RESUME SCREENING USING NLP

A PROJECT REPORT

Submitted by

HARSHITH REDDY PATHPI

BABU VARSHAD KATIKIREDDY

Under the guidance of

Professor Mr. Vahid Behzadan

For the Course DSCI-6004-01

NATURAL LANGUAGE PROCESSING

A logo of a university

Description automatically generated

UNIVERSITY OF NEW HAVEN

WEST HAVEN, CONNECTICUT

SPRING 2024

Title:

RESUME SCREENING USING NLP

Abstract:

This paper presents a comprehensive approach to automate the classification of resumes using machine learning techniques. The proposed method aims to streamline the recruitment process by categorizing resumes into specific job domains, such as IT, finance, marketing, etc. Leveraging natural language processing (NLP) and machine learning algorithms, we preprocess textual resume data, extract relevant features, train classifiers, and evaluate performance. Experimental results demonstrate the efficacy of our approach, showcasing competitive classification accuracy compared to baseline methods.

Introduction:

In today's competitive job market, HR professionals and recruiters face the daunting task of efficiently processing and analyzing numerous resumes to identify suitable candidates. Manual resume screening is time-consuming and prone to errors, highlighting the need for automated solutions. This paper addresses the challenge of resume classification using machine learning techniques, presenting a robust framework to categorize resumes based on their content.

Objective**:**

The primary objective of our project is to design and implement a robust machine learning pipeline capable of categorizing resumes into relevant job categories automatically. By employing state-of-the-art NLP techniques and machine learning algorithms, we seek to achieve the following objectives:

* Develop a comprehensive dataset of labeled resumes spanning multiple job categories.
* Preprocess the textual data to remove noise and standardize the format for feature extraction.
* Extract informative features from the cleaned resume text using advanced techniques such as TF-IDF vectorization.
* Train and evaluate machine learning models, including but not limited to K-Nearest Neighbors (KNN) and support vector machines (SVM), for accurate classification.
* Fine-tune the model hyperparameters and optimize the classification performance using techniques such as grid search and cross-validation.
* Deploy the trained model as a scalable and efficient system for automated resume categorization, accessible via a user-friendly interface or API.

Technical Details:

Data Preprocessing:

Data preprocessing plays a crucial role in preparing textual resume data for classification. We employ a series of steps to clean and preprocess the raw resume text, ensuring that it is suitable for feature extraction and model training.

Text Cleaning:

The first step involves removing noise and irrelevant information from the textual data. We utilize regular expressions to identify and eliminate URLs, which often appear as hyperlinks or web addresses within resumes. Additionally, common Twitter-related terms such as "RT" (retweet) and "cc" (carbon copy) are removed to ensure that social media artifacts do not influence the classification process. Hashtags and user mentions are also stripped from the text, as they are typically associated with social media platforms and do not contribute to the content of the resume. Punctuation marks and special characters are removed to standardize the text format and facilitate subsequent processing steps. Finally, non-ASCII characters are replaced with whitespace to ensure compatibility with text processing algorithms.

Tokenization and Stopword Removal:

After cleaning the text, we tokenize the resume content into individual words using the nltk library's word tokenizer. Tokenization breaks down the text into a sequence of words, enabling further analysis at the word level. We then remove stopwords, which are common words that do not carry significant meaning in the context of resume classification. Stopwords such as "the," "and," "is," etc., are removed to focus on content-bearing words that contribute to the classification task.

Feature Extraction:

Feature extraction involves transforming the preprocessed textual data into numerical representations that can be used as input to machine learning algorithms. In our approach, we leverage the TF-IDF (Term Frequency-Inverse Document Frequency) vectorization technique to extract features from the cleaned resume text.

TF-IDF Vectorization:

TF-IDF is a statistical measure used to evaluate the importance of a term in a document relative to a collection of documents. It consists of two components: term frequency (TF) and inverse document frequency (IDF). TF measures the frequency of a term within a document, while IDF measures the rarity of a term across all documents in the corpus. The product of TF and IDF yields a weight that represents the importance of a term in distinguishing a document from others in the corpus. TF-IDF assigns higher weights to terms that are frequent in the document but rare across the corpus, thus capturing the discriminative power of terms.

Model Training:

With the preprocessed text represented as TF-IDF feature vectors, we proceed to train a machine learning classifier to categorize resumes into predefined job categories. In this section, we detail the training process of the K-Nearest Neighbors (KNN) classifier, a popular algorithm for classification tasks.

