extract information from PDF documents and perform a variety of other operations on the files.

The first step is to install the 'PyPDF' package, once the package is installed, the next step is to import the required libraries and open the PDF file using the 'PyPDF' package.

Next, we need to extract the text from the PDF file. We can do this by iterating through each page of the PDF file and extracting the text using the 'extractText()' function.

After the text is extracted, it may contain unwanted characters or formatting, which needs to be cleaned. Pre-processing can be done to clean the extracted text. This may include removing special characters, removing stop words, and tokenizing the text.

In conclusion, the 'PyPDF' package is a useful tool to extract essential data from resumes in PDF format. Once the text is extracted, it can be further processed to clean and refine the data, making it useful for various applications, including candidate screening and job matching.

ii) To classify the resume into satisfactory and unsatisfactory resumes:

The data extracted from the pdf is pre-processed so that unnecessary data that does not impact the result is removed. The important function of this pre-processing is to extract only useful data. NLP (Natural Language Processing) is utilized in pre-processing. The process of pre-processing using NLP is depicted in Figure 2 and the algorithm is discussed as follows:

Algorithm for Pre-processing the data extracted using **PyPDF**

Input: Text extracted from a PDF document using the **PyPDF** package
Output: Pre-processed text that is cleaned and refined for further use

---

1. Remove special characters from the text using regular expressions
   - Initialize a regular expression pattern to match all non-alphanumeric characters
   - Use the **re.sub()** function to replace all matches with a space character
2. Convert the text to lowercase
   - Use the **lower()** method to convert all characters to lowercase
3. Remove stop words from the text using the **nltk** package
   - Initialize a set of stop words using the **stopwords.words('english')** method
   - Tokenize the text using the **word_tokenize()** function from the **nltk** package
   - Use a list comprehension to remove all stop words from the tokenized text
4. Tokenize the text using the **nltk** package
   - Use the **word_tokenize()** function to split the text into individual tokens
5. Perform stemming using the Porter Stemmer algorithm from the **nltk** package
   - Initialize a **PorterStemmer()** object
   - Use a list comprehension to apply the stemming algorithm to each token in the tokenized text
6. Return the pre-processed text as a list of stemmed tokens

Further, the classification is performed using machine learning models. Four benchmark models namely logistic regression [18], Gaussian Naïve Bayes [19], SVM, and Decision tree Classifier [20] are trained using a set of satisfactory and unsatisfactory resumes. Once the model is trained, it is used for the classification of the resumes as satisfactory and unsatisfactory. Satisfactory resumes are considered for job recommendations and unsatisfactory resumes are recommended for upgrade or rewrite. The satisfaction of a training structure in machine learning approaches can be judged based on several criteria, including:

1. Accuracy: This measure how well the model predicts the correct class labels for the test data. Higher accuracy indicates better performance.
2. Precision: This is the ratio of true positive predictions to the total number of positive predictions. It measures the ability of the model to identify true positives and avoid false positives.
3. Recall: This is the ratio of true positive predictions to the total number of actual positive instances in the test data. It measures the ability of the model to identify all positive instances, including those that might be missed.
4. F1 Score: This is the harmonic mean of precision and recall. It provides a balanced measure of both metrics and is a good indicator of overall performance.

The results of the models are recorded for accuracy, Precision, Recall, and F1Score and later compared in the results section.

The output from the NLP model is fed to the machine learning models. To better understand the standard of the resume, a word cloud is generated. Word cloud is a way of visualizing content or data. The most frequently used words in the resume will be generated as a word cloud for a better understanding of what skill set the candidate has. Figure 3 shows the word cloud of an unsatisfactory resume where the recruiter could not come to a conclusion on the expertise of the candidate, whereas the word cloud depicted in Figure 4 clearly showcases the expertise of the applicant.

iii) Classify satisfactory resumes for job recommendations based on skill set:

The skill set of the satisfactory resumes extracted using NLP is stored and the distinct skills are identified. For instance, the skillset required for Business analytics is extracted from the set of business analytics resumes as [‘Tableau’, ’Pandas’, ‘Artificial Intelligence’, ’Pyspark’, ‘IBM Watson’, ’Sklearn’, ‘R programming’, ‘Keras’, 'Map Reduce’, ’Modelling’,

