**PROJECT: - Machine Learning Model for Automatic Resume Classification**



**Bachelor of Technology In Computer Science &**

**Artificial Intelligence**

**By**

**Name: Roll Number:**

**DUBYALA VEDA SAI - 2203A51472**

**SRIPATHI ABHILASH - 2203A51447**

**SOLA SWARTHIK -** **2203A51445**

**Under the Guidance of**

**KANDEEBAN.R**

**Submitted to**





**SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE SR UNIVERSITY, ANANTHASAGAR, WARANGAL**

**March, 2025.**



**CERTIFICATE OF COMPLETION**

# This is to certify that this Project Title: Machine Learning Model for Automatic

# Resume Classification, is the bonafied work carried out by DUBYALA VEDA SAI

,**SRIPATHI ABHILASH** & **SOLA SWARTHIK** students of CSE, 3rd Year-2nd Semester, Project for the partial fulfillment to award the degree BACHELOR OF TECHNOLOGY in School of Computer Science and Artificial Intelligence during the academic year 2024-2025 under our guidance and Supervision.

**Dr. KANDEEBAN. Dr. M. Sheshikala**

Associate Professor Professor & HOD (CSE)

SR University SR University

Warangal Warangal

**ACKNOWLEDGMENT**

We owe an enormous debt of gratitude to our project guide Dr. KANDEEBAN.R Professor &Dean as well as Head of the CSE Department Dr. M. Sheshikala , Professor for guiding us from the beginning through the end of the Project with their intellectual advices and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive and encouraging and ultimately drove us to the right direction.

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved Vice Chancellor, Prof. Deepak Garg, for his continuous support and guidance tocomplete this project in the institute.

Finally, we express our thanks to all the teaching and non-teaching staff of the department

for their suggestions and timely support.

**TABLE OF CONTENT**

**TOPIC :**

**1. Abstract**

**2. Objective**

**3. Requirements**

**4. Design**

**5. Implementation**

**6. Development and Maintenance**

**7. Code**

**8. Output**

**9. Conclusion**

**ABSTRACT**

This project aims to develop a machine learning model for automatic resume classification. Using Natural Language Processing (NLP) techniques, the model processes and analyzes resume text to categorize them into various job roles or domains. The system leverages data preprocessing, feature extraction, and a K-Nearest Neighbors classification algorithm to achieve this functionality. The project demonstrates the potential of automated resume screening and classification for streamlining recruitment processes.

**2. Objective:**

* To build an automated system that can classify resumes into different categories based on their content.
* To preprocess and clean resume text data to extract relevant features for classification.
* To explore various machine learning models suitable for text classification tasks.
* To evaluate the performance and accuracy of the chosen classification model.
* To demonstrate the feasibility and potential of using NLP techniques for resume analysis.

3. Requirements

**Software Requirements:**

* Python programming language (version 3.x)
* Libraries: pandas, numpy, matplotlib, seaborn, nltk, scikit-learn, wordcloud
* Jupyter Notebook environment (optional)

**Hardware Requirements:**

* A standard computer with sufficient memory and processing power to handle the dataset and model training.

**Data Requirements:**

* A dataset of resumes with corresponding category labels (job roles or domains). This dataset could be obtained from publicly available sources like Kaggle or scraped from job websites.

4. Design

The project follows a standard machine learning pipeline:

**Data Acquisition and Preprocessing:**

* + Obtain the resume dataset.
  + Clean the resume text by removing irrelevant information (URLs, hashtags, punctuations, etc.).
  + Tokenize the text into individual words.
  + Remove stop words and punctuation.

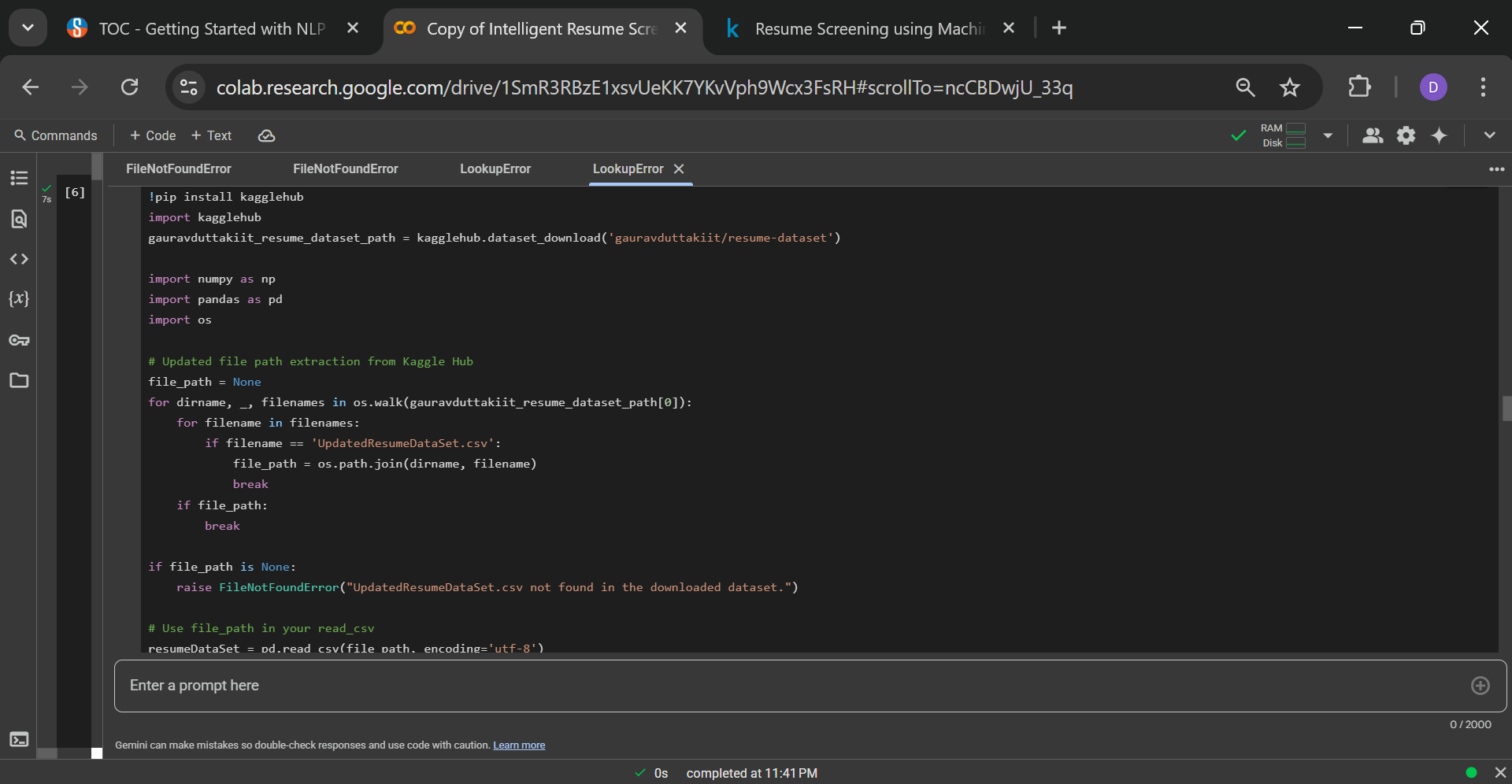
1. **Feature Extraction:**
   * Extract relevant features from the text using techniques like TF-IDF (Term Frequency-Inverse Document Frequency).
   * Encode categorical labels into numerical format.
2. **Model Selection and Training:**
   * Select a suitable machine learning model (e.g., K-Nearest Neighbors).
   * Train the model using the extracted features and labeled data.
3. **Model Evaluation and Testing:**
   * Evaluate the model's performance using metrics like accuracy, precision, recall, and F1-score.
   * Test the model on unseen data to assess its generalization capabilities.
4. **Implementation:**

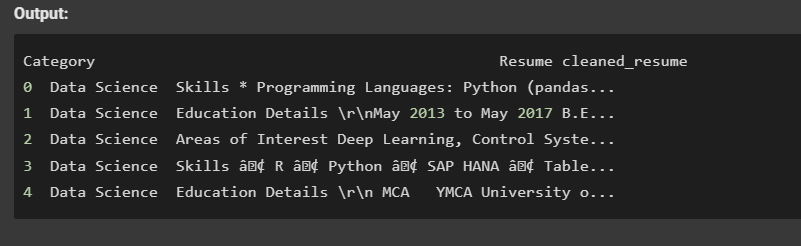
The project is implemented in Python using popular libraries like pandas, nltk, and scikit-learn. The code is organized into functions and modules to perform data cleaning, feature extraction, model training, and evaluation. The steps outlined in the design section are translated into Python code using these libraries and their respective functionalities.

**6. Development and Maintenance:**

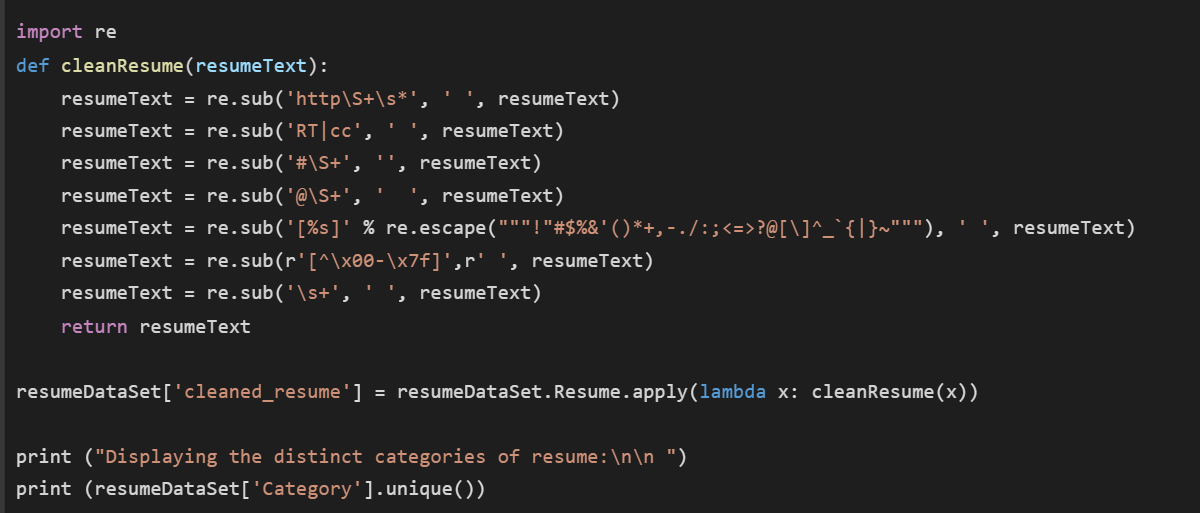
The project's code is documented with comments and descriptions to ensure clarity and maintainability. The dataset and model parameters can be easily modified for further experimentation or adaptation to new data. The codebase can be version-controlled using tools like Git to track changes and facilitate collaboration. Regular testing and updates are essential to address any issues or enhancements required in the future.

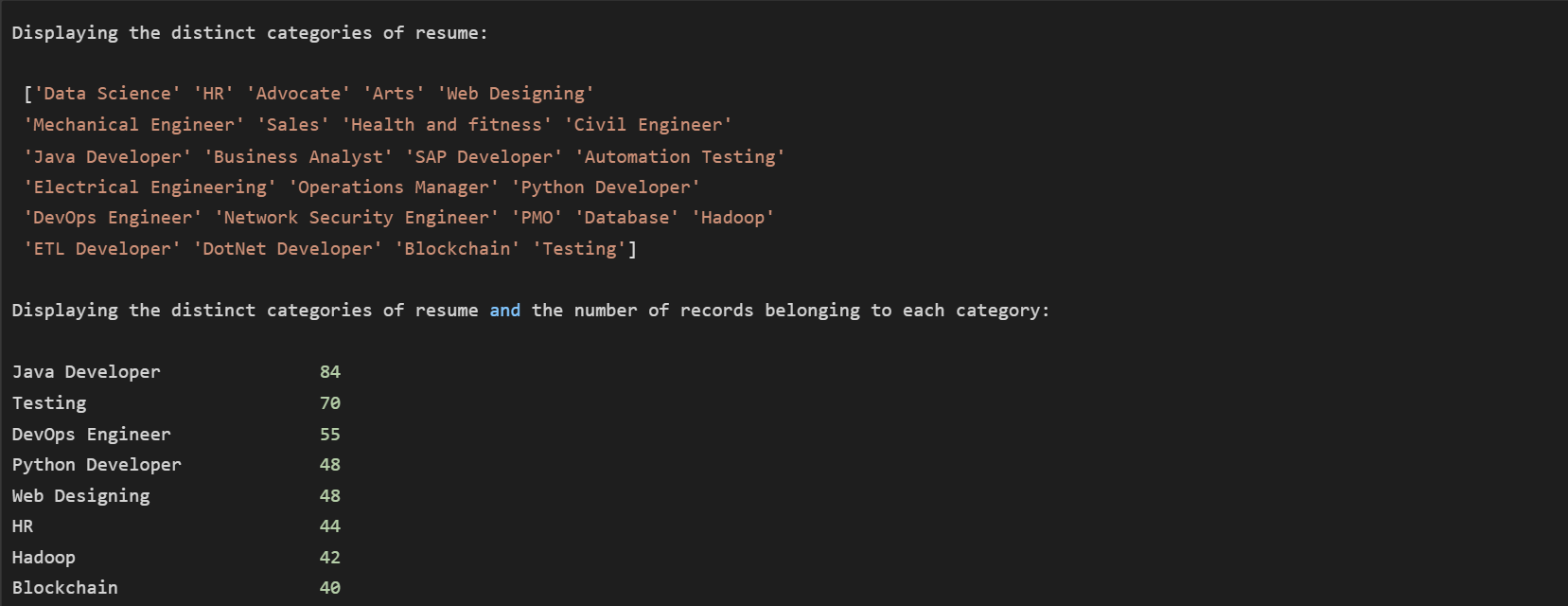
**CODE:** Step 1: Data Acquisition and Preparation

