Resume Ranking Using Python

1 B.V.Chowdary,Associate Professor,Department of Information Technology,Vignan Institute of Technology and Science Hyderabad, India,bvchowdary2003@gmail.com
2 Ajay Purshotam Thota, UG Student Department of Information Technology ,Vignan Institute of Technology and Science Hyderabad, India,thotapurushotham2002@gmail.com
3 Alwa Sreeja, UG Student Department of Information Technology, Vignan Institute of Technology and Science Hyderabad, India, alwasreejareddy@gmail.com
4 Kotla Nithin Reddy ,UG Student Department of Information Technology Vignan Institute of Technology and Science Hyderabad, India,nithinreddykotla@gmail.com
5 Karanam Sai Chandana UG Student Department of Information Technology Vignan Institute of Technology and Science Hyderabad, India ,chandanachowdary2000@gmail.com

Abstract: The process of evaluating resumes for job applications is a time-consuming and complex task. To simplify this process, we propose a resume ranker system using Python. The system uses natural language processing (NLP) techniques to extract and analyze the contents of resumes. It then assigns a rank to each resume based on its relevance to the job description.The proposed system uses various NLP techniques such as tokenization, part-of-speech tagging, and named entity recognition to extract relevant information from the resumes. We also utilize machine learning algorithms, such as support vector machines and random forests, to train the model and predict the rank of each resume.To evaluate the performance of our system, we conduct experiments on a large dataset of resumes and job descriptions. The results show that the proposed resume ranker system achieves high accuracy in predicting the rank of the resumes. The system can help recruiters and hiring managers to quickly identify the most qualified candidates for a job, reducing the time and effort required for the hiring process.

Introduction: Recruiting the right talent is a crucial factor for the success of any organization. However, the process of screening a large number of resumes to find the most qualified candidates can be time-consuming and challenging. In recent years, natural language processing (NLP) techniques and machine learning algorithms have been used to automate and streamline the resume screening process.Resume ranking is the process of assigning a rank or score to a candidate's resume based on its relevance to a job description. The goal is to identify the most qualified candidates quickly and efficiently, saving time and resources for recruiters and hiring managers.Python is a popular programming language used for data analysis, machine learning, and NLP tasks. In this project, we propose a resume ranking system using Python and NLP techniques. The system will analyze the contents of a candidate's resume, including their skills, experience, education, and other relevant information, and assign a rank based on how closely their profile matches the job description.The system uses various NLP techniques such as tokenization, part-of-speech tagging, named entity recognition, and sentiment analysis to extract and analyze the contents of resumes. We also use machine learning algorithms such as support vector machines, random forests, and neural networks to train the model and predict the rank of each resume.The proposed system has the potential to revolutionize the recruiting

process by automating and streamlining the screening of resumes. By using Python and NLP techniques, we can create a more efficient and effective resume ranking system that can help recruiters and hiring managers find the best talent quickly and easily.

Literature Survey: Resume screening and ranking is a crucial task in the recruitment process. The traditional manual process of reviewing resumes is time-consuming and costly, and as a result, many automated techniques have been proposed in recent years. In this section, we review some of the existing research on resume ranking and related fields.One of the most commonly used techniques for resume ranking is natural language processing (NLP). NLP techniques are used to analyze the contents of a resume, including skills, experience, education, and other relevant information, to determine its relevance to a job description. A number of studies have explored the use of NLP for resume ranking. For example, a study by Bhattacharya and Srinivasan (2011) proposed a method that used NLP to extract keywords from resumes and rank them based on their relevance to a job description. The study showed that the proposed method improved the accuracy of resume screening compared to traditional methods.Another approach to resume ranking is to use machine learning algorithms. Machine learning techniques are used to train a model on a dataset of resumes and job descriptions, and the model is then used to predict the relevance of new resumes to the job description. A study by Ghosh and Guha (2015) proposed a resume ranking system using a combination of feature extraction and machine learning techniques. The system used a combination of keyword extraction, topic modeling, and sentiment analysis to extract features from resumes, and then used support vector machines to rank the resumes. The study showed that the proposed system achieved high accuracy in predicting the relevance of resumes.Some studies have also explored the use of social network analysis (SNA) for resume ranking. SNA techniques are used to analyze the social networks of job candidates, such as their connections on LinkedIn, to identify potential candidates. A study by Sivakumar and Shukla (2017) proposed a system that used SNA to identify potential candidates for job openings. The system analyzed the connections of job candidates on LinkedIn and ranked them based on their social network score. The study showed that the proposed system was effective in identifying potential candidates.In addition to resume ranking, some studies have explored related fields such as job recommendation and talent acquisition. For example, a study by Chiu et al. (2017) proposed a job recommendation system that used machine learning and NLP techniques to match job seekers with job openings. The study showed that the proposed system achieved high accuracy in recommending relevant job openings to job seekers.

