NAME: V. Kavya Sahithi

Reg. No.: 22BCE9568

Building a Resume Parser Using ChatGPT

ABSTRACT

In today's fast-paced recruitment ecosystem, the manual processing of resumes presents a considerable challenge for HR professionals and recruiters. With companies receiving hundreds or even thousands of job applications for a single position, the need for an automated, intelligent, and accurate resume parsing system has become more critical than ever. This project aims to build a Resume Parser using Natural Language Processing (NLP) techniques in Python to automatically extract vital candidate information such as name, email, phone number, skills, education, and work experience from resumes. The system is designed to streamline the candidate shortlisting process and make hiring more efficient.

The project is implemented in a Google Colab environment, which allows for cloud-based development, making it easily accessible and shareable. The resumes are uploaded in a compressed ZIP format and may include .pdf or .docx files. Once uploaded, they are extracted into a folder, and each file is individually processed to extract relevant details. The design of the system ensures modularity, where each component (e.g., email extraction, phone number parsing, skill matching) is developed as a separate function, allowing for flexibility and future enhancements.

Several Python libraries and technologies are employed in this project. The PyPDF2 library is used for reading PDF documents, while docx2txt is used to process DOCX files. spaCy, an advanced NLP library, plays a crucial role in identifying named entities such as the candidate's name. For pattern recognition tasks such as extracting emails and phone numbers, Python's built-in re (regular expressions) module is used. The pandas library is utilized to structure the parsed data into a tabular format, which is then exported as a CSV file for further use or analysis. Additionally, Google Colab's files module is used for seamless file uploads and downloads.

The overall design follows a sequential flow. Initially, resumes are uploaded and extracted. Next, each resume is processed using the extract_text_from_file() function that handles file-specific text extraction. The extracted raw text is passed through a series of parsing functions such as extract_name(), extract_email(), extract_phone(), extract_skills(), extract_education(), and extract_experience(). The results are collected into a structured Python dictionary, appended to a list, and finally converted into a DataFrame using pandas. This DataFrame is saved as a CSV file (parsed_resume_dataset.csv) which can be easily downloaded and integrated into recruitment systems or dashboards.

The output of the system is a structured dataset containing key information from each resume. This output can significantly reduce the time required for initial resume screening and can be integrated into larger HR systems for candidate ranking or recommendation. While the current system uses static keyword lists for skills and educational qualifications, it can be enhanced using machine learning models or custom-trained NLP pipelines to recognize a broader set of entities and relationships within resume texts.

In conclusion, this resume parser demonstrates the effective use of NLP techniques for automating HR processes. It highlights how open-source tools can be leveraged to build scalable and adaptable solutions for real-world challenges. The expected outcome is a functional system that not only reduces the manual effort in resume screening but also enhances the accuracy and consistency of candidate evaluation. With potential for future improvements such as skill relevance scoring and job-role matching, this project lays the groundwork for intelligent resume analysis systems in modern recruitment workflows.