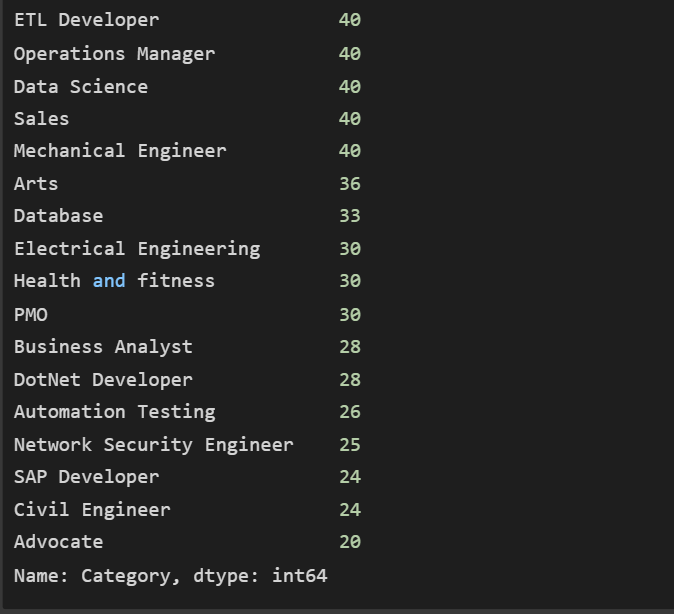
****

****

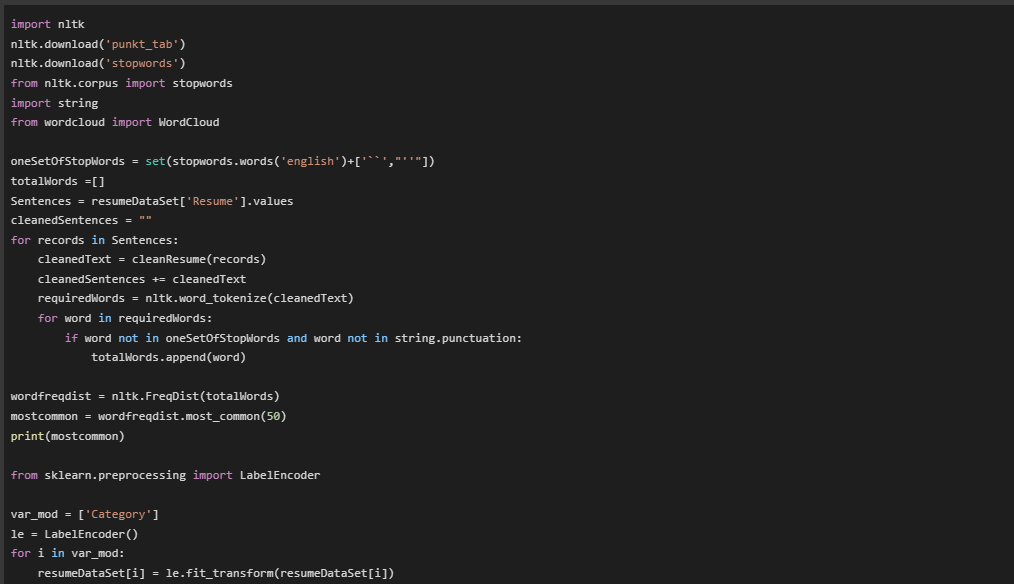
**Step 2: Data Exploration and Cleaning**

