**Resume Classification Using NLP**

***1. Problem Statement:***

In the competitive world of modern hiring, organizations are often inundated with a large number of resumes for every job opening. These resumes may vary drastically in format, content, and structure, making it difficult for human recruiters to quickly and effectively evaluate each one. The traditional manual screening process is not only time-consuming but also susceptible to various challenges:

* **Scalability Issues**: As the number of applicants increases, the capacity of human reviewers to efficiently analyze resumes remains limited, leading to delays and operational bottlenecks.
* **Inconsistency and Human Bias**: Different reviewers may interpret resumes differently, and unconscious biases may affect decision-making. Candidates from non-traditional backgrounds or underrepresented groups may be unfairly filtered out.
* **Keyword Dependency**: Manual review often relies on quickly spotting keywords, which may not reflect true candidate potential, especially if the resume uses synonyms or varied phrasing.
* **Time Pressure**: Under tight deadlines, recruiters may overlook potentially well-qualified candidates or make hasty decisions.

To overcome these issues, there is a need for an automated, intelligent system that can interpret and evaluate resumes with a high degree of accuracy and consistency.

This project focuses on developing an **NLP-based Resume Classification System** that:

* Automatically reads and interprets resume content regardless of layout or structure.
* Classifies resumes into predefined job roles such as Data Scientist, Web Developer, Software Engineer, Human Resource, and others.
* Supports HR teams by pre-filtering resumes and ranking them according to job relevance.
* Offers a scalable solution that maintains consistent decision criteria across all applicants.
* Enables faster shortlisting and helps hiring teams focus on interviewing rather than screening.

In essence, this project seeks to apply Natural Language Processing (NLP) and Machine Learning (ML) to modernize and optimize the recruitment process. The long-term goal is to integrate such systems into applicant tracking systems (ATS), enhancing not only operational efficiency but also the overall quality of hires by focusing on data-driven evaluation.

***2. Introduction:***

In today's fast-paced, competitive job market, recruitment processes are evolving rapidly to keep pace with the increasing volume of applicant data. With the digital transformation of recruitment, organizations now face the challenge of processing massive numbers of resumes in varied formats, ranging from traditional documents to modern, creative templates. Manual evaluation of these resumes not only strains time and resources but also risks inconsistencies and unintentional bias.

**Leveraging Natural Language Processing (NLP):**

Natural Language Processing (NLP) is a branch of artificial intelligence that focuses on enabling computers to interpret, analyze, and generate human language. In this project, NLP serves as the cornerstone for transforming unstructured resume data into structured insights. By using various linguistic and statistical techniques, the system can:

* **Extract Key Information:** Identify essential data such as skills, qualifications, work history, and educational background.
* **Understand Context:** Capture the nuance and context behind job titles, project descriptions, and technical skills, even when presented in diverse formats.
* **Standardize Data:** Convert varied resume formats into a consistent representation, making it easier to compare and assess candidates.

***3. Project Objectives and Goals:***

The primary objective of this project is to automate the screening and categorization process of resumes. By classifying resumes into predefined job categories (e.g., Data Scientist, Web Developer, Software Engineer, HR), the system provides a more objective, efficient, and scalable solution. Specific goals include:

* **Enhancing Efficiency:** Reduce the time and effort required to sift through large volumes of resumes, enabling recruiters to focus on top candidates.
* **Improving Consistency:** Establish a uniform evaluation framework that mitigates personal biases or inconsistent review criteria.
* **Data-Driven Decision Making:** Offer insights backed by data analytics, helping to identify trends in candidate qualifications and job market demands.
* **Facilitating Advanced Analytics:** Lay the groundwork for integrating further enhancements like candidate ranking, job matching, and personalized recommendations based on the skill gap analysis.

***4. Broader Impact:***

Automating resume screening via NLP not only boosts operational efficiency for human resources but also supports diversity and inclusion initiatives by ensuring that every candidate is evaluated based on objective, data-driven criteria. Additionally, this approach can seamlessly integrate with existing Applicant Tracking Systems (ATS), providing real-time feedback and analytics to improve the overall hiring process.

In summary, this project combines the power of NLP and machine learning to create a robust, automated system that elevates the recruitment process. It offers tangible benefits such as reducing manual workload, increasing evaluation accuracy, and ultimately contributing to better hiring decisions.

The system will not only automate classification but can also assist in ranking candidates by relevance. It provides a foundation for further enhancements like matching candidates with job descriptions, identifying missing skills, and recommending courses for upskilling.

***5. Methodology:***

This project follows a structured pipeline using Natural Language Processing (NLP) and machine learning to classify resumes into job categories. The methodology includes data processing, visualization, feature extraction, model training, and evaluation.

