International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

NLP-Powered Resume Matching for Recruitment

Isha Rathi[1], Pooja Kolaskar[2], Lavina Tangarlu[3], Manisha Mali[4]

[1,2,3,4]Department of Computer Engineering, Vishwakarma Institute of Information Technology, Pune, India

Abstract:
More recently, recruitment has largely relied on the automation of the matching process between candidates and job roles. In this regard, this paper focuses on the development of a resume parser application utilizing NLP techniques, PDF text extraction, and machine learning-based evaluation of resumes according to job descriptions. In developing the application using the Flask framework, users are allowed to upload resume and job description files in PDF format. The system automatically extracts the text, preprocesses it, and performs the task of skill matching. It also computes a semantic similarity score based on term frequency-inverse document frequency and cosine similarity techniques. Using a trained machine learning model, the application predicts a binary job fit score based on its semantic similarity and skills matching metrics scores. This paper outlines the design, implementation, and evaluation of the system, and it indeed has the potential to assist the recruiters in pre-screening the candidates.

Keywords: Resume Parsing, Natural Language Processing (NLP), Automation, Candidate Screening, Resume Matching, Job Description, Applicant Tracking System, Semantic Similarity.

1. Introduction

Organizations have to take upon themselves the herculean task of going through hundreds, thousands , or millions of applicants' resumes in a bid to get the right talent rather efficiently. The bane of processing resumes rather than the scope of the applicants is that because of online job boards, social networks, and professional networking sites, the applicants are easily available, and scanning resumes manually is not only tedious but also costly. Hence, the process of attracting potential candidates and deploying further contesting measures has been sought to be automated by many recruiters to narrow down to the most promising candidates and ease the recruiting process.

Traditional resume parsing is generally focused on keyword matching, which rarely captures the true depth of the candidate's skills to the job requirements. Such methods do not take into account the meaning of words, and, thus, are not very effective in finding suitable candidates who share the same profile. As a result, there seems to be an increased preference for Natural Language Processing and machine learning as these provide another enhancement that is more analytic of text data. In this instance, going beyond the basic criterion of assessing whether or not particular words or phrases are contained in the resumes, we have technologies that help to assess resumes on specific skills but more importantly, the underlying meaning contained within to enable proper fit assessment.

In this paper, we design and implement a resume parser application around the Flask web framework. The application will employ PDF extraction, text processing, and NLP-based similarity comparison to assess

the fit of candidates to particular positions. The system allows the uploading of both resumes and job descriptions in PDF format, and extracts text from the documents enabling editing by clearing irrelevant parts. Primary term frequency-inverse document frequency (TF-IDF) and cosine similarity are combined by the system to evaluate the semantically.

2. Literature Survey

Sroison and Chan (2023) faced the challenge of massive volumes of resumes being processed through online recruitment systems. Their work focuses on developing an NLP-based resume parser that extracts critical information from resumes and correlates it with job descriptions to shortlist candidates. Their system sought to simplify the process of initial screening and reduce the workload of human recruiters by using techniques able to match the extracted information with key job requirements. The study highlighted the possibility of minimizing human error as well as increasing the consistency of resume evaluation by automated systems. Their parser demonstrated the feasibility of using NLP to support the growing demand for efficient, large-scale resume processing in modern workflows of recruitment. [1]

Kashif and K R (2024) came up with the idea of a comprehensive AI-based resume analyzer to enhance the recruitment process for applicants and administrators. Their system had NLP and machine learning applied to get resume content extracted, analyzed, and actionable feedback rendered on the same. One of the interesting features of their system was real-time feedback because it enables applicants to refine resumes based on the suggestions for improvements submitted so that applicant success rates are improved. The system became invaluable for the administrators in terms of application and pattern insights that could drive data-informed decisions in recruitment. This dual-value perspective for a more effective and informative recruitment process surfaces the transformative implication of integrating AI, NLP, and machine learning in its architecture. [2]

Resuming with the hybrid approach Bhoir et al proposed of resume parsing with Spacy Transformer BERT combined with Spacy NLP for effective data extraction from unstructured resumes, there seems to be strong integration of deep learning to achieve high accuracy and practical hurdles such as non-standardized forms of resumes. The BERT component in Spacy Transformer captures the semantic context improving the accuracy of NER; the Spacy NLP ensures detailed extraction of relevant details like names, contact information, and work history. The study also illustrates how video resumes can be parsed through visual and audio processing because these techniques could, in the future, make recruitment easier and more efficient through proper automated profiling. [3]