K-Nearest Neighbors (KNN) Classifier:

KNN is a non-parametric and instance-based learning algorithm used for classification and regression tasks. It operates on the principle of similarity, where the class of a sample is determined by the class labels of its nearest neighbors in the feature space. In the context of resume classification, KNN calculates the distance between a test sample (resume) and its k nearest neighbors in the feature space (TF-IDF vectors of training resumes). The class label of the test sample is then determined by a majority vote among its nearest neighbors. The choice of k, the number of neighbors considered, and the distance metric (e.g., Euclidean distance, cosine similarity) are hyperparameters that can be tuned to optimize classification performance.

Training Procedure:

To train the KNN classifier, we utilize the TF-IDF feature vectors extracted from the training set of labeled resumes. The classifier learns to associate each feature vector with its corresponding job category label through a process called supervised learning. During training, the algorithm adjusts its internal parameters based on the input-output pairs provided by the training data, optimizing its ability to classify unseen resumes accurately.

Results:

Experimental results on the test dataset demonstrate the effectiveness of our approach in accurately categorizing resumes into predefined categories. The trained KNN classifier achieves a competitive accuracy of over 85% on the test set, indicating its ability to generalize well to unseen data. Comparison with baseline methods reveals improvements in classification performance, underscoring the benefits of utilizing TF-IDF features and KNN classification.

A screen shot of a computer

Description automatically generated

Output from the model to screen top 3 resumes of each position.  
A screenshot of a computer

Description automatically generated

Analysis and Discussion:

The success of our approach can be attributed to the combination of effective data preprocessing, feature extraction, and model training techniques. By cleaning and tokenizing the resume text and converting it into TF-IDF feature vectors, we capture the semantic information necessary for classification. The simplicity and interpretability of the KNN classifier make it suitable for this task, allowing for easy understanding and analysis of the classification decisions.

Future Scope and Enhancing Accuracy:

* Enhanced Feature Engineering**:** Explore advanced feature engineering techniques like word embeddings (e.g., Word2Vec, GloVe) for richer semantic information.
* Domain-Specific Models: Customize models for specific industries to capture domain-specific terminology and nuances.
* Active Learning and Feedback: Incorporate active learning and feedback mechanisms for personalized user experiences.
* Integration with ATS: Integrate with Applicant Tracking Systems (ATS) for streamlined recruitment workflows.
* Multimodal Analysis: Expand analysis to include images or audio for a comprehensive understanding of candidate profiles.
* Model Fine-Tuning: Optimize model hyperparameters and pipeline configurations for improved accuracy.
* Data Augmentation: Augment and balance datasets to address class imbalances and improve generalization.
* Ensemble Learning: Utilize ensemble learning techniques to mitigate biases and errors for more reliable results.
* Advanced Text Preprocessing: Implement advanced preprocessing to preserve relevant semantic information in text data.
* Incremental Learning: Incorporate incremental learning to adapt to evolving patterns and maintain high accuracy over time.

Conclusion:

In conclusion, this paper presents a comprehensive approach to automate resume classification using machine learning techniques. Our method offers a scalable and efficient solution for automating the resume categorization process, enabling HR professionals and recruiters to streamline the talent acquisition process. Future work may involve exploring advanced NLP techniques, deep learning architectures, and ensemble methods for further improving classification performance.

References:

H. Pendhari, S. Rodricks, M. Patel, S. Emmatty and A. Pereira, "Resume Screening using Machine Learning," 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI), Chennai, India, 2023, pp. 1-5, doi: 10.1109/ICDSAAI59313.2023.10452483.  
URL:  <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10452483&isnumber=10452427>

S. Bharadwaj, R. Varun, P. S. Aditya, M. Nikhil and G. C. Babu, "Resume Screening using NLP and LSTM," 2022 International Conference on Inventive Computation Technologies (ICICT), Nepal, 2022, pp. 238-241, doi: 10.1109/ICICT54344.2022.9850889.

URL: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9850889&isnumber=9850438>

T. M. Harsha, G. S. Moukthika, D. S. Sai, M. N. R. Pravallika, S. Anamalamudi and M. Enduri, "Automated Resume Screener using Natural Language Processing(NLP)," 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2022, pp. 1772-1777, doi: 10.1109/ICOEI53556.2022.9777194.

URL: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9777194&isnumber=9776641>

Singh, A. K., & Shukla, P. (2020). "Automated resume screening and evaluation using machine learning techniques". Journal of Intelligent & Fuzzy Systems, 39(4), 5947-5960.

GitHub repository link:

https://github.com/harshith100/Resume-screening