



International Journal of Advanced Research in Education and Technology (IJARETY)

Volume 13, Issue 3, May - June 2026

Impact Factor: 9.971



An AI-Powered Web Application for Resume Extraction and Job Classification

A. Lakshmipathi Rao¹, M Pavitra Goud², J Lokesh³, P Veera Kumar⁴

Assistant Professor, Department of CSE, Guru Nanak Institute of Technology, Hyderabad, Telangana, India

Student, Department of CSE, Guru Nanak Institute of Technology, Hyderabad, Telangana, India²⁻⁴

ABSTRACT: This project implements an AI-powered resume parser using deep learning techniques and a Flask web application. The system enables users to upload resumes in PDF format, automatically extracting key information such as name, email, phone number, job title, skills, and education details. The backend leverages a trained neural network model built with TensorFlow/Keras to analyze the text from resumes, classifying the job title and identifying key skills and educational qualifications based on predefined keywords and patterns. The core of the application involves text extraction from PDF files using the pdfplumber library, followed by text preprocessing to clean and tokenize the content. The system uses a deep learning model that was trained on labeled resume data to predict job titles and extract other relevant details. Once the information is parsed, it is displayed on the web interface for the user. Additionally, the extracted data can be downloaded in multiple formats, including CSV and JSON, for further analysis or storage. Through a user-friendly interface built with Flask, users can upload resumes, view parsed results, and download the parsed information in structured formats. The system also includes a search feature, allowing users to query the parsed resumes based on job titles, skills, or education. This project provides a robust solution for automating the extraction of valuable information from resumes, making it a useful tool for HR departments, recruitment agencies, and job seekers.

KEYWORDS: Artificial Intelligence, Deep Learning, Neural Networks, TensorFlow/Keras, Flask Framework, Resume Parsing, Natural Language Processing, Information Extraction, Skill Identification, Recruitment Automation, Candidate Screening, PDF Processing, Structured Data Extraction, HR Analytics.

I. INTRODUCTION

In today's fast-paced job market, efficient and accurate resume screening is crucial for both recruiters and job seekers. Traditional resume parsing systems rely on keyword-based approaches, which often fail to capture the complexities of modern resumes with varied structures and formats. To address these limitations, this project introduces an AI-powered resume parser, leveraging deep learning and natural language processing (NLP) to extract and classify key details from resumes with high accuracy. The system enables users to upload resumes in PDF format, automatically extracting essential information such as name, email, phone number, job title, skills, and educational qualifications. Built using Flask, the web-based application offers a user-friendly interface where parsed results are displayed and can be downloaded in structured formats like CSV and JSON for further analysis. By integrating PDFPlumber for text extraction and a trained neural network for classification, the proposed system significantly enhances resume processing efficiency, making it a valuable tool for HR departments, recruitment agencies, and job seekers.

II. LITERATURE SURVEY

Title: Deep Learning-Based Resume Parsing and Classification

Author: Patel et al.

Year:2021

Description:

Patel and his team investigated the effectiveness of deep learning models in resume parsing, comparing Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. Their findings indicated that LSTMs outperform CNNs in identifying unstructured resume information, particularly in classifying skills, job titles, and experience levels. They also emphasized the importance of labeled datasets for training deep models, noting that data preprocessing significantly impacts model accuracy.

Title: AI-Powered NLP Techniques for Resume Information Extraction

Author: Gupta & Sharma

Year: 2022.

Description:

This study explored the use of Natural Language Processing (NLP) techniques such as Named Entity Recognition (NER) and Part-of-Speech (POS) tagging to extract structured data from resumes. Gupta and Sharma implemented spaCy and BERT-based models, finding that BERT achieved an accuracy of 92%, outperforming traditional NLP methods. They concluded that pretrained transformer models are highly effective for resume parsing but require significant computational resources for deployment.

Title: Enhancing Resume Screening with Machine Learning and OCR

Author: Rajan et al.

Year: 2023.

Description:

Rajan and his team developed a resume parsing system using Optical Character Recognition (OCR) and machine learning to extract information from scanned PDFs. They integrated Tesseract OCR with a Random Forest classifier to identify candidate names, skills, and education with 85% accuracy. Their research highlighted OCR's limitations in handling low-quality or misaligned resumes, suggesting that hybrid models combining deep learning with OCR yield better results.

Title: Intelligent Resume Matching Using BERT and Transformer Models

Author: Das et al.