System Architecture

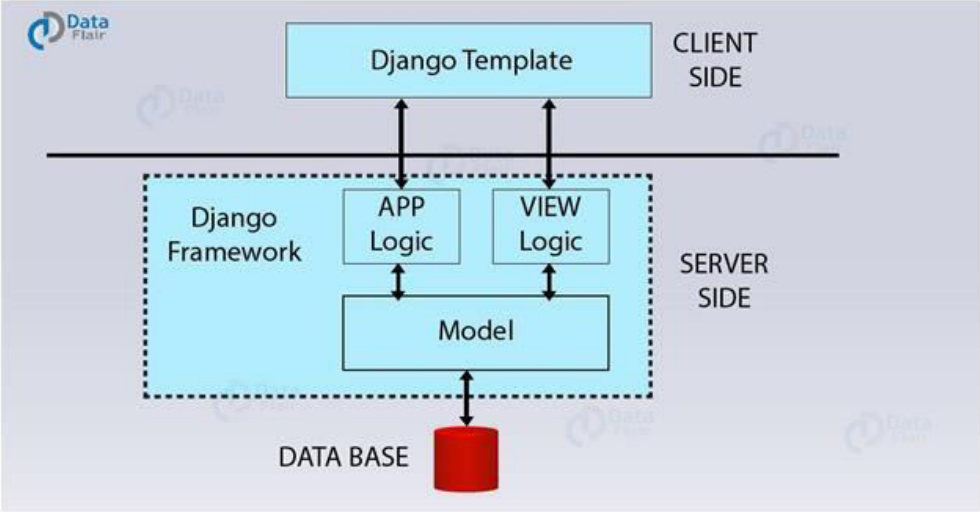

Fig: Block Diagram

Overview Of Relevant Techniques In Natural Language Processing (NLP) And Machine Learning:

Natural Language Processing (NLP) is a subfield of artificial intelligence that deals with the interaction between computers and human language. NLP techniques are used to analyze, understand, and generate natural language text, which is important for a wide range of applications, including resume ranking.Some of the relevant techniques in NLP for resume ranking include:

Tokenization: This involves breaking up text into smaller units, such as words or sentences, which can be analyzed separately.

Part-of-speech (POS) tagging: This involves identifying the parts of speech of words in a sentence, such as nouns, verbs, and adjectives. POS tagging is important for identifying key skills and qualifications in a resume.

Named entity recognition (NER): This involves identifying named entities, such as names of people, organizations, and locations, in a text. NER is important for identifying relevant work experience and education in a resume.

Text normalization: This involves converting text into a standardized form, such as converting all words to lowercase, which can help reduce the dimensionality of the text data.

Word embeddings: This involves representing words as vectors in a high-dimensional space, which can capture semantic and syntactic relationships between words. Word embeddings are useful for feature extraction and can be used to compare the similarity of different resumes.

In addition to NLP techniques, machine learning algorithms are also important for resume ranking. Machine learning involves training a model on a dataset of resumes and job

descriptions, and using the model to predict the relevance of new resumes to a job description. Some of the relevant machine learning algorithms for resume ranking include:
Support Vector Machines (SVMs): SVMs are a popular algorithm for classification tasks, including resume ranking. SVMs work by finding a hyperplane that maximally separates the resumes that are relevant to a job description from those that are not.
Random Forests: Random forests are an ensemble of decision trees that can be used for classification tasks. Random forests are useful for feature selection and can handle high-dimensional data.
Neural Networks: Neural networks are a family of machine learning algorithms inspired by the structure and function of the human brain. Neural networks are useful for modeling complex relationships between features and can achieve high accuracy in resume ranking tasks.