Figure 3. Word cloud generated for an unsatisfactory resume

Figure 4. Word cloud generated for a satisfactory resume

’Regression’, ‘Dundas BI’, ’Patterns’, ’Data Mining’, ‘Plotly’, ’Text Mining’, ’Oops’, ’Deep Learning’, ’Web Analytics’, ’Time Series’, ’Regression’, ’Tensorflow’, ’Azure’, ’Linear Regression’, ’Logistic Regression’, ’Decision Tree’, ’Random Forest’, ’Data Structure’, ‘Keras’]. The skillset likely grows with the increase in the number of resumes used for training. As more resumes for a specific job role are used for training the model becomes more effective. The resumes are categorized based on the extracted data for each job description and the distribution of the categories is visualized in the results section. The job recommendations are done based on the category of the resume. The job description or requirement is used as the input and the appropriate candidate’s resumes are fetched as recommendations based on the extracted info. The performance of the models is measured using accuracy, where accuracy is the number of correct predictions among the total predictions made, it is calculated using Eq. (1) as below.

$$Accuracy = \frac{Number\ of\ predictions\ made\ correctly}{total\ number\ of\ predictions\ made} \quad (1)$$

The percentage of accurate predictions made by the model is measured by accuracy. Accuracy alone might not be sufficient in cases with unbalanced data, where the proportion of instances in each class differs dramatically.

What percentage of the positive forecasts is actually positive, according to a criterion called precision? It shows how well the model is determining a class's true positive values.

What percentage of all the real positive records are accurately identified, according to recall? It shows how successfully a model can locate each instance that belongs to a class. Recall is calculated for each class in multiclass classification tasks and then averaged using either a macro-average or a micro-average.

Recall is a metric used to measure the ability of a model to identify all relevant instances of a particular class. Macro-average and micro-average are two ways to calculate the average recall score across all the classes in a multiclass classification task.

In macro-average recall, the recall score for each class is first calculated separately, and then the average of those scores is taken. This approach treats each class equally, regardless of its size or frequency in the dataset.

On the other hand, in micro-average recall, the recall score for each class is calculated and then averaged, taking into

account the size or frequency of each class. This approach gives more weight to classes that have more instances in the dataset.

The main difference between macro and micro-averaging is that macro-averaging gives equal weight to each class, while micro-averaging gives more weight to larger classes. Therefore, if there is a class imbalance, micro-average recall would be more appropriate than macro-average recall. In the proposed system, recall is calculated using macro average.

The weighted harmonic mean of precision and recall is known as the F-Score, and a score of 1 denotes the highest possible precision and recall values. It is a helpful indicator for assessing the model's overall performance.

4. RESULTS AND DISCUSSIONS

4.1 Dataset exploration

Two datasets are used in the proposed approach, one for classifying satisfactory and unsatisfactory resumes. One for categorization of the resumes as per the job description. A total of 150 resumes are used for classifying satisfactory and unsatisfactory resumes, 75 for each category. A total of 500 resumes are used for training the model for job categorization with 141 job descriptions. For every job description, a set of skill sets is stated that has around 30 to 50 keywords. The skill set generated by the model is compared with a reputed job portal website https://www.hireitpeople.com/resume-database/ and found that the skillset fetched using the trained models is above par as per the requirement.

4.2 Model evaluation

The resumes are segregated as satisfactory and unsatisfactory and the accuracy of the models is compared. Logistic regression and Gaussian Naive Bayes perform better among the four models. Table 1 shows the accuracy attained among the models.

The frequently used words in the resume will be extracted for a better understanding of what skill set the person has. The algorithm used to find the frequency distribution of words is given as follows.

1. Read the sample resume file and store it as a string variable.
2. Clean the text data by removing any non-alphanumeric characters (punctuation, special characters, etc.), and convert all text to lowercase.
3. Tokenize the cleaned text into individual words.
4. Create an empty dictionary to store the frequency count of each word.
5. Loop through each word in the list of tokenized words:

a. If the word is not already in the dictionary, add it with a value of 1.

b. If the word is already in the dictionary, increment its value by 1.

6. Sort the dictionary by value in descending order to get a list of the most frequently used words.
7. Output the list of words and their frequency count.

Most frequent words are plotted as a frequency distribution graph for a sample resume as depicted in Figure 5.

The resumes are categorized based on the job categories and once the job description is given as the input, the resumes applicable for the job are fetched and given. The number of resumes belonging to the top 30 categories is displayed in Figure 6 as a bar graph.

Table 1. Accuracy of the models

S. No	Model Name	Accuracy
1	Logistic Regression	0.9763
2	Gaussian Naive Bayes	0.9516
3	SVM	0.333333
4	Decision Tree Classifier	0.8943

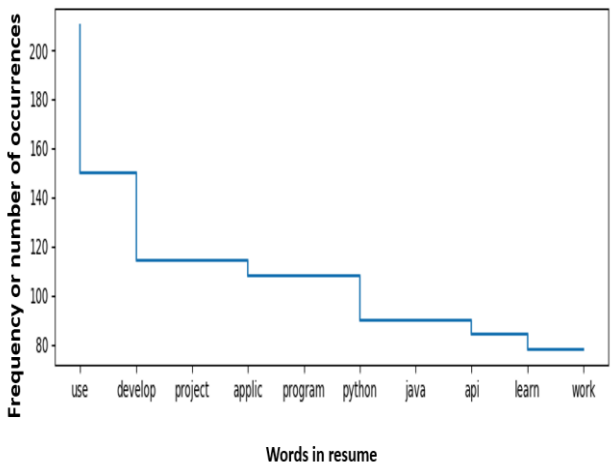

Figure 5. Frequency distributions of words in a sample resume

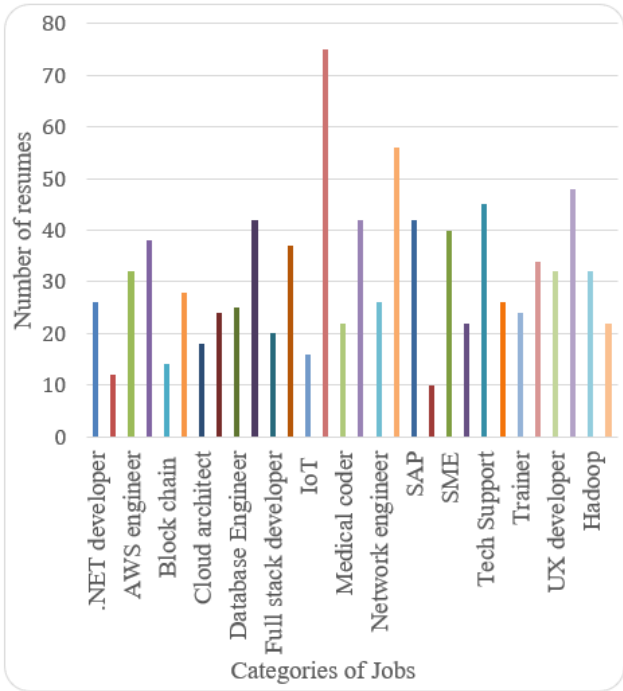

Figure 6. Resume categories with their numbers

Several resumes fall under multiple categories for example a same resume is applicable for backend developer and also for database engineers, similarly the same resume that is categorized under subject matter expert (SME) is applicable under other categories also. Hence there is an overlap in several categories. The accuracy of models was tested with a

different set of resumes. The accuracy measure is depicted in Figure 7. As the number of resumes increase during training per category the accuracy also varied. For instance, with less number of resumes logistic regression performed well, when the number of resumes increased per category, Gaussian Naive Bayes performed well. The category of PHP developer was considered for reference and a total of 30 resumes were used for training and 15 resumes were tested for accuracy. Thus logistic regression model can be utilized when less number of resumes are there and Gaussian Naive Bayes can be utilized when more resumes are available for training the data.

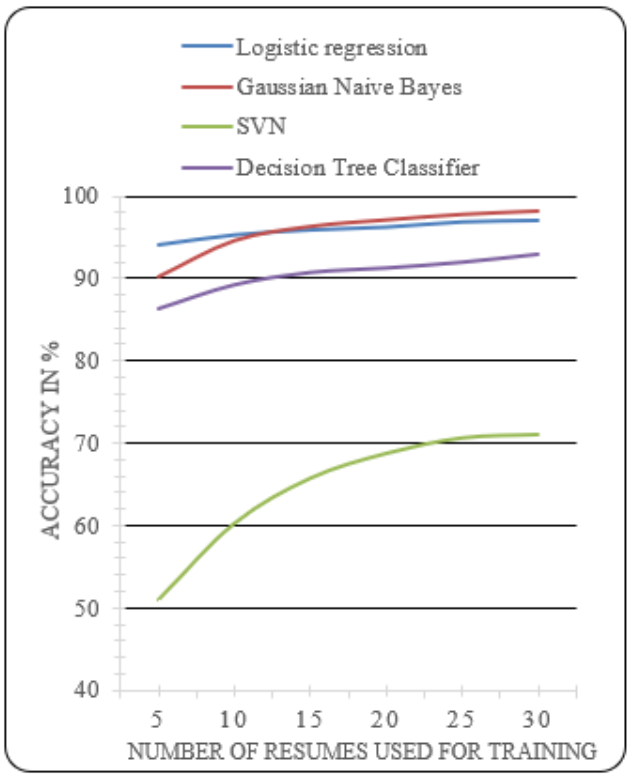

Figure 7. The accuracy measure

From the table, it is observed that the accuracy of models SVM and Decision Tree Classifier is lower than the accuracy of models Logistic Regression and Gaussian Naive Bayes.