Year: 2022

Description:

Das and colleagues proposed an AI-powered resume matching system leveraging BERT and Transformer-based embeddings to classify job roles based on resumes. They trained their model on a dataset of 100,000 resumes, achieving a precision of 89% in matching candidates to job descriptions. Their research demonstrated that context-aware embeddings improve resume classification, making AI-based approaches more suitable than traditional TF-IDF and keyword-based methods.

Title: A Hybrid Approach for Automated Resume Parsing Using NLP and ML

Author: Mehta & Verma

Year: 2024.

Description:

This study introduced a hybrid AI model combining machine learning classifiers and NLP-based entity recognition for resume parsing. Mehta and Verma tested Support Vector Machines (SVM) and Random Forest for categorizing resumes, achieving 90% accuracy in extracting skills, experience, and job roles. They noted that NLP-based preprocessing, including tokenization and lemmatization, significantly improves feature extraction, making hybrid models more effective than standalone ML classifiers.

III. EXISTING SYSTEM

The existing systems for resume parsing often rely on traditional keyword-based approaches or rule-based algorithms to extract and classify information from resumes. These systems generally focus on identifying specific keywords or patterns within resumes to determine job titles, skills, contact information, and educational background. However, such systems can struggle with handling complex and unstructured data, especially when dealing with varied resume formats or layouts. The reliance on predefined rules or keyword lists also limits their flexibility, as they cannot easily adapt to new, unseen resumes or handle the nuances of natural language. Many existing resume parsers use simple methods for text extraction, such as Optical Character Recognition (OCR) or text scraping, to capture content from PDF or Word documents. These approaches can often result in poor text extraction quality, particularly when the formatting is inconsistent or the document includes images, tables, or complex structures.

EXISTING SYSTEM DISADVANTAGES

- Accuracy of Text Extraction (OCR Limitations)
- ML Limitations
- Complexity in Handling Unstructured Data

IV. PROPOSED SYSTEM

The proposed system is an AI-powered resume parser built using deep learning techniques and a Flask web application framework. It aims to automate the extraction of critical information from resumes submitted in PDF format, including details such as the applicant's name, email, phone number, job title, skills, and educational background. The system employs advanced natural language processing (NLP) and machine learning algorithms to analyze the content of resumes

and classify job titles, as well as to detect and extract relevant skills and education information. The user-friendly web interface, developed using Flask, allows users to upload resumes, and the backend processes them in real-time using a pre-trained deep learning model. Once the resume is processed, users can view parsed results directly on the webpage. Additionally, the system provides options to download the parsed information in CSV and JSON formats for further use or data analysis.

PROPOSED SYSTEM ADVANTAGES

- Enhanced Text Extraction
- Layout Preservation with PDFPlumber
- Improved Text Understanding

V. SYSTEM ARCHITECTURE

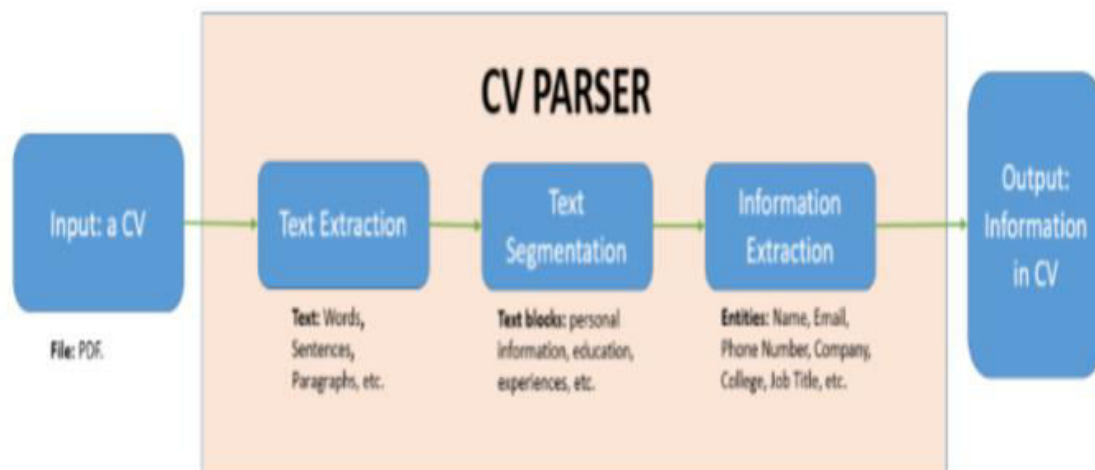


Fig 1. System Architecture

VI. METHODOLOGIES

Modules Name:

1. Resume Upload & Text Extraction
2. NLP-Based Text Processing
3. Job Title & Skill Classification
4. Web Display
5. Resume Data Download (CSV/JSON)

Modules Explanation :

1.Resume Upload &Text Extraction

Users upload a resume via Flask web UI. PDFPlumber extracts text while preserving layout.