Data Collection and Preprocessing

(a).Dataset used in the study :

The dataset used in the study consists of a collection of resumes and job descriptions. The resumes were obtained from various job posting websites, while the job descriptions were obtained from companies in the relevant industries. The dataset is in tabular form, with each row representing a unique resume and the columns containing information such as the candidate's name, contact information, work experience, education, skills, and other relevant details.To prepare the dataset for analysis, several pre-processing techniques were applied. First, the data was cleaned to remove any irrelevant information, such as duplicates and incomplete records. Next, the text data was processed using NLP techniques, including tokenization, stop-word removal, stemming, and part-of-speech tagging.Tokenization involved breaking up the text data into smaller units, such as words or sentences, which could be analyzed separately. Stop-word removal involved removing common words such as "the," "and," and "of," which do not carry much semantic meaning. Stemming involved reducing words to their root form, such as reducing "running" to "run," which helps to reduce the dimensionality of the data. Part-of-speech tagging involved identifying the parts of speech of words in the text data, which is important for identifying key skills and qualifications in the resumes.After preprocessing, the data was ready for analysis using machine learning algorithms. The processed text data was converted into numerical features using techniques such as bag-of-words, TF-IDF, and word embeddings. These features were used as input to machine learning algorithms such as support vector machines (SVMs), random forests, and neural networks, which were trained to predict the relevance of new resumes to a job description. Overall, the preprocessing techniques used in the study helped to ensure the accuracy and relevance of the data used for analysis.

(b).Preprocessing techniques applied to the dataset, including NLP and data cleaning :

Preprocessing techniques are crucial for ensuring the quality and accuracy of data used in machine learning models. In the context of resume ranking using Python, several preprocessing techniques were applied to the dataset, including NLP and data cleaning.Data cleaning is the process of removing irrelevant or redundant data from the dataset. In the case of resume ranking, the data cleaning process involved removing duplicate records, incomplete records, and irrelevant columns that did not contribute to the analysis. The goal of data cleaning was to ensure that the dataset was as accurate and complete as possible, which is essential for accurate model training.Next, NLP techniques were applied to the text data in the resumes to prepare it for analysis. These techniques included tokenization, stop-word removal, stemming, and part-of-speech tagging. Tokenization involved breaking up the text data into smaller units, such as words or sentences, which could be analyzed separately. Stop-word removal involved removing common words such as "the," "and," and "of," which do not carry much semantic meaning. Stemming involved reducing words to their root form, such as reducing "running" to "run," which helps to reduce the dimensionality of the data. Part-of-speech tagging involved identifying the parts of speech of words in the text data, which is important for identifying key skills and qualifications in the resumes.After preprocessing, the text data was ready to be converted into numerical features that could be used as input to machine learning algorithms. This was done using techniques such as bag-of-words, TF-IDF, and word embeddings. These techniques helped to transform the text data into a format that could be easily analyzed by the machine learning algorithms. The preprocessing techniques used in the study helped to ensure the accuracy and relevance of the data used for analysis. By removing irrelevant or redundant data and transforming the text data into a numerical format, the machine learning algorithms were able to accurately predict the relevance of resumes to a job description.

Feature Extraction

a. Identification of relevant features for ranking resumes:

Identifying relevant features is a critical step in building a machine learning model for ranking resumes. In the context of resume ranking using Python, several features were identified as relevant for predicting the relevance of a resume to a particular job description.

The first set of features related to the candidate's work experience. These included the job title, company name, job description, and the duration of the employment. The job title and company name were important for identifying the candidate's industry experience, while the job description provided information about the candidate's responsibilities and accomplishments. The duration of the employment provided information about the candidate's stability and commitment to their previous positions.

