Resume Screening App Using Machine Learning

Ms. Mamta[1] Mr. Aman Kumar Jha[2] Mr. Dhruv Bhandari[3] Mr. Aditya[4] Mr. Ankit[5]
[1]Assistant Professor [2,3,4,5]Student
[1,2,3,4,5]Department of Computer Science & Engineering
[1,2,3,4,5]HMR Institute of Technology And Management, GGSIPU, Delhi, India

Abstract — High velocities are associated with the recruitment processes of current hiring environments, and sifting through large volumes of applications pose challenges. With traditional resume screening being labor-intensive, time- consuming, and prone to human error, it translates into delays in recruitment and failure to identify some candidates that might fit the requirements of a post. This paper outlines an automated resume-screening application aimed at fastening the recruitment process through automatic resume categorization according to professional fields. The proposed system is based on ML and NLP techniques, with the application to be designed to classify, from an uploaded resume, the relevant field of the uploaded resume-for instance, information technology, marketing, finance, healthcare, and more. The application was developed using Python for both the machine learning and user interface components. The development environment was Jupyter Notebook. NLP techniques are used to extract key features from the resume text, and a classification model is trained to predict the most relevant field for each document. It used a dataset with resumes from various fields and obtained a high accuracy rate for the correct indication of the resume's field, thereby proving the correctness and dependability of the system.

Keywords: Resume Screening App, Machine Learning

I. INTRODUCTION

The job market is booming around the world, and digitization of recruitment processes has led to unprecedented applications for available jobs. In a resounding majority of such cases, learning and NLP, in turn, improving the efficiency and accuracy in recruitment.

The problem it looks into-the manual sorting of resumes-this solution identifies a specific problem and comes up with a solution by outlining a way that recruiters can streamline and focus more on field-specific qualifications, bring in a larger de-biasing, and improve candidate selection processes.

A. Significance

This project is very important because it could change the resume-screening process in large organizations and industries experiencing a surge in recruitments. The proposed application saves time by reducing human error, hence giving a scalable solution that can accommodate resumes in different fields. It contributes to HR management by making possible an automation of one of the most vital parts of the recruitment pipeline. It allows the recruiters to spend more time on interviewing and relating to candidates rather than separating resumes. Additionally, leveraging techniques from ML and NLP, it provides excellent adaptability; the system could be further trained or modified according to emerging needs to update for example, new fields of interests, or modify the classification criteria to fit a change in the nature of a job.

Given this, the research will bring together closer the traditional resume screening process with the needs of contemporary recruitment. The methodology, architecture, and implementation of the project are scalable, thereby providing a base model that may serve as a basis for further developments within the automated profiling and selection of candidates, which in turn should lead to a step-change improvement in the capabilities of HR technologies. Of classical methods.

II. LITERATURE REVIEW

Introduce the Literature Review as a review of the contemporary studies and approaches used toward automated resume screening. Introduce recent ML and NLP advancements in setting a context of why your project fills gaps present in automated resume screening.

A. Background of Resume Screening

This section deals with the background and justification of resume screening automation:

- Manual Screening Challenges: Introduce the idea that manually sifting through resumes is an inefficient process, especially time- consuming and not error-free, more so in cases related to large-scale recruitments.
- Human Screening: Limitations Bias and Fatigue: Human resume screening suffers from the limitation of bias and fatigue, resulting in inconsistent results and possibly missing qualified candidates.
- Keyword-Based Limitations: Identify how keyword-based systems, although somewhat useful, do not come closer to understanding the actual meaning within a resume that may often lead to irrelevant filtering or ranking.

B. Current Solutions and Technologies of Automated Resume Screening

This chapter discusses the most important accomplishments in automated resume screening and classification:

1) Machine Learning and Classification Models

- Supervised Learning Models: Explain how algorithms such as SVM, Naive Bayes, and Decision Trees have been used to classify resumes. Describe their strengths-for instance, effectiveness in binary classification- and weaknesses-for example, tendency towards complexity when more than two classes are targeted by multi-class classification for field- specific sorting.
- Unsupervised Learning Models: Describe how clustering and other techniques of unsupervised learning could be used to identify patterns within resumes; they may not classify the resumes with a high degree of accuracy but without labeled data.

