APPLIED MACHINE LEARNING

ASSIGNMENT 1

TRAN DUC ANH DANG

103995439

Methodology

Base Model Selection and Configuration:

- **ResNet-50 Backbone**: At the core of the **BirdClassifier** is the pre-trained ResNet-50 network, chosen for its deep residual learning capabilities which facilitate the training of deeper network architectures. This pre-trained model provides a complex hierarchy of image features, originally trained on the broad ImageNet dataset. Such a rich feature set is critical for handling the complex image recognition tasks required by the CUB-200 dataset.
- **Transfer Learning**: By adopting transfer learning, the **BirdClassifier** can leverage previously learned patterns and features, significantly reducing the need for extensive training data and accelerating the learning process. This approach is particularly beneficial given the dataset's quality issues, allowing the model to achieve higher accuracy by fine-tuning pre-trained layers rather than building from scratch.

Fine-Tuning Strategy:

- **Selective Layer Training**: The model employs a selective fine-tuning strategy where the **fine_tune_start** parameter determines the starting point for trainable layers. In this configuration, all layers before the fifth are set to non-trainable, preserving universal features that are broadly applicable across various image recognition tasks. Layers from the fifth onwards are fine-tuned to adapt to the specific characteristics of bird species in the dataset.
- **Adaptability**: For scenarios demanding extensive adaptation to the dataset's unique characteristics, setting **fine_tune_start** to a negative value allows all layers to be trainable, enhancing the model's ability to adjust to specialized data.

Output Layer Reconfiguration:

- **Custom Fully Connected Layer**: The original output layer of the ResNet 50 is replaced with a tailored sequence designed to classify 200 bird species. This includes:
  - **Linear layer** reducing the feature dimension to 512.
  - **ReLU activation function** to introduce non linearity, aiding complex pattern recognition.
  - **Dropout layer** with a rate of 0.5 to prevent overfitting by randomly omitting subsets of features during training.
  - **Final Linear layer** mapping the 512 features to 200 output classes, corresponding to each bird species.

Regularization and Overfitting Mitigation:

- **Dropout**: Incorporated at a rate of 0.5 to ensure the model does not rely excessively on any neuron, thus promoting feature redundancy and robustness.
- **Layer Freezing**: By freezing initial layers, the **BirdClassifier** minimizes overfitting on more abstract features, which are more universal and less specific to the task.

Training and Validation Process

- **Loss Function and Optimizer**: Using **Cross Entropy Loss**, suitable for multi class classification problems, and an Adam optimizer with an initial learning rate of 0.001.
- **Early Stopping**: Implemented to halt training if validation loss does not improve after a specified patience, thus avoiding overfitting.
- **Learning Rate Scheduler**: A **ReduceLROnPlateau** is used to decrease the learning rate by a factor of 0.1 after validation loss plateaus, helping to fine tune model weights more effectively in later training stages.

Data Handling and Augmentation

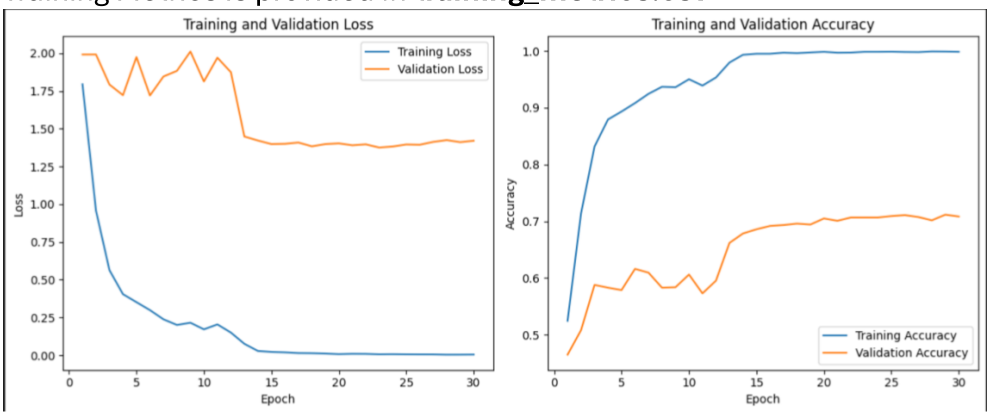
- **Image Transformations**: The training images undergo transformations such as resizing, centre cropping, tensor conversion, and normalization to make them suitable for the model. These steps standardize the input data and help the model learn more generalized features.

Performance Evaluation

- Implements an evaluation function that calculates Top-1 accuracy and average accuracy per class, ensuring a comprehensive assessment of the model's performance across individual classes and the dataset.

Result and Discussion

Training Metrics is provided in **training_metrics.csv**



Training Performance:

- **Loss Reduction**: The training loