Resume Image Classifier

1. Approach

The approach to the assignment involves the development of a resume image classifier using a Convolutional Neural Network (CNN). The choice of a CNN is well-founded due to its effectiveness in image classification tasks, particularly in capturing intricate patterns and features in resume images. The approach prioritizes identifying distinctive visual elements in resumes for adequate classification.

2. Model Selection Rationale

Model Architecture:

The chosen model architecture is a sequential CNN with the following key components:

- Convolutional layers for feature extraction.
- MaxPooling layers for spatial down-sampling.
- Fully connected layers for classification.

Justification:

The sequential CNN architecture suits this image classification task, providing hierarchical feature extraction and spatial hierarchy understanding. The chosen number of layers and nodes in each layer balances the complexity and efficiency of the given assignment. Alternative models such as VGG16 or ResNet were considered unnecessary for the specific characteristics of resume images.

3. Coding Style

Well-Structured Code:

The code is well-structured, utilizing appropriate Python conventions for readability and maintainability. Each section is adequately commented on, clarifying the purpose and logic behind implemented steps.

Readability:

Meaningful variable names enhance code comprehension, and consistent formatting ensures a clean and professional coding style.

4. Model Results

Performance Metrics:

The model's performance is assessed using key metrics:

- **Accuracy: 81.3%**
- **Precision: 83.33%**
- **Recall: 83.33%**
- **F1-score: 83.33%**

Visualization:

The training history, including accuracy and loss over epochs, is visualized through plots. This provides an insightful representation of the model's behavior during training.

5. Reproducibility

Code Reproducibility:

The shared code includes comprehensive instructions to replicate the experiments successfully. Dependencies are clearly outlined, and preprocessing steps are detailed. Provided dataset paths and configurations enable the reproduction of model training and evaluation.

Transparency:

Comments guide users through key aspects of the code, ensuring a seamless reproduction experience and enhancing overall transparency.

Overall Evaluation

The approach is straightforward and thoughtful, leveraging a suitable model architecture for the given task. The code adheres to high readability standards and is well-commented, facilitating understanding and collaboration. Model results, including accuracy metrics and visualizations, are presented and structured. The shared code is designed for reproducibility, ensuring transparency and validation of reported results.