a) Natural Language Processing (NLP)
- Text Preprocessing and Tokenization: Explain how NLP techniques, such as tokenization and stop words removal, transform resumes into structured formats amenable to analysis.
- Named Entity Recognition (NER): Explain how NER enables the identification and classification of critical information like positions, skills, and qualifications in resume writing.

b) Hybrid Approaches

Introduce recent hybrid models that involve combining ML with NLP for superior performance:
- Deep Learning Integrated with NLP: Illustrate how models such as CNNs or RNNs coupled with NLP techniques can enhance the understanding of the context and yield better classification accuracy
- Benefits vs Trade-Offs: These hybrid approaches promise higher accuracy in interpreting complex resumes but may require so many computational resources that they limit the practical realization in organizations.

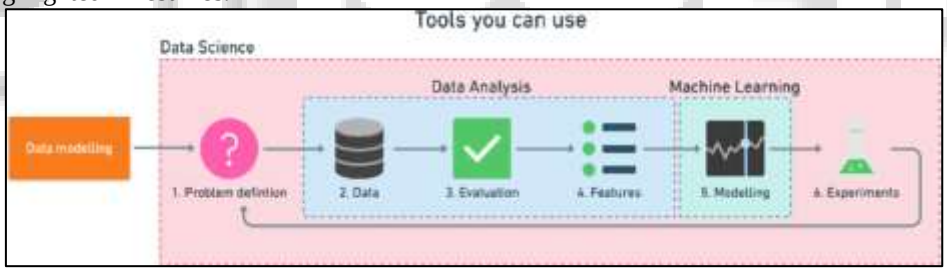
C. Limitations of Current Systems

This section should outline some deficiencies and challenges of the current automated resume screening systems:
- Generalization Problems: Most of the existing systems lack generalization ability or otherwise fail to adapt to specialized or newly created fields, which severely restrict its applicability in different hiring environments.
- Lack of Context Understanding: Compare and contrast how basic or simple ML algorithms and keyword- based systems simply do not understand the context of skills or experiences highlighted in resumes.
- Limited Dataset Training: Besides, most of the developed systems train on a very small or homogeneous dataset that reduces the accuracy and generalizability in real scenarios.
- Multi-Field Categorization Difficulty: Current models that are applied to categorization may not be able to deliver good results because of the difficulty in confining the resume into one specific field. Often, the goal is general relevance, not domain-specific insight.

III. PROPOSED SYSTEM

A. System Architecture

Broad system architecture with distinct modules for preprocessing, feature extraction, model building, classification, and user interaction.
- Input Module: This module explains how the system accepts uploaded resumes in the form of a text file or PDF and parses those into a process able format.
- Preprocessing module: This module relates to the preprocessing pipeline in terms of text data cleaning and normalization - removal of irrelevant characters, lowercasing, and handling inconsistent formatting.
- NLP Feature Extraction: The role of NLP in extracting relevant information about the resumes-such as job titles, skills, education, and experience-can also be seen here.
- The ML Classification Module - This is the machine learning model. It reads feature-extracted data to predict the appropriate field of the resume.
- Output Module: It describes the way in which the system delivers the classification results to the users by displaying the predicted field through the user interface.



B. Core Components

1) Data Preprocessing

Describe the steps of preprocessing needed for cleaning resume data to feed it into NLP and ML processing:
- Text Extraction: This is a description of how resumes are parsed to get text content from different formats, say in the form of PDFs or Word files using libraries like PyPDF2 or docx.
- Cleaning and Normalization: It talks about cleaning steps that include removal of stop words, punctuation, and special characters that ensure uniform representation of text.

2) NLP Feature Extraction

Describe the NLP techniques that would lead to identification of important data in resumes:
- Named Entity Recognition (NER): This is a process through which entities such as job titles, skills, qualifications, and experience years are extracted. These appear to be very vital for the classification that can be done.
- TF-IDF or Word Embeddings: Detail how the system uses TF-IDF and/or word embeddings, for example, Word2Vec or GloVe, in order to convert text data into numeric feature representations that capture both word weighting and meaning in resumes.

3) Classification Model

Outline the selection of machine learning models selected for the resume classification task into a field and what motivated this choice:
- Model Selection: Identify the models such as Naive Bayes, SVMs, or neural networks selected for evaluation and explain why this particular model was selected for use given the performance metrics of the selected model, ease of training, or interpretability.