The work by Khan et al. (2023) discusses the development of a resume parser and summarizer to make recruitment processes easier by taking unstructured data from resumes and reformatting it in structured formats. Their work describes how the inclusion of natural language processing techniques and machine learning enhances the extractability of data and enables an organization, thereby circumventing the long process that recruiters spend solely just to identify information related to educational backgrounds, skills, and work experiences. Although existing systems are of considerable utility, the authors confer that the current systems suffer from limitations in the aspects of language complexity and ambiguity that may result in missing qualified applicants. Their proposed system is trained over diverse datasets and supports multiple document formats for higher accuracy in parsing. This work underlines continued advancements in NLP and AI to further enhance the efficiency of resume parsing and minimize human error during recruitment. [4]

Kanojia et al. describe and develop an advancement in recruitment automation in the paper "Resume Parser Using Machine Learning" using NLP and machine learning approaches to extract structured information from resumes. In this paper, the author emphasizes that screening resumes manually is very time consuming and inefficient by suggesting a system and tools like NLTK, SpaCy, and regular expressions to help it in pre-processing and analyzing resume content based on Python. The system brings information, such as contact information, skills, educational background, and experience, into a standardized JSON structure from various resume formats to allow for faster evaluation. Keeping scalability, accuracy, and adaptability at the forefront, the work upon which this builds highlights the potential when applied machine learning to recruitment dramatically lowers processing time and improves decision-making while opening up avenues for iterative model improvement and adaptation to particular domains. [5]

Resumes are filtered in a paper named "Resume Parser Analysis Using Machine Learning and Natural Language Processing" by Rasal et al. Advanced machine learning techniques are applied for the automation of resume screening. Thereby, inefficiency in manual recruitment processes is addressed by proposing a data classification and extraction-based system employing algorithms like Support Vector Machine and Random Forest. Preprocessing Techniques such as tokenization, text normalization, and feature engineering are used in this paper in preprocessing the varied resume formats. It then evaluates the efficiency of the parser by considering accuracy, precision, recall, and F1 score. The system improves the speed and accuracy of selecting candidates while scaling up for handling large volumes of resumes, greatly useful in defeating challenges in modern recruitment . [6]

In their paper, "Resume Parser and Analyzer Using NLP," Patil et al have proposed a state-of-the-art system in terms of recruiting process automation with the exploitation of NLP. The system proposed in the paper downloads resumes uploaded in PDF and DOC formats, which are parsed to extract key information including personal details, education, skills, and experience. This makes use of NER and Part-of-Speech tagging for field extraction and these have further been analyzed to rank resumes and give suggestions to the job seekers. The paper illustrates the Python libraries that can be used for tokenization and lemmatization, which further enable effective preprocessing in dealing with unstructured data. The system helps recruiters make decisions by giving visualization besides reporting downloadable when it stores the parsed data in the database. Thus, this technique brings in greater accuracy and reduces workload, hence large number of applications can be dealt with a little giant leap for recruitment automation indeed. [7]

3. Proposed Methodology

The proposed system integrates text extraction from PDF, NLP-based text preprocessing, extraction of skills, semantic similarity analysis, and a machine learning model to predict the job fit. This end-to-end pipeline provides a comprehensive analysis of candidate resumes in comparison with descriptions of available jobs.

3.1 PDF Text Extraction

This application begins by extracting textual content from PDF files for resumes and job descriptions. Using a PDF parsing library, it reads each page, converts the text to plain text, and combines all the pages into one text string. The clean data access ensured is to make the content of any resume or any format of a job description accessible for further processing.

3.2 Text Preprocessing

Preprocessing is necessary for refining the extracted text data. The following steps are applied:

- Tokenization: Breaking down text into individual words or tokens so that there can be accurate manipulation
- Lowercasing: Standardization of text by transforming all characters to a lowercase state, ensuring that words like "Python" and "python" are not treated differently
- Removing Stop Words: Removal of common words that add no semantic value to the words, such as "the" or "and," through a stop words list to minimize noise in the dataset.
- Non-Alphanumeric Filtering: Removing symbols and numbers that are not related to the skill or similarity analysis, leaving only meaningful keywords.