**Data Collection**

* **Sources**: Resumes were gathered from multiple domains and labeled based on the job role (e.g., Data Scientist, HR, Web Developer, etc.).
* **Formats**: The documents were collected in .txt, .pdf, and .docx formats. Appropriate libraries like PyPDF2 and python-docx were used for text extraction.

**Text Extraction and Parsing**

* Extracted raw text from resumes.
* Parsed resumes into structured sections (e.g., Skills, Experience, Projects) if headings were available.
* Removed non-content elements like page numbers and template artifacts.

**Text Preprocessing**

Before feature extraction, resumes underwent a cleaning process to remove noise and standardize content. The cleanResume() function used regular expressions to clean the text by:

* Removing **URLs**, **hashtags**, and **mentions**
* Stripping **special characters**, **punctuation**, and **non-ASCII symbols**
* Normalizing **extra spaces**

**Exploratory Text Analysis and Visualization**

To gain initial insights into the textual data, two key visual tools were used:

* **Word Cloud Generation**:
  + Created word clouds to visually represent the most frequently occurring words in resumes across categories.
  + Larger words indicate higher frequency, giving intuitive insights into commonly mentioned skills and terms.
* **Top 10 Most Repeated Words**:
  + Identified and listed the top 10 most frequent words for each job category.
  + Helped in understanding which keywords dominate each class, aiding both manual interpretation and feature selection.

**Vectorization**

* **TF-IDF Vectorization:**Resume text was transformed into numerical feature vectors using TfidfVectorizer, with English stopwords removed. This technique captures the importance of each word relative to the entire dataset.

**Label Encoding**

* Each resume was tagged with a target label (e.g., HR, Web Developer, Data Scientist).
* Labels were encoded into integers for model compatibility.

**Model Building**

Three machine learning models were used:

* **KNeighborsClassifier (KNN)**:
  + Classifies resumes by comparing them to the closest labeled examples in the training data.
* **Support Vector Classifier (SVC)**:
  + Separates classes using optimal decision boundaries, especially effective in high-dimensional TF-IDF space.
* **RandomForestClassifier**:
  + Uses multiple decision trees to improve classification accuracy and robustness.

**Model Evaluation**

* **Accuracy**: Measures how often the model gets the label correct.
* **Precision & Recall**: Evaluated per category to understand relevance and completeness.
* **F1 Score**: Balanced metric to judge performance.
* **Confusion Matrix**: Visualized misclassifications and confusion between categories.

***6. Application Deployment with Streamlit/Gradio:***

To make the model accessible and user-friendly, it was integrated into an interactive web-based application using **Streamlit** and **Gradio**. This interface allows users to upload resumes or input raw text and receive real-time job category predictions—eliminating the need to run any backend code.

**Key Features:**

* **Resume Upload Interface**: Supports .txt, .pdf, and .docx file uploads.
* **Text Input Option**: Users can also directly type or paste resume text for instant classification.
* **Instant Prediction**: Automatically processes the input, applies preprocessing, and uses the trained model to predict the job role.
* **Clean UI**: Both Streamlit and Gradio offer lightweight, intuitive interfaces suitable for non-technical users.
* **Deployment Ready**: The app can be hosted locally or deployed on cloud platforms like **Streamlit Cloud** or **Hugging Face Spaces**.

This deployment bridges the gap between technical ML models and practical HR applications, enabling organizations to adopt resume classification with ease and efficiency.

**Deployed URL:** [**https://huggingface.co/spaces/Pramod345/resume-classifier**](https://huggingface.co/spaces/Pramod345/resume-classifier)

***7. Conclusion:***

This project demonstrates the effectiveness of combining Natural Language Processing (NLP) and machine learning to automate the classification of resumes. By cleaning and vectorizing text data, and applying models like KNeighborsClassifier, SVC, and RandomForestClassifier, the system accurately predicts the most suitable job category for each resume.

Beyond high classification performance, the solution also includes practical features like:

* **Real-time predictions** via a Streamlit or Gradio web app
* **Multiple input methods**, including file upload and direct text entry

The system offers significant benefits to HR departments by:

* **Reducing manual workload**
* **Accelerating candidate shortlisting**
* **Ensuring consistent and unbiased screening**

This project not only solves a real-world problem but also lays the foundation for more advanced applications like candidate-job matching, skill gap analysis, and personalized career guidance.

The deployment-ready interface ensures immediate usability by recruiters and hiring managers. Future enhancements could integrate semantic analysis models like BERT for deeper contextual understanding.

Overall, this work bridges the gap between AI innovation and practical HR solutions, empowering organizations to make smarter, faster, and fairer hiring decisions.