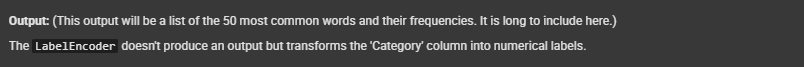
****



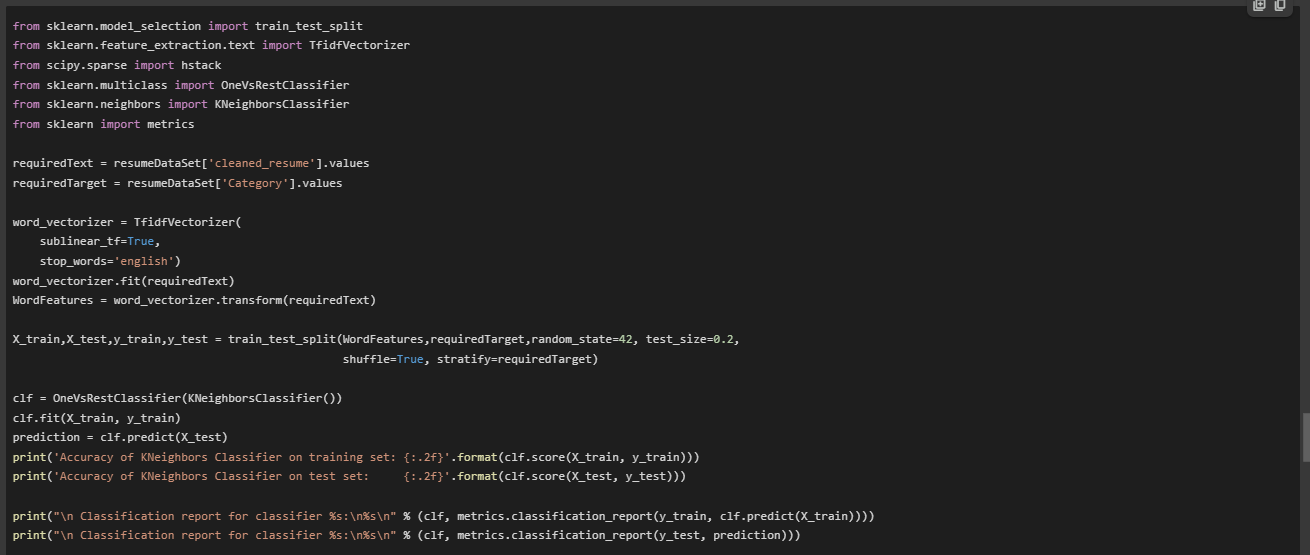


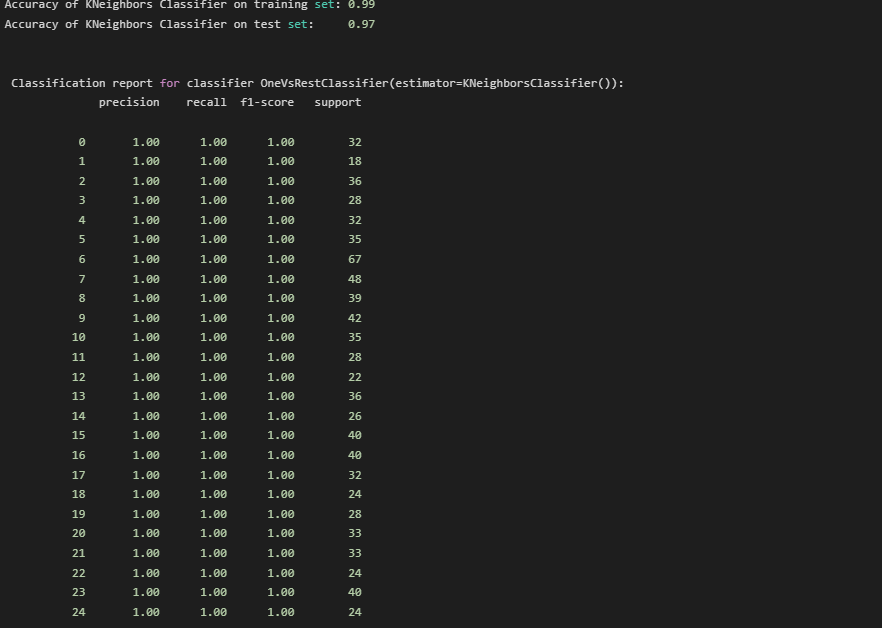
Step 3: Feature Extraction