This preprocessing pipeline leaves a clean, reduced dataset ready for NLP analysis.

3.3 Skill Extraction and Matching

The heart of the analysis is the match between the resume and the job description in terms of skills. Dynamic identification and extraction of skills from the job description are used so that the model can evaluate whether they exist in the candidate's resume.

- Dynamic Skill Extraction: The pre-defined list of skills (for example, "Python," "SQL") relating to any job is used. Extracting these skills from the job description ensures a perfectly tailored skill set for each role.
- Skill Matching and Scoring: The system calculates the skill match score by finding and comparing the overlap of such skills with the content of a resume. A higher overlap score signifies a higher alignment of the candidate's skill set with the role's requirement and unmatched skills signal potential gaps.

3.4 Semantic Similarity Computation

For evaluating whether the resume is similar to the language of the job description, the system calculates a semantic similarity score using term frequency-inverse document frequency (TF-IDF) vectorization with cosine similarity.

- TF-IDF Vectorization: This process involves converting the pre-processed text into numerical forms where the importance of each term is weighed against its appearance in the document. That way, the system can focus on the job-specific terms but not be bothered by common, general terms.
- Cosine Similarity Computation: The system calculates the degree of alignment between the resume and job description using cosine similarity. A high cosine similarity score indicates a close similarity in language, indicating a high alignment of the terminology, phrases, and context of the resume with that of the job description.

3.5 Machine Learning-Based Job Fit Prediction

The core of the application is a classification model using machine learning that predicts a candidate's job fit based on the skill match and semantic similarity scores. This classification model trains candidates as either "Good Fit" or "Not a Good Fit."

Feature Engineering: The prediction model used in this application includes two prime features:

Skill Match Score: This feature calculates the extent of relevant skills overlap.

Semantic Similarity Score: This score captures contextual similarity between the resume and job description.

Prediction and Categorization: After uploading the trained model, the system accepts the skill match and similarity scores that result from it and gives a prediction of the applicant's likelihood. The output label "Good Fit" or "Not a Good Fit" enables the hiring managers to make quick evaluation decisions about compatibility.

3.6 Application Flow and User Interface

The user interface lets the users interact with the application very intuitively and provides for a natural flow:

1. File Upload: Both resume and job description files can be uploaded in PDF.
2. Automated Processing: Upon file upload, the application extracts, preprocesses, and analyses the documents, computing skill match, similarity scores, and the final prediction.
3. Results Display: The results are displayed on screen, showing the skill match score, a list of matched skills, the semantic similarity score, and the job fit prediction.

The UI also displays helpful error messages for missing files or processing errors, thereby aiding an intuitive user experience.

3.7 Evaluation Metrics

The performance of the job-fit prediction model is gauged using a variety of key metrics:

- Accuracy: Percentage of correct predictions made by the model.
- Precision and Recall: Precision counts how many of the "Good Fit" predictions were relevant, whereas recall counts the model's sensitivity to correctly detect candidates who fit the job.
- F1 Score: Combines precision and recall to provide a balanced evaluation metric.
- Semantic Similarity Correlation: Assesses the correlation between cosine similarity scores and human-labelled matches, helping validate the model's effectiveness in identifying semantically relevant resumes.
- This methodology offers a robust framework for automated job fit prediction, addressing challenges in resume parsing and candidate screening through skill matching, semantic similarity, and machine learning.