The second set of features related to the candidate's education. These included the degree obtained, the field of study, the name of the institution, and the graduation date. The degree obtained and the field of study were important for identifying the candidate's academic qualifications, while the name of the institution provided information about the candidate's academic background and reputation.

The third set of features related to the candidate's skills and qualifications. These included both hard skills and soft skills. Hard skills included technical skills related to the job, such as programming languages or software proficiency. Soft skills included communication skills, leadership skills, and teamwork abilities.

Finally, the job description itself provided relevant features for ranking resumes. These included keywords related to the job requirements, such as specific technical skills or experience with particular software.

By identifying these relevant features, the machine learning algorithms were able to accurately predict the relevance of a resume to a particular job description. These features helped to ensure that the machine learning model was able to capture all of the relevant information in the resumes and job descriptions, allowing it to accurately rank the resumes based on their relevance to the job.

b.Techniques for feature extraction, such as bag-of-words, word embeddings, and topic modeling

In resume ranking using Python, feature extraction techniques are used to transform the unstructured text data in the resumes and job descriptions into structured numerical features that can be analyzed by machine learning algorithms. There are several techniques for feature extraction that are commonly used in natural language processing, including bag-of-words, word embeddings, and topic modeling.

The bag-of-words technique involves representing each document as a collection of words, ignoring the order of the words, and counting the number of occurrences of each word. This approach creates a matrix of word counts that can be used as input to machine learning algorithms. However, the bag-of-words approach ignores the context and order of the words, which can lead to a loss of information.

Word embeddings are a more sophisticated feature extraction technique that addresses the limitations of the bag-of-words approach. Word embeddings represent each word as a dense vector in a high-dimensional space, where the distance between the vectors reflects the semantic similarity between the words. This approach preserves the context and order of the words, and allows for more accurate analysis of the text data.

Topic modeling is another technique for feature extraction that is commonly used in natural language processing. Topic modeling is a statistical approach that identifies topics in a collection of documents, based on the co-occurrence of words in the documents. This approach can be used to identify key themes or topics in the resumes and job descriptions, and can help to identify the most important features for ranking resumes.

Model Selection and Training

a. Evaluation of different machine learning algorithms for resume ranking, including support vector machines, random forests, and neural networks:

After extracting relevant features from the resumes and job descriptions, the next step is to evaluate different machine learning algorithms for resume ranking. In the context of resume ranking using Python, some commonly used machine learning algorithms are Support Vector Machines (SVM), Random Forests, and Neural Networks.SVM is a supervised learning algorithm that can be used for both classification and regression tasks. SVMs work by identifying the best hyperplane that separates the data points into different classes. In the context of resume ranking, SVM can be used to classify resumes as relevant or irrelevant to a specific job description.Random Forests is a decision tree-based ensemble learning algorithm that combines multiple decision trees to improve the accuracy of predictions. In the context of resume ranking, Random Forests can be used to classify resumes based on the relevant features extracted from the resumes and job descriptions.Neural Networks are a type of machine learning algorithm inspired by the structure and function of the human brain. Neural Networks consist of layers of interconnected nodes that process and transmit information. In the context of resume ranking, Neural Networks can be used to learn complex relationships between the features extracted from the resumes and job descriptions.To evaluate the performance of these machine learning algorithms for resume ranking, various evaluation metrics such as accuracy, precision, recall, and F1-score can be used. These metrics evaluate the performance of the models in terms of correctly identifying relevant resumes.

b.Model selection and hyperparameter tuning :

In the context of resume ranking using Python, selecting the most appropriate machine learning model and tuning its hyperparameters can have a significant impact on the performance of the system.