The reason for the lower accuracy of model SVM could be because SVM is a linear classifier that works well when there is a clear linear separation between the classes. If the classes are not linearly separable, SVM may not perform well. In the context of resume classification, as the features are not linearly separable, such as there are overlapping skills or qualifications between job categories, hence SVM suffers to achieve the accuracy and it is not be the best model.

The reason for the lower accuracy of model Decision Tree Classifier could be because decision trees are prone to overfitting when the tree depth is too large or when there are too many features. Overfitting occurs when the model fits the training data too well, and as a result, it fails to generalize to new data. In the context of resume classification, possibility of the decision tree model for overfitting the training data is high, hence it is not able to classify new resumes accurately.

As discussed earlier accuracy calculation alone does not prove the system to be effective the other metrics precision, recall and F-measure were also measured and depicted as in the following Table 2 and also as a graph in Figure 8.

Table 2. Performance measures of the models

S. No	Model Name	Precision	Recall	F-Measure
1	Logistic Regression	1.0	0.99	0.99
2	Gaussian Naive Bayes	0.99	0.97	0.97
3	SVM	0.49	0.35	0.37
4	Decision Tree Classifier	0.96	0.91	0.92

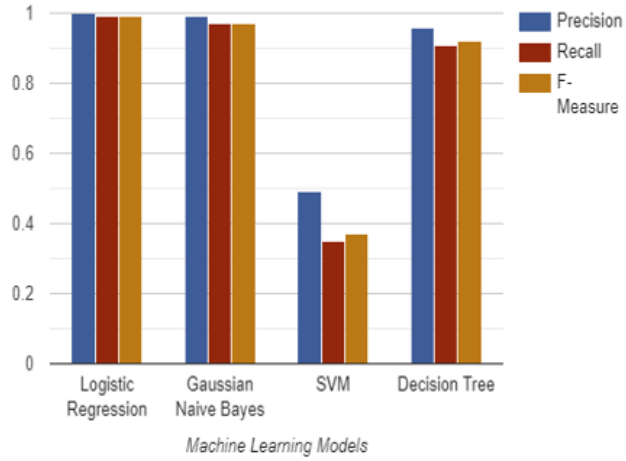

Figure 8. Performance measures of the ML models

The performance of the models remains quite similar for all the job categories. With a resume that can be applicable for multiple jobs and with the increase in the number of applicants with multiple skill sets, the process of resume screening will lead to high complexity.

5. CONCLUSION AND FUTURE SCOPE

The proposed system was able to extract the skill sets of a candidate from the resume which is in the pdf format using NLP. The NLP extracts the skillsets precisely which is evaluated for the standard resume. The machine learning models to successfully classify the satisfactory and unsatisfactory resume are implemented and the accuracy of the models were compared. Categorization of resumes for various job categories is also realized and it is observed that Gaussian Naïve Bayes and logistic regression perform well over the other two models.

The Proposed system would ease the resume screening process by reducing and recommending unsatisfactory resumes from the total resumes available. The resume categorization is also performed so that the screening will bring only the appropriate resume for the job description searched.

The system can be further enhanced by splitting the resume into sections and exactly predicting the sections of the resume to be strengthened so that the resume gains more visibility among the recruiters and falls under the satisfactory category in the next screening.

Multilingual support: The proposed system is currently designed for resumes in English. Multilingual support can be added to enable screening of resumes in different languages.

