Machine Learning Engineer Task: Resume Categorization

Objective: Design, implement, and train a machine learning model to automatically categorize resumes based on their domain (e.g., sales, marketing, etc.). Following this, develop a script that can be run via the command line to process a batch of resumes, categorize them, and output results to both directory structures and a CSV file.

Task Breakdown:

1. **Find and Download the Dataset:**
   - Find and download the dataset from https://www.kaggle.com/datasets/snehaanbhawal/resume-dataset
   - You can use any other dataset if you want. If you decide to do so, make sure to add a link to your dataset in your report.
2. **Data Exploration and Preprocessing:**
   - Examine the dataset provided and understand the distribution of the different categories.
   - Process the resumes to convert them into a format suitable for training (tokenization, feature extraction, etc.).
   - Split the dataset into training, validation, and test sets.
3. **Model Selection and Training:**
   - Select a suitable machine learning or deep learning model.
   - Implement and train the model using the training set.
   - Evaluate the model's performance using the validation set and refine it for better accuracy and efficiency.
4. **Script Development:**
   - Create a Python script named `script.py`.
   - The script should take as input a directory containing the resumes to be categorized.
   - Using the trained model, the script should categorize each resume.
   - For each resume, move it to the respective category folder (e.g., if a resume is classified as 'sales', move it to a 'sales' folder within the directory; if there isn't a sales folder, create one).

- The script should also create and write to a CSV file named `categorized_resumes.csv`. The CSV should have two columns: `filename` and `category`.

5. **Command Line Execution:**
   - The script should be executable from the command line as follows: `python script.py path/to/dir`

Documentation:

- Alongside your code, provide clear documentation on:
  - The chosen model and rationale behind the selection.
  - Any preprocessing or feature extraction methods employed.
  - Instructions on how to run the script and expected outputs.

6. **Evaluation Metrics:**
   - Provide metrics like accuracy, precision, recall, and F1-score of your trained model using the test dataset.
   - The model will be run on completely unseen data, which will account for a significant portion of the score.
   - Any additional insights or visualizations on the model's performance will be a plus.

Deliverables:

1. Jupyter Notebook or Python scripts detailing your exploration, preprocessing, and model training process.
2. The final trained model file.
3. `script.py` script.
4. `categorized_resumes.csv` (as a sample output after running your script on a test set).
5. Documentation and instructions.