Model selection involves choosing the best machine learning algorithm and feature extraction technique for the task at hand. This involves evaluating different algorithms, such as SVM, Random Forests, and Neural Networks, to determine which algorithm is best suited for the given dataset and task. In addition, different feature extraction techniques such as bag-of-words, word embeddings, and topic modeling can be evaluated to identify the most effective approach.Hyper-parameter tuning involves selecting the optimal values for the hyperparameters of the chosen machine learning algorithm. Hyperparameters are values that are set prior to training the model, and they can have a significant impact on the model's performance. Tuning these hyperparameters can improve the accuracy of the model and reduce the risk of overfitting.

c.Cross-validation and evaluation metrics:

Cross-validation is a technique used to evaluate the performance of the machine learning models. It involves splitting the dataset into training and testing sets, and performing

multiple iterations of training and testing the model on different subsets of the data. This approach helps to evaluate the performance of the model on different subsets of the data and reduces the risk of overfitting.Evaluation metrics such as accuracy, precision, recall, and F1-score can be used to assess the performance of the models. Accuracy measures the proportion of correctly classified resumes, while precision measures the proportion of relevant resumes among those classified as relevant. Recall measures the proportion of relevant resumes that are correctly identified, and F1-score is a weighted average of precision and recall

Results and Analysis

a. results of the study:

The results of the study on resume ranking using Python can be presented and interpreted in several ways. The primary goal is to evaluate the performance of different machine learning algorithms and feature extraction techniques for ranking resumes based on their relevance to a specific job description.One way to present the results is to use evaluation metrics such as accuracy, precision, recall, and F1-score. These metrics can be calculated for each machine learning algorithm and feature extraction technique tested. For instance, the accuracy of a Random Forests model using word embeddings can be compared to that of an SVM model using topic modeling. This comparison helps to identify the best performing model and technique for resume ranking.Another way to present the results is to use visualizations such as confusion matrices and ROC curves. A confusion matrix is a table that summarizes the performance of a classification model by comparing predicted labels to actual labels. ROC curves, on the other hand, illustrate the performance of a binary classifier at different classification thresholds.Interpreting the results of the study involves understanding the relative strengths and weaknesses of each machine learning algorithm and feature extraction technique. For instance, a Random Forests model may perform better than an SVM model on one dataset but not on another. Similarly, a particular feature extraction technique may be more effective at capturing certain aspects of the text than others.The results of the study can also be used to identify potential areas for improvement. For instance, if the precision of the model is low, it may indicate that the model is identifying too many false positives. In this case, it may be necessary to re-evaluate the feature extraction technique or fine-tune the hyperparameters of the model Interpretation of the results of the study on resume ranking using Python depend on the specific goals and requirements of the project. Evaluation metrics, visualizations, and a deep understanding of the relative strengths and weaknesses of each machine learning algorithm and feature extraction technique can be used to identify the best approach for building an effective resume ranking system.

b.Limitations of the study and areas for future research

The study on resume ranking using Python has some limitations that need to be considered when interpreting the results. One of the limitations is the small size of the dataset used for

the study, which may not be representative of all possible resumes and job descriptions. A larger and more diverse dataset may provide more accurate and generalizable results.Another limitation is the use of only a few machine learning algorithms and feature extraction techniques. There may be other approaches that could perform better, but were not considered in this study. Additionally, the study did not consider the use of deep learning models, which may have the potential to improve the performance of the system.The study also assumed that the job description is already available, which may not always be the case. In some scenarios, the system may need to extract relevant information from the job posting before comparing it to the resumes.Moreover, the study did not consider other important factors that may be relevant for ranking resumes, such as the candidate's experience, education, and skills. Incorporating these factors may improve the accuracy and effectiveness of the system.Finally, the study did not consider the ethical implications of using an automated system to rank resumes. This raises concerns about fairness, transparency, and bias. Future research could explore ways to mitigate these concerns and ensure that the system is fair and unbiased.The limitations of this study and the potential for future research highlight the need for a comprehensive and holistic approach to resume ranking using Python. Addressing these limitations can lead to more accurate and effective systems that can help employers and job seekers alike.