REFERENCES

[1] Villeda, M., McCamey, R., Essien, E., Amadi, C. (2019). Use of social networking sites for recruiting and selecting in the hiring process. International business research, 12(3): 66-78. http://dx.doi.org/10.5539/ibr.v12n3p66

[2] Roy, P.K., Chowdhary, S.S., Bhatia, R. (2020). A Machine Learning approach for automation of Resume Recommendation system. Procedia Computer Science, 167: 2318-2327. https://doi.org/10.1016/j.procs.2020.03.284

[3] Alamelu, M., Kumar, D.S., Sanjana, R., Sree, J.S., Devi, A.S., Kavitha, D. (2021). Resume validation and filtration using natural language processing. In 2021 10th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON), Jaipur, India, pp. 1-5. https://doi.org/10.1109/IEMECON53809.2021.9689075

[4] Ali, I., Mughal, N., Khand, Z.H., Ahmed, J., Mujtaba, G. (2022). Resume classification system using natural language processing and machine learning techniques. Mehran University Research Journal of Engineering & Technology, 41(1): 65-79. http://dx.doi.org/10.22581/muet1982.2201.07

[5] Poovizhi, P., Ezhilarasi, K., Gayathri, G., Megala, R., Anisha, D. (2022). Automatic scraping of employment record using machine learning—An assistance for the recruiter. In Smart Data Intelligence, pp. 561-577. https://doi.org/10.1007/978-981-19-3311-0_4

[6] Nimbekar, R., Patil, Y., Prabhu, R., Mulla, S. (2019). Automated resume evaluation system using NLP. In 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), Mumbai, India, pp. 1-4. https://doi.org/10.1109/ICAC347590.2019.9036842

[7] Liddy, E. D. (2001). Natural language processing. https://surface.syr.edu/istpub/63/.

[8] Sinha, A.K., Akhtar, A.K., Kumar, A. (2021). Resume screening using natural language processing and machine learning: A systematic review. Machine Learning and Information Processing, 207-214.

[9] Goyal, U., Negi, A., Adhikari, A., Gupta, S.C., Choudhury, T. (2021). Resume data extraction using NLP. In Innovations in Cyber Physical Systems, pp. 465-474. https://doi.org/10.1007/978-981-16-4149-7_41

[10] Ransing, R., Mohan, A., Emberi, N.B., Mahavarkar, K. (2021). Screening and Ranking Resumes using Stacked Model. In 2021 5th International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques (ICEECCOT), Mysuru, India, pp. 643-648. https://doi.org/10.1109/ICEECCOT52851.2021.9707977

[11] Bhor, S., Gupta, V., Nair, V., Shinde, H., Kulkarni, M.S. (2021). Resume parser using natural language processing techniques. International Journal of Research in Engineering and Science (IJRES), 9(6): 01-06.

[12] Kadam, R., Suhas, G., Mukri, U., Khandare, S. (2022). NLP-Based Resume Screening and Job Recruitment Portal. In Data Intelligence and Cognitive Informatics, pp. 1-21. Springer, Singapore. http://dx.doi.org/10.1007/978-981-16-6460-1_1

[13] Tejaswini, K., Umadevi, V., Kadiwal, S.M., Revanna, S. (2022). Design and development of machine learning based resume ranking system. Global Transitions Proceedings 3(2): 371-375. https://doi.org/10.1016/j.gltp.2021.10.002

[14] Mishra, R., Rathi, S. (2022). Enhanced DSSM (deep semantic structure modelling) technique for job recommendation. Journal of King Saud University-Computer and Information Sciences, 34(9): 7790-7802. https://doi.org/10.1016/j.jksuci.2021.07.018

[15] Kino, Y., Kuroki, H., Machida, T., Furuya, N., Takano, K. (2017). Text analysis for job matching quality improvement. Procedia computer science, 112: 1523-1530. https://doi.org/10.1016/j.procs.2017.08.054

[16] Kumar, N., Gupta, M., Sharma, D., Ofori, I. (2022). Technical job recommendation system using APIs and web crawling. Computational Intelligence and Neuroscience, 2022(5): 1-11. http://dx.doi.org/10.1155/2022/7797548

[17] Sridevi, G.M., Suganthi, S.K. (2022). AI based suitability measurement and prediction between job description and job seeker profiles. International Journal of Information Management Data Insights, 2(2): 100109. https://doi.org/10.1016/j.jjimei.2022.100109

[18] Muhajir, D., Akbar, M., Bagaskara, A., Vinarti, R. (2022). Improving classification algorithm on education dataset using hyperparameter tuning. Procedia Computer Science, 197: 538-544. https://doi.org/10.1016/j.procs.2021.12.171

[19] Mwaro, P.N., Ogada, K., Cheruiyot, W., (2020). Applicability of Naïve Bayes model for automatic resume classification. International Journal of Computer Applications Technology and Research 9(9): 257-264. http://dx.doi.org/10.7753/IJCATR0909.1002

[20] Swami, P., Pratap, V. (2022). Resume classifier and summarizer. In 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), Faridabad, India, pp. 220-224). https://doi.org/10.1109/COM-IT-CON54601.2022.9850527