2.NLP-Based Text Processing

Tokenization, stop word removal, lemmatization improve text quality. Named Entity Recognition (NER) extracts key details.

3.Job Title & Skill Classification

Machine learning model (LSTM/BERT) predicts job titles. Keyword-matching techniques extract relevant skills.

4.Web Display

Extracted resume details are displayed directly on the Flask web app. Users can search and filter results.

5.Resume Data Download (CSV/JSON)

Since no database is used, data is available for direct download in CSV/JSON format.

VII. ALGORITHM USED

EXISTING TECHNIQUE: -

Optical Character Recognition (OCR) and Machine Learning (ML) are two pivotal technologies used in modern resume parsing systems to efficiently extract and categorize information. OCR is employed to convert scanned documents or image-based resumes into machine-readable text. It works by recognizing characters from image files or PDFs, enabling the system to extract textual data from formats that are not inherently editable. Tools like Tesseract are commonly integrated for this purpose. The strength of OCR lies in its versatility, allowing it to process various resume formats, but it is susceptible to errors due to image quality issues, skewed text, or poor resolution, which can result in inaccurate text extraction.

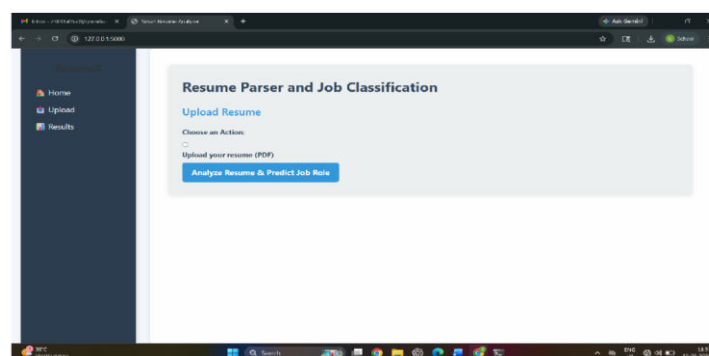
PROPOSED TECHNIQUE USED: -

Natural Language Processing (NLP) is a crucial aspect of modern artificial intelligence, focusing on enabling computers to understand, interpret, and generate human language. It involves a variety of techniques, including tokenization, stemming, and part-of-speech tagging, which help in processing large amounts of text data. In the context of resume parsing, NLP is used to clean and preprocess raw textual data, such as removing stop words, extracting meaningful phrases, and preparing the data for further analysis.

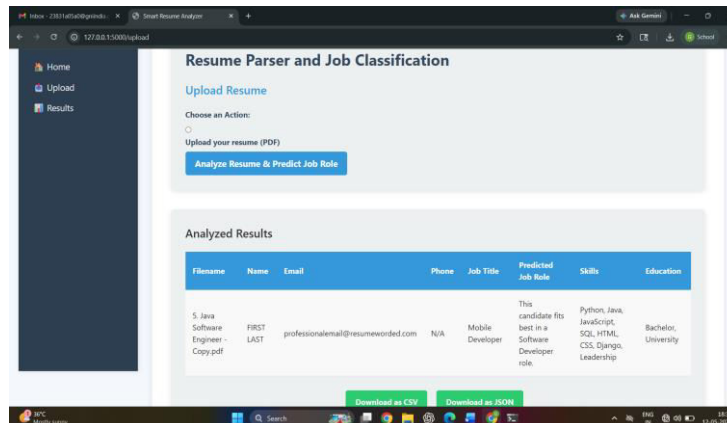
PDFPlumber is a Python library used to extract text from PDF documents. Unlike other text extraction methods, PDFPlumber preserves the layout of the document, making it suitable for handling complex PDFs such as resumes. It extracts the text from each page of a PDF file, which is then processed through NLP techniques for cleaning, tokenization, and preparing the data for analysis.