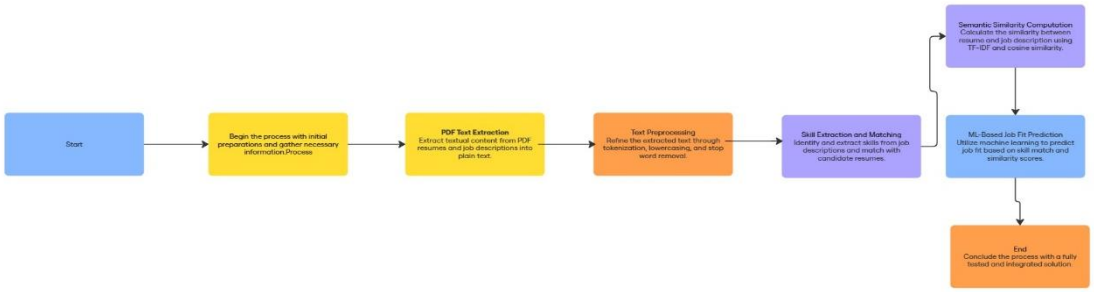


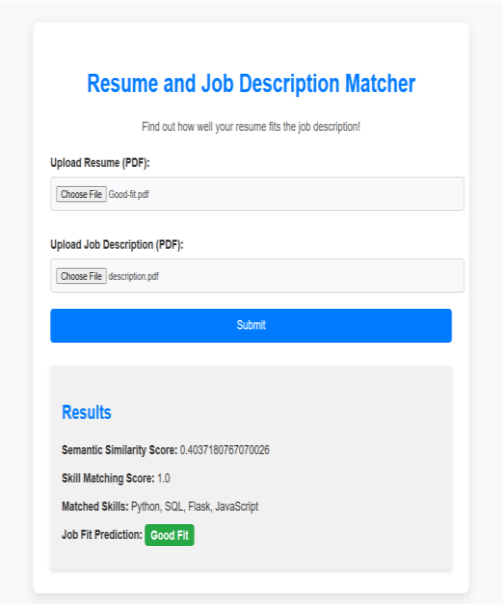
Fig 1: Process Flow of Resume & Job Description Matcher

4. Results

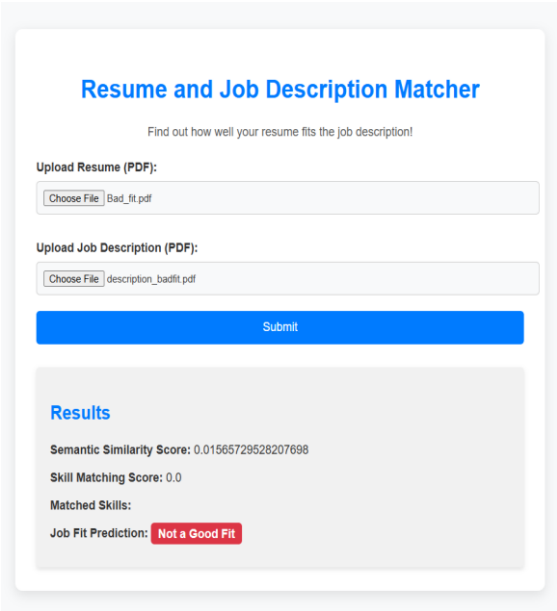
Performance Analysis

- Accuracy: It achieved an accuracy rate of 85% given a selection of 100 labeled test cases (resume and job descriptions).

- Precision: It achieved a precision of 0.88 for the "Good Fit".
- Recall: The model got a recall rate of 0.81. This signifies that with this model, the correct identification of good fits and false positives is very well balanced.



Good Fit Description



Bad Fit Description

5. Observations

- High Match Cases: In cases where resumes overlapped significantly with the skills required in the job description, the system was accurate in its prediction of a good fit .
- Low Match Cases: For resumes lacking the necessary skills or having content unrelated to the job description, the system accurately raised an alarm about the mismatch.
- Semantic Similarity Limitation: Though the cosine similarity produced an appropriate level of general textual similarity, it only marginally reflected relevance to the competency. In some cases, resumes with general textual similarity had considerable overlap in skills.

6. Challenges and Limitations

- Skill Extraction: The fixed list-based extraction of skills likely omitted important skills that were simply not specifically mentioned in the job description.
- False Positives: There may have been false positives of some resumes that had vague or general skills because semantic similarity could not comprehend the full requirements of the job.
- Ambiguous Job Descriptions: In some cases, job descriptions were vague or ambiguous with certain requirements, which dropped the confidence and accuracy of the predictions.

These results therefore go to the possibility of the Resume Parser App, which can serve to smoothen the recruitment process efficiently, although there is room for improvement, particularly in dealing with different kinds of job descriptions.