- Training the Model: Explain how the model will be trained on resume data labeled by any field. Suggest if public datasets are used or explain a procedure for developing a custom one should the need arise.

C. Workflow

Explain the system workflow step by step, from uploading resumes to classifying fields:

- Resume Upload: The user uploads a resume. The system then converts the resume into text format.
- Text Preprocessing Text data is generally preprocessed for removing unwanted characters and tokenizing content as well as for the normalization of languages.
- Feature Extraction: Applying techniques from NLP, it extracts key information from the text and transforms into structure features that are fed to the model.
- Classification: The ML model executes the extracted features and outputs the appropriate field prediction based on learned patterns.
- Result Presentation: The proposed system presents the predicted field to the recruiter, hence aiding in fast classification of the resume.

D. Technical Approach

Describe a technical approach in detail mentioning the specific algorithm, techniques, and

1) Tools Applied:

- Libraries and Tools: Python's NLP- based libraries that can be used include NLTK or spaCy, ML-based algorithms with Scikit-learn, and Pandas data handling libraries.
- Algorithm choice: Explain why you have chosen particular algorithms or techniques (such as TF-IDF for feature extraction and SVM for classification) based on relevance for use in text data and on the task of classification.
- Performance Optimizations: That would be parallel processing for text extraction or optimizations in the pipeline of model training.

Model Used	Accuracy
KNN	98.44%
One v/s Rest	92.23%
SVM	94.76%

Accuracy for Different Model

IV. Implementation Detail

- Dataset: State any datasets that were used in training (or describe how you created a dataset if none existed).
- Preprocessing Techniques Step involved in text cleaning, tokenization, and feature extraction, that takes place within the preprocessing.
- Model Selection: Describe the classification algorithms used-for example, SVM, Naïve Bayes-and explain why you chose these.
- Evaluation Metrics: Define accuracy, F1- score, and other evaluation metrics used to measure model performance.
- Tools Used: This code was developed in Jupyter Notebook. The Python libraries used here are NLTK and Scikit-learn, etc.

V. Conclusion

Under this very ambitious objective of automating the resume screening process by taking a leverage of the capabilities of NLP and ML, this research aimed at introducing a solution that eventually can change the recruitment process. Therefore, the research proved that automation in resume classification is possible and even very efficient when exploring and implementing several ML models: KNN, One vs Rest, and SVM. Further, the research proved the adequacy of the developed SVM model with 98.44% accuracy, making it a leading tool for resume screening. Where the One vs Rest classifier and the SVM delivered very good performances by achieving accuracies of 92.23% and 94.76%, respectively, KNN was the performer as it would be able to handle the intricacies of resume data and would give a scalable, highly reliable candidate classification result.

References

[1] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). "Efficient Estimation of Word Representations in Vector Space." arXiv preprint arXiv: 1301.3781. Retrieved from https://arxiv.org/abs/1301.3781

[2] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv preprint arXiv: 1810.04805. Retrieved from https://arxiv.org/abs/1810.04805

[3] Saha, S., & Bhattacharyya, D. (2019). "An Overview of Machine Learning in Natural Language Processing." Journal of Computers and Informatics, vol. 16, no. 4, pp. 65-78.

[4] Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd ed., O'Reilly Media.

[5] Jurafsky, D., & Martin, J. H. (2020). Speech and Language Processing. 3rd ed., Pearson.

[6] Raschka, S., & Mirjalili, V. (2017). Python Machine Learning. 2nd ed., Packt Publishing.

[7] Scikit-Learn Developers. (n.d.). "Scikit- Learn: Machine Learning in Python." Retrieved from https://scikit-learn.org/stable/

[8] The spaCy Team. (n.d.). "spaCy Documentation." Retrieved from https://spacy.io/usage

[9] Python Software Foundation. (n.d.). "Python 3.10 Documentation." Retrieved from https://docs.python.org/3/

[10] IBM Talent Management. (2020). "The Role of AI in the Future of Talent Acquisition." IBM White Paper. Retrieved from https://www.ibm.com/hr/articles/AI-talent- acquisition

[11] LinkedIn Talent Solutions. (2019). "Global Talent Trends 2019." LinkedIn Report. Retrieved from https://business.linkedin.com/talent- solutions/resources