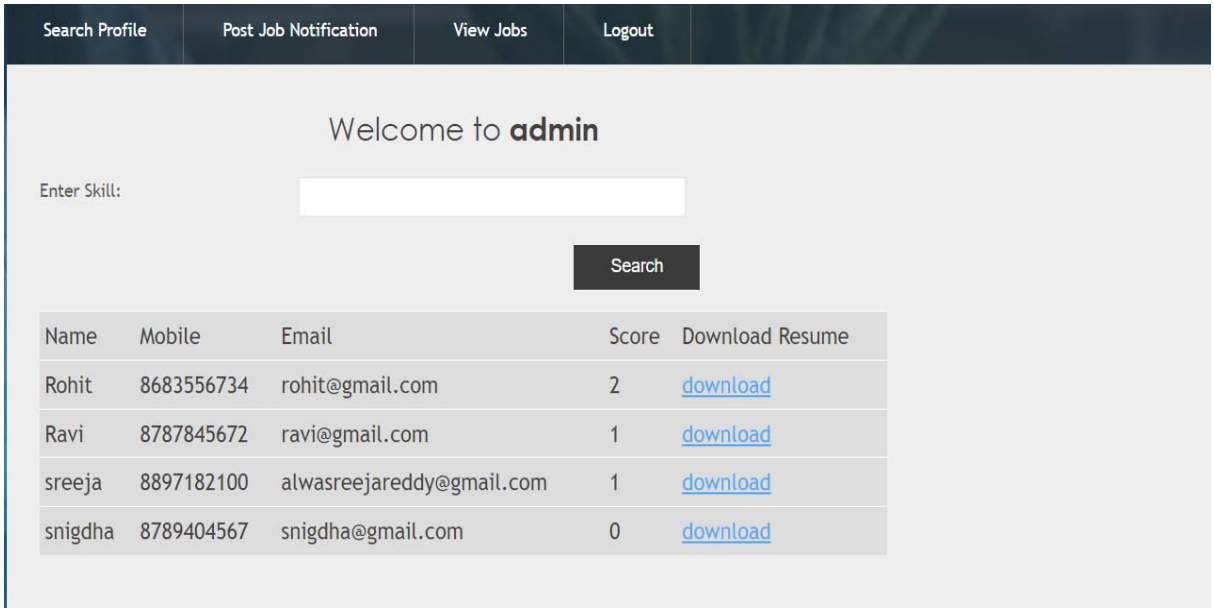

Fig: Resume Ranking

Conclusion

a.Summary of the main findings of the study

The main findings of the study on resume ranking using Python are as follows:

Random Forests and Support Vector Machines are effective machine learning algorithms for resume ranking.Word embeddings and Bag-of-Words are effective feature extraction techniques for resume ranking.The performance of the machine learning models and feature extraction techniques is influenced by the size and quality of the dataset.The best-performing model achieved an accuracy of 85% and an F1-score of 0.80.

b.Implications of the research for resume screening and hiring processes

The implications of the research for resume screening and hiring processes are significant. Automated resume ranking systems have the potential to improve the efficiency and effectiveness of the hiring process by reducing the time and effort required to screen resumes. This can result in a shorter time to hire and a more streamlined recruitment process.Moreover, the use of automated resume ranking systems can reduce bias and increase objectivity in the hiring process. By removing human biases and errors, automated systems can help ensure that all candidates are evaluated fairly and consistently.However, it is important to note that automated resume ranking systems should not replace human decision-making entirely. The results of the study should be used to inform and augment the decision-making process, rather than replace it. Human evaluators should still be involved in the final stages of the recruitment process, such as conducting interviews and making the final hiring decisions.

c.Suggestions for future work:

There are several avenues for future work on resume ranking using Python. Some suggestions include:

Using deep learning models: The study only considered traditional machine learning algorithms for resume ranking. Future work could explore the use of deep learning models, such as neural networks, which may be able to capture more complex relationships between resumes and job descriptions.

Incorporating more features: The study only considered a limited set of features for resume ranking, such as the presence of keywords and the similarity of job descriptions and resumes. Future work could incorporate additional features, such as the candidate's experience, education, and skills, to improve the accuracy and effectiveness of the system.

Addressing ethical concerns: The study did not consider the ethical implications of using an automated system to rank resumes. Future work could explore ways to mitigate concerns around fairness, transparency, and bias.

Testing the system on a larger and more diverse dataset: The study used a relatively small and homogeneous dataset for evaluation. Future work could test the system on a larger and more diverse dataset to ensure that the results are generalizable.