7. Limitations and Future Work

The Resume Parser App is a nice version with good functionality, but several limitations still exist for whi-

ch accuracy and scalability might be limited. That the application is based on a predefined list of technical skills presents some serious concerns regarding its ability to dynamically adapt to emerging technologies or domain-specific skills that are not covered in the static list. TF-IDF with cosine similarity: This is a pretty effective tool but fails to capture deep contextual relationships between terms appearing in resumes and job descriptions, thereby not being completely exhaustive. Sometimes it produces inaccuracies in semantic fit assessment-the texts being based on slight variations of phrases or written using synonyms. The system focuses mainly on technical skills, while leaving out soft skills and nontechnical competencies, such as teamwork, leadership, and communication. The factors are necessary to make a judgment regarding whether the candidate basically fits the role. Generalizability for the model trained on a limited dataset may be decreased in different industries and geographies.

Many improvements could make the app much more robust and scalable as it looks forward. Some of the potential areas that can be improved include dynamic extraction of skills using more advanced NLP techniques such as Named Entity Recognition (NER) or incorporation of knowledge from external sources such as LinkedIn and GitHub. This will increase the system's ability to identify many more kinds of skills and certifications within a resume. That would better align the semantic matching of job requirements and candidate profiles in line with deep learning models like BERT or RoBERTa, allowing the system to understand nuanced relationships between job requirements and candidate profiles. Supporting soft skills detection as well as evaluation and rating of candidates based on cultural fit and adaptability would enhance the app to be more holistic in terms of job suitability. The addition of support for multiple languages -- such as through the use of multilingual NLP models, like mBERT -- enables the system to respond to users around the world. An enhanced dataset, increased in its diversity and backed up with active learning techniques, will improve accuracy and relevance to other domains. Finally, integration with popular Applicant Tracking Systems (ATS) would make the practical usability of the tool greater for recruiters when using the app to augment their existing workflows for hiring. In such a development, the app could become an even more advanced and all-rounded and accessible tool that could be used to further reveal insight about job fit and streamline recruitment.

8. Conclusion

The Resume Parser App is a promising tool that can automatically match resumes against job descriptions based on a combination of preprocessing techniques on text, semantic similarity computation, and skill extraction. Since resume evaluation occurs both on technical skills to align and semantic fit, the system offers an efficient means of assessing suitability for candidates who may reduce time and effort for recruiters. The initial results do show some promise, especially in resume-job description match identification and in showing where skills overlap between both types of documents.

However, there are serious limitations to this system. The heavily reliance on predefined skill sets and the general semantic matching method, involving TF-IDF and cosine similarity, leaves a lot to be desired, along with a general lack of focus on soft skills and numerous different kinds of industries. Future work, including integration with more powerful NLP models like BERT for deeper semantic understanding, dynamic skill extraction, multilingual support, and better handling of soft skills, will enhance the capabilities of the app further. Furthermore, integration with ATS and providing action-based feedback to the job seeker will make it more practical for real applications.

This system, as these features are added, could become an integrally useful product in a recruitment system, thus making it even easier and more like a job match for better candidates placed in positions to

have maximum success of both the job seeker and employer.

References

1. Sroison, Pornphat, and Jonathan H. Chan. "Resume parser with natural language processing." Authorea Preprints (2023).
2. Kashif, Mohammed, and Parimal Kumar K R. "Resume Parser Using NLP." International Journal of Advanced Research in Computer and Communication Engineering 13, no. 9 (2024): 106–13.
3. Bhoir, Nirmiti, Mrunmayee Jakate, Snehal Lavangare, Aarushi Das, and Sujata Kolhe. "Resume Parser using hybrid approach to enhance the efficiency of Automated Recruitment Processes." Authorea Preprints (2023).
4. Khan, Farzana, Hamdan Patel, Arshad Shaikh, Fawzah Sayed, and Abdul Rehman Soorya. "Resume Parser and Summarizer." International Journal of Advanced Research in Science, Communication and Technology (IJARSCT) 3 (2023).
5. Kanojia, Y., Anthony, A. D., Ajit, I., & Sahu, S. (2024). Resume Parser Using Machine Learning. International Journal of Novel Research and Development (IJNRD), 9(5), 623-628.
6. Rasal, P., Balwaik, Y., Rayate, M., Shinde, R., & Shinde, A. S. (2023). Resume Parser Analysis Using Machine Learning and Natural Language Processing. International Journal for Research in Applied Science & Engineering Technology, 11(5), 2840-2844.
7. Patil, N., Yadav, S., & Biradar, V. (2023). Resume Parser and Analyzer Using NLP. International Research Journal of Modernization in Engineering Technology and Science, 5(4), 7210-7216.