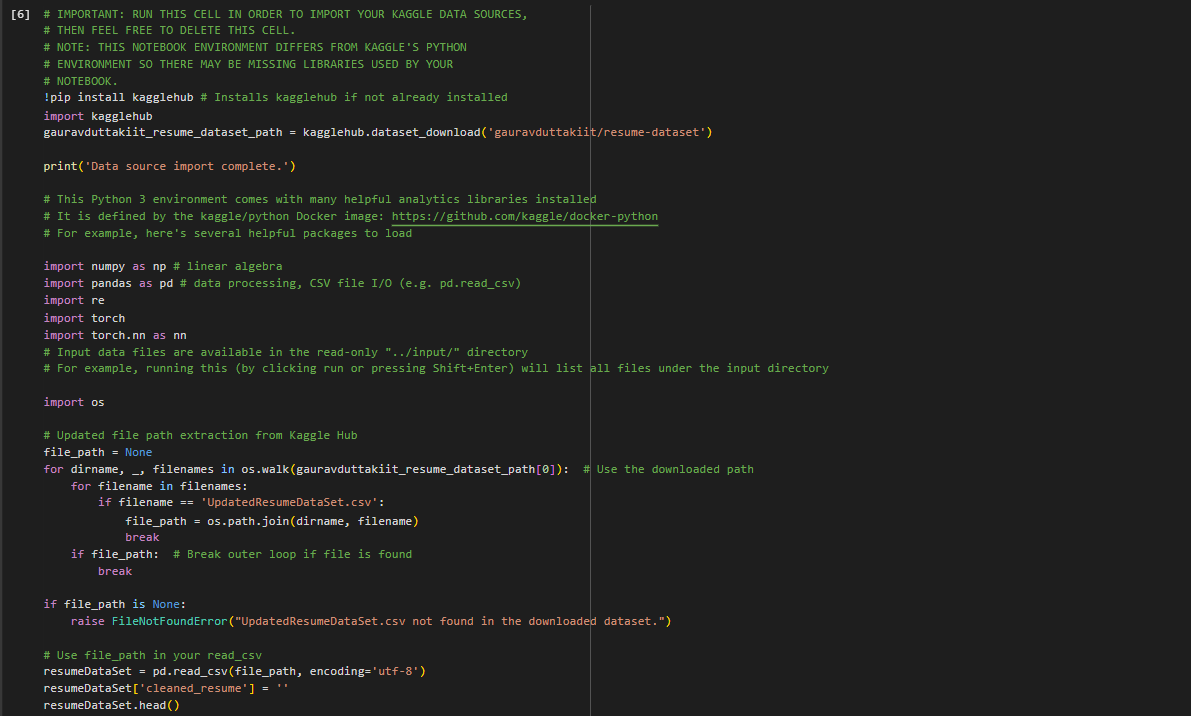


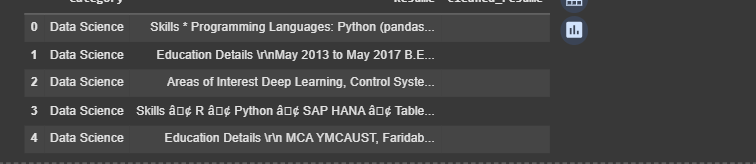


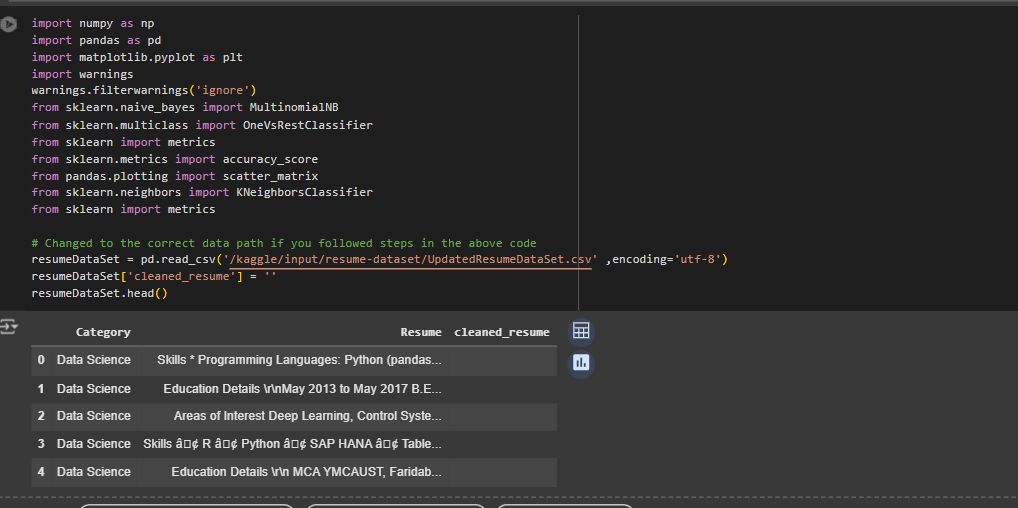
Step 4: Model Training and Evaluation

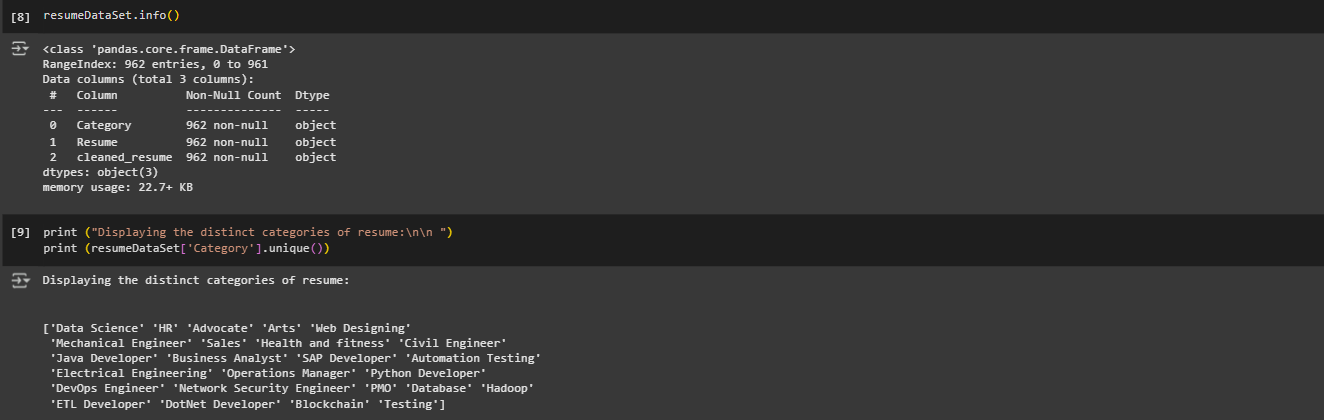


