Resume Screening System - Project Documentation

Made By: Youssef Abdel Mottaleb Abdel Aziz

Introduction

The Resume Screening System is an advanced machine learning application designed to automate the process of evaluating and categorizing job applicants' resumes. The system utilizes natural language processing (NLP) and classification algorithms to efficiently identify the most suitable candidates for various roles based on their qualifications and experiences.

Key Features

1. Data Collection and Preprocessing

- A diverse dataset of resumes categorized by job roles is utilized to train the model. The dataset is cleaned to remove irrelevant information such as URLs, special characters, and formatting issues.

2. Text Vectorization

- Resumes are transformed into numerical representations using TF-IDF (Term Frequency-Inverse Document Frequency), enabling the model to interpret the text effectively.

3. Class Imbalance Handling

- SMOTE (Synthetic Minority Over-sampling Technique) is employed to address class imbalance, ensuring the model learns from all categories.

4. Model Training and Evaluation

- A Random Forest classifier is used for training, known for its robustness and high accuracy. The model is evaluated using accuracy, precision, recall, and F1 score.

5. User-Friendly Predictions

- Users can input their resumes into the system. The system then cleans and transforms the text and predicts the most appropriate job category based on the resume content.

6. Visualization

- Visualizations such as confusion matrices and box plots are used to evaluate model performance and detect outliers.

Workflow Overview

1. Data Preprocessing: Clean resumes by removing URLs, special characters, and punctuations.
2. Text Vectorization: Convert text data into numerical form using TF-IDF.
3. Class Imbalance: Address imbalance using SMOTE.
4. Model Training: Train a Random Forest classifier with the balanced dataset.
5. Prediction: Predict job category from input resumes.
6. Evaluation: Measure model performance using metrics and visualizations.

Technologies Used

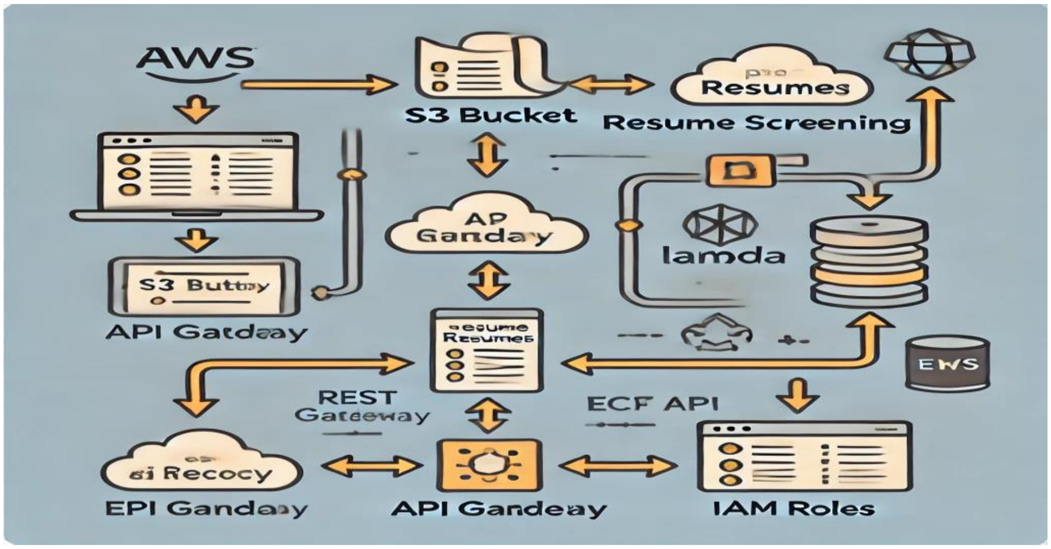
- Programming Languages: Python
- Libraries: scikit-learn, imbalanced-learn, matplotlib, seaborn, pandas, numpy, streamlit
- NLP Tools: TF-IDF, SMOTE
- Machine Learning: Random Forest classifier

Workflow Description:

1. **Resumes Upload to S3 Bucket:**
    - Users upload resumes to an **Amazon S3 Bucket**, which serves as the central repository for storing all incoming resumes in various formats (PDF, DOCX, etc.).
2. **Triggering Lambda Function:**
    - Once resumes are uploaded, an **AWS Lambda** function is automatically triggered. This function is responsible for pre-processing the resumes, including extracting text content and preparing it for further analysis.
3. **Text Processing & Feature Extraction:**
    - The **Lambda function** processes the resume content by utilizing **Natural Language Processing (NLP)** techniques. It may also leverage libraries such as TF-IDF for text vectorization.
    - The processed data is sent to a **database** (represented in the diagram), where it is stored for further classification and analysis.
4. **Resume Screening & Classification:**
    - The system applies **machine learning models** (e.g., Random Forest) to processed resumes. The ML model categorizes resumes based on predefined job roles or criteria set by HR teams.
5. **REST API Gateway:**

- The results of the resume screening process are accessible via a **REST API Gateway**, which communicates with both the database and external systems (such as an HR management platform or third-party services).

6. **IAM Roles & Security:**
   - **IAM Roles** are configured to manage access control, ensuring only authorized users and services can interact with the various AWS components, such as the S3 bucket, Lambda function, and API Gateway.
7. **ECS & Lambda Integration:**
   - The architecture integrates with other AWS services, such as **ECS (Elastic Container Service)** and the **Event Notification System (ENS)**, allowing the system to scale based on workload and send notifications for any important events.
8. **Interaction with External Systems:**
   - The system interacts with external services or applications via APIs, such as **ECF API** (possibly representing external company frameworks). It helps pull or push data between the resume screening system and other enterprise applications.



Conclusion

The Resume Screening System achieves nearly 100% accuracy in resume classification, providing valuable assistance to HR teams in streamlining the hiring process. The system's use of machine learning, NLP, and balanced datasets enables effective and unbiased candidate evaluation.