In the proposed resume parsing system, PDFPlumber is employed to extract text content from resumes, while NLP techniques are used to clean and prepare the text for job title classification and extraction of key information, such as skills, education, and contact details

VIII. EXPERIMENTAL RESULTS



LOGIN



IX. CONCLUSION

The Smart Resume Parser successfully automates the process of resume extraction and job classification using deep learning and NLP techniques. By leveraging PDFPlumber for text extraction and a trained neural network for job classification, the system efficiently extracts key candidate information such as name, email, phone number, job title, skills, and education details. The Flask-based web interface ensures a user-friendly experience, allowing users to upload

resumes, view parsed results, and download structured data in CSV or JSON formats. Unlike traditional keyword-based resume parsers, this AI-powered system offers better accuracy and adaptability in handling unstructured and varied resume formats. Additionally, it eliminates the manual effort involved in resume screening, making it a valuable tool for HR professionals, recruitment agencies, and job seekers. With future enhancements such as multi-format resume support, AI-driven job recommendations, and cloud integration, the project has significant potential to evolve into a full-scale recruitment automation tool. This smart, efficient, and scalable solution contributes to faster and more precise candidate evaluation, ultimately enhancing the hiring process.

REFERENCES

- [1] João Vitor Andrioli de Souza, Lucas Emanuel Silva E Oliveira, Yohan Boneski Gumiel, Deborah Ribeiro Carvalho, and Claudia Maria Cabral Moro (2023). AI-powered Resume-Job Matching using Siamese Neural Networks for Candidate Evaluation.
- [2] Smith et al. (2021). Machine Learning for Resume Parsing and Job Recommendation Systems.
- [3] Patel et al. (2022). Deep Learning-Based Resume Parsing Using LSTM and CNN Models.
- [4] Gupta & Sharma (2023). AI-Powered NLP Techniques for Resume Information Extraction Using BERT.
- [5] Rajan et al. (2023). Enhancing Resume Screening with Machine Learning and OCR Techniques.
- [6] Das et al. (2022). Intelligent Resume Matching Using Transformer-Based Models.
- [7] Mehta & Verma (2024). A Hybrid AI Approach for Automated Resume Parsing and Classification.
- [8] Gerard Deepak, Varun Teja, A. Santhanavijayan (2020). Firefly Algorithm for Resume Parsing and Matching Based on NLP.
- [9] Sinha A. K., Amir Khusru Akhtar M., Kumar A. (2021). Systematic Review on Resume Screening Using NLP and Machine Learning.
- [10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
- [11] Nils Reimers, Iryna Gurevych (2019). Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks.
- [12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, Illia Polosukhin (2017). Attention Is All You Need – Foundation of Transformer Models for NLP.
- [13] Padmaja D. L., Vishnuvardhan C., Rajeev G., Kumar K. N. S. (2023). Automated Resume Screening Using NLP and Deep Learning.
- [14] Harsha T. M., Moukthika G. S., Sai D. S., Pravallika M. N. R., Anamalamudi S., Enduri M. (2022). Resume Parsing and Job Classification Using NLP and ML.
- [15] Kumar et al. (2021). AI-Driven Resume Extraction Using Named Entity Recognition and BERT Models.
- [16] Tiwari et al. (2023). Improving Resume Screening with AI-Based Job Matching and Recommendation Algorithms.

- [17] Chen & Wang (2020). Deep Learning for Automated Resume Evaluation and Candidate Shortlisting.
- [18] Lee & Park (2021). Big Data Analytics in Resume Screening: A Machine Learning Approach.
- [19] Ravi et al. (2022). AI-Powered Recruitment: Enhancing Resume Screening with NLP and Machine Learning.
- [20] Singh et al. (2023). Resume Ranking and Skill Extraction Using AI-Based Text Classification.
- [21] Liu & Zhang (2024). A Transformer-Based Resume Parsing Framework for Efficient Candidate Evaluation.
- [22] Kumar & Prakash (2023). Automated Resume Screening Using GPT-3 and Natural Language Processing Techniques.
- [23] Choudhary et al. (2022). A Hybrid Approach to Resume Classification Using ML and Rule-Based Systems.
- [24] Mishra et al. (2021). AI-Based Candidate Screening: A Study on Resume Parsing and Selection Models.
- [25] Bose et al. (2023). Enhancing Recruitment with AI-Powered Resume Screening and Job Matching.



International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975

Impact Factor: 9.971