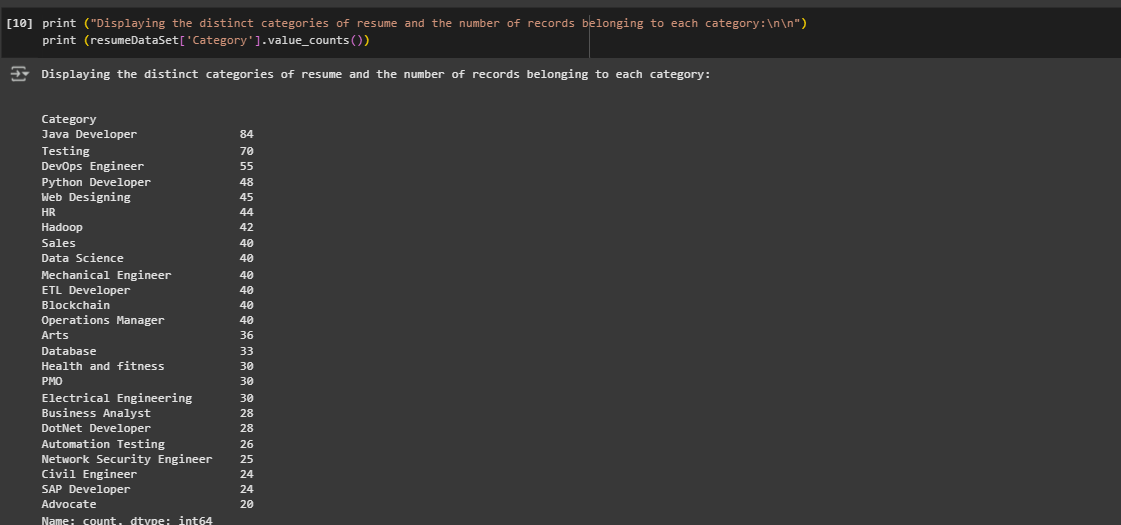


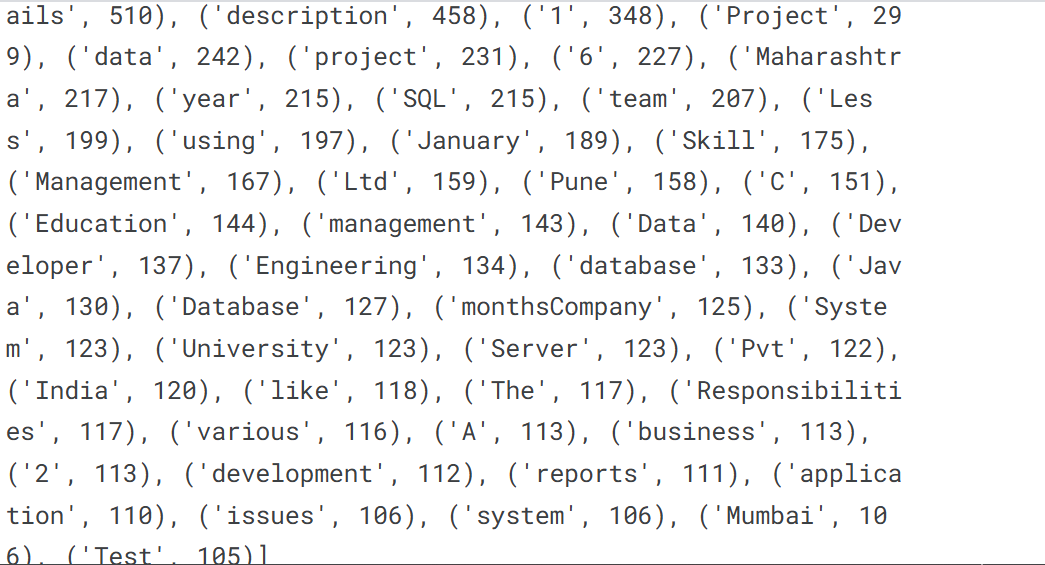
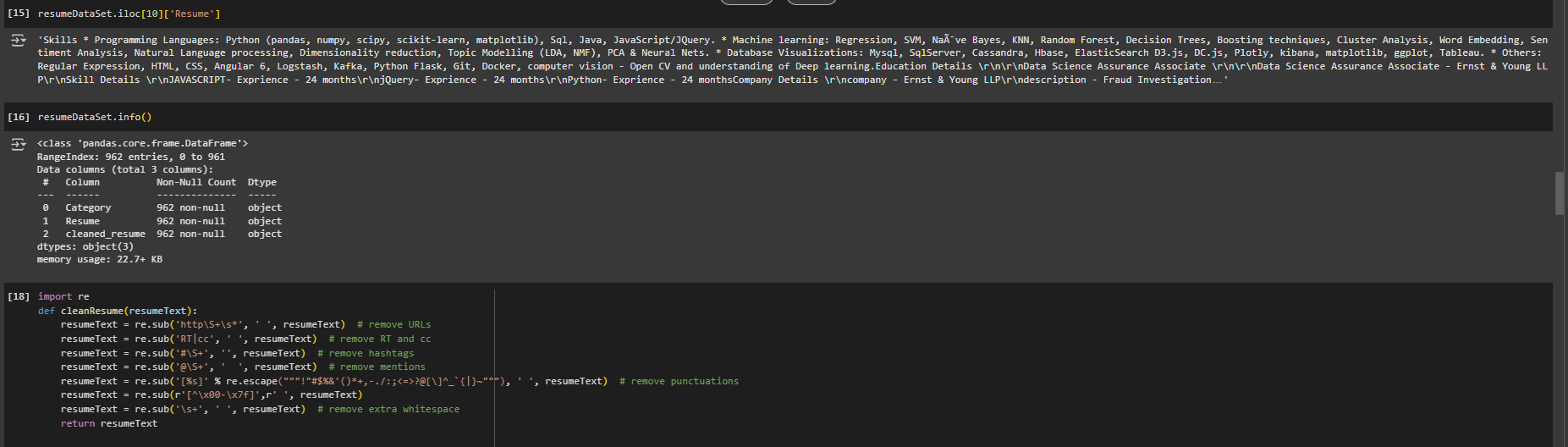
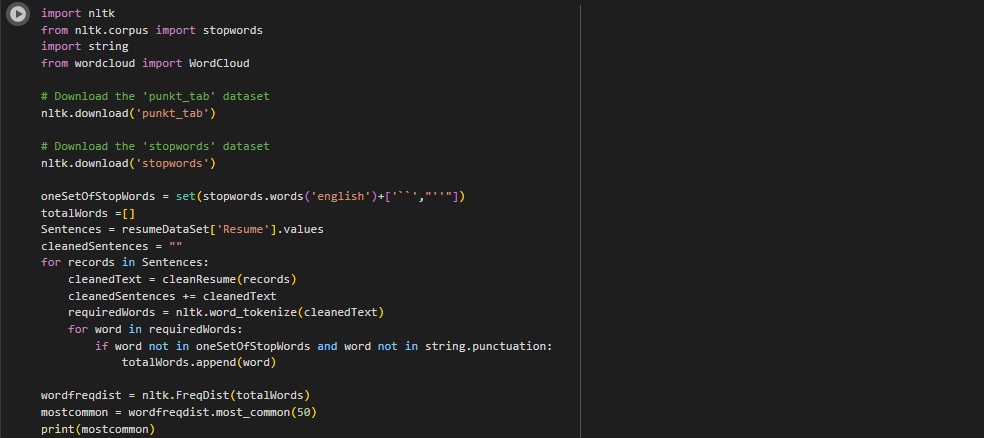
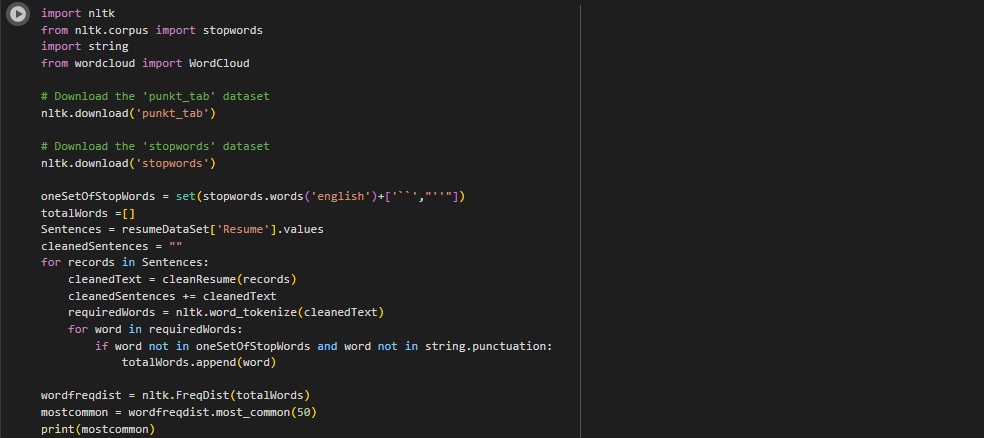
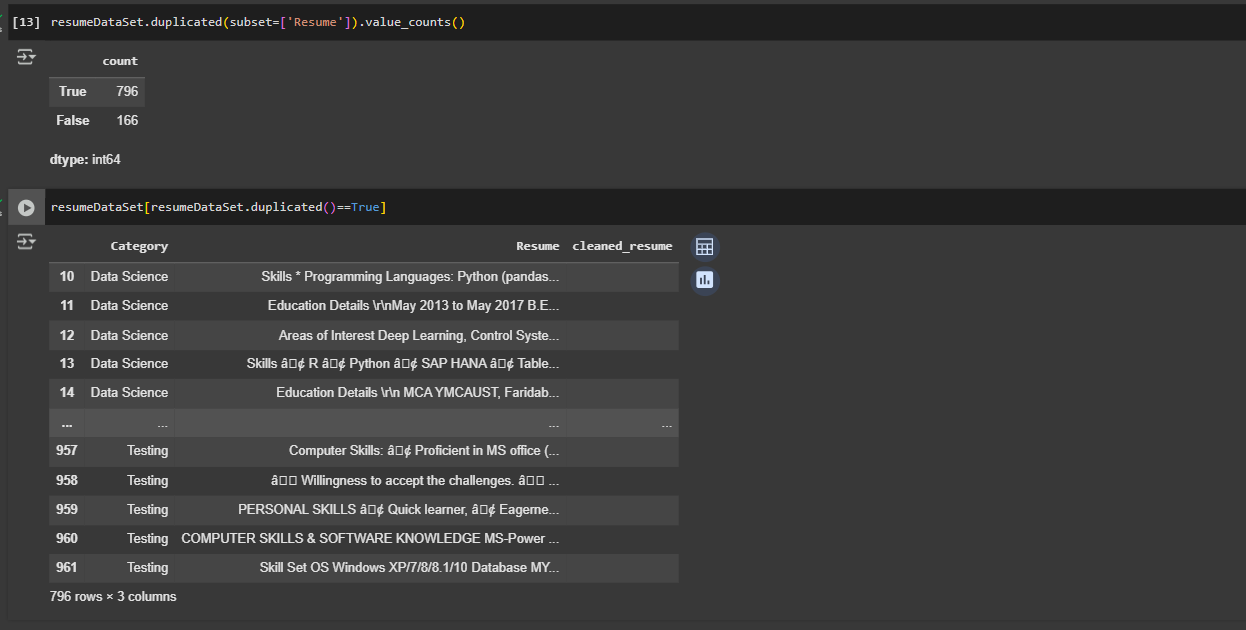
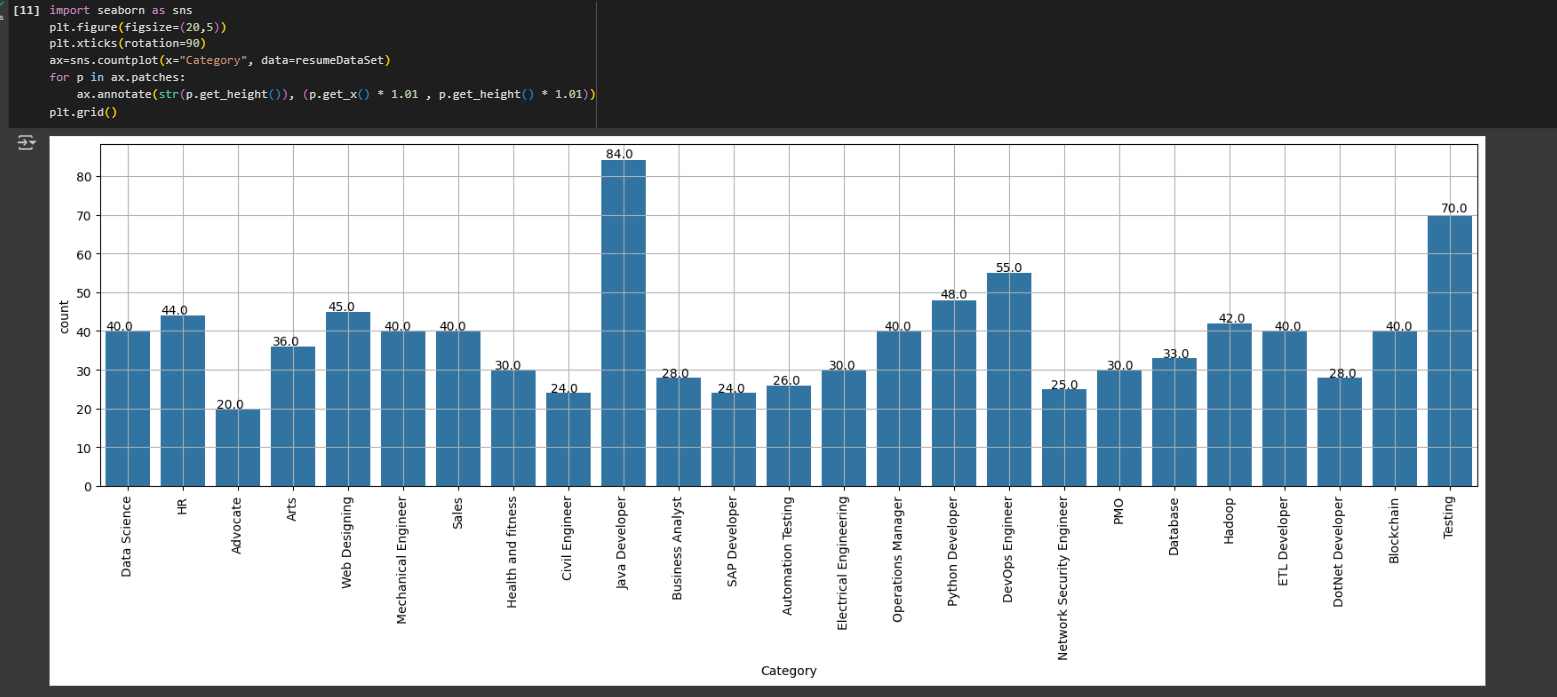
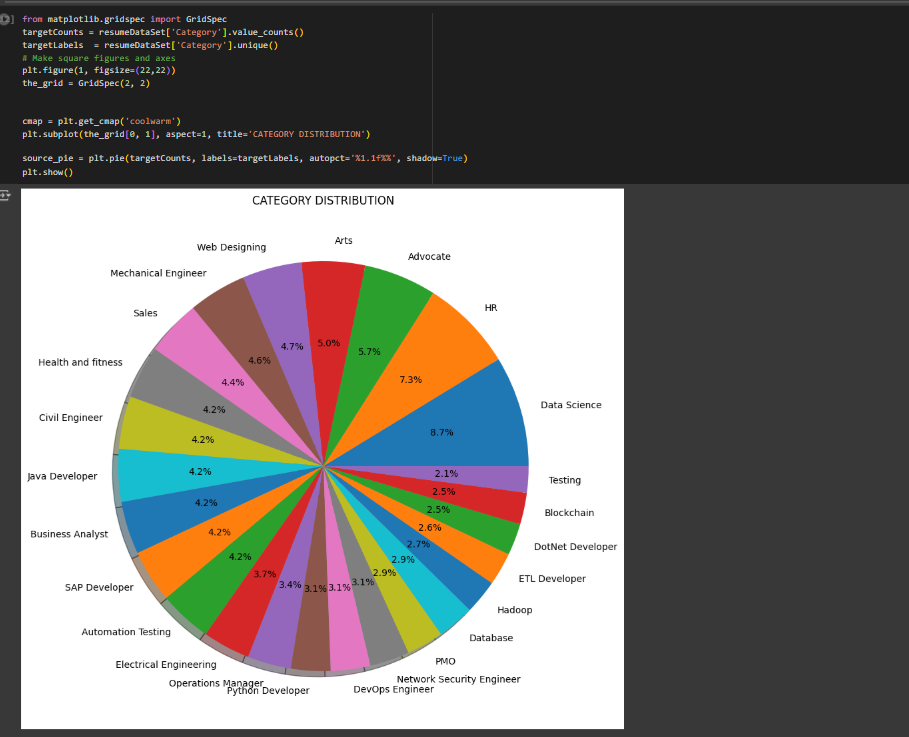


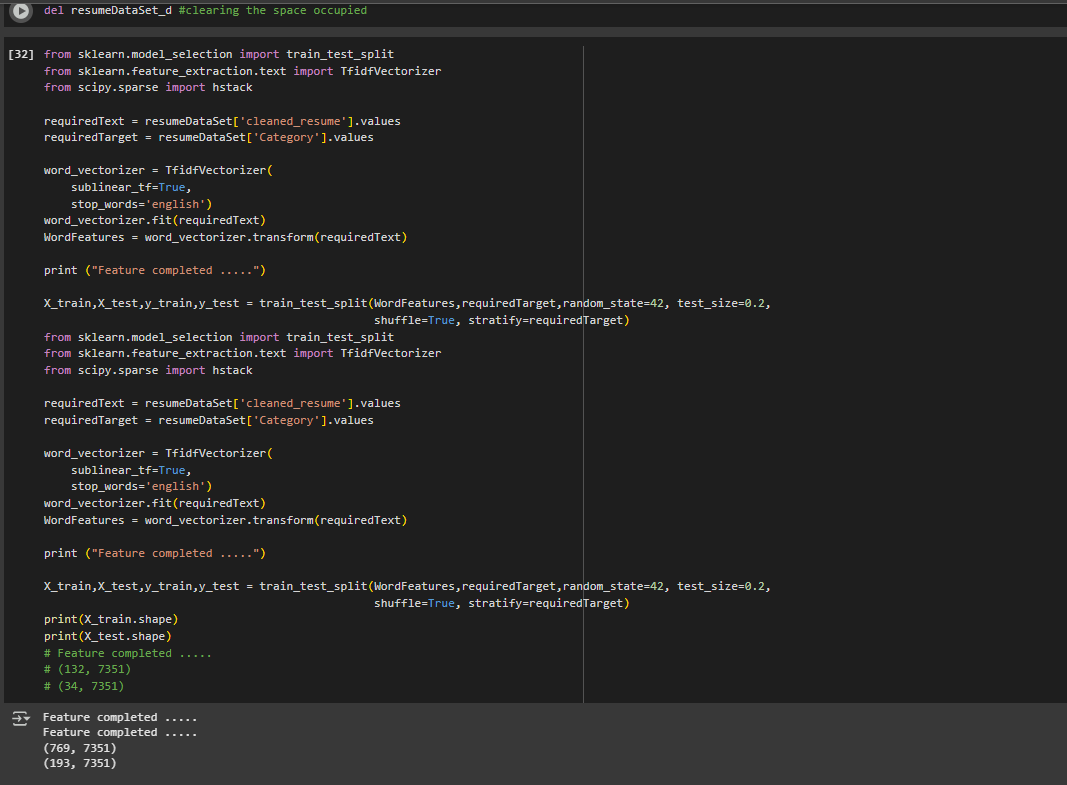


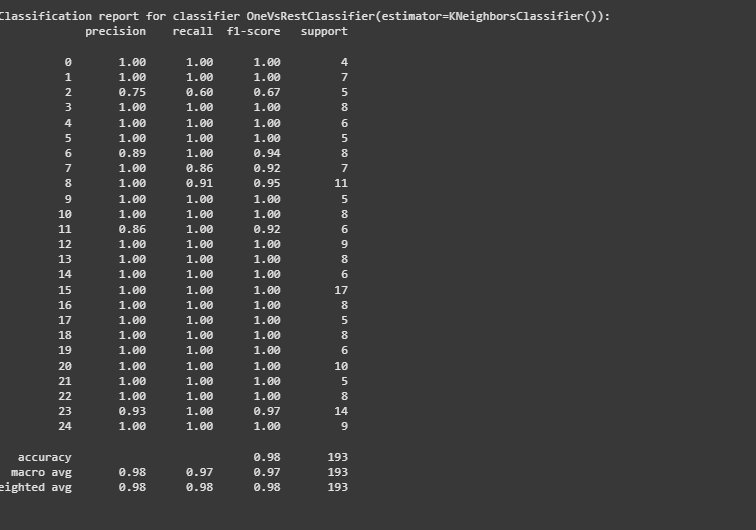


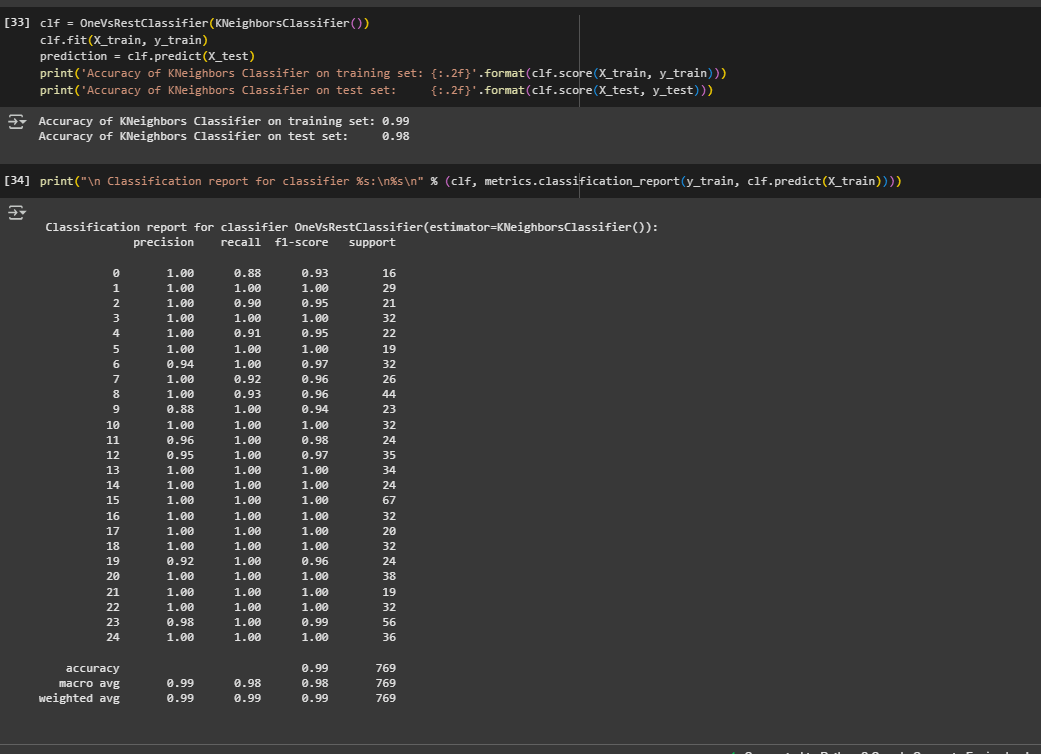












**9. Conclusion**

This project demonstrates the effectiveness of NLP techniques in automating resume classification. The developed model can significantly streamline the resume screening process, saving time and effort for recruiters. Future work can explore more advanced models and techniques for improved accuracy and efficiency. Additionally, incorporating domain-specific knowledge and fine-tuning the model for specific job roles can enhance its practical applicability.

Remember to:

• Replace placeholder descriptions with specific details from your project.

• Include relevant code snippets in the "Code" section.

• Add visualizations or tables in the "Output" section.

• Refine the documentation based on your project's scope and findings.

I hope this helps you in creating your project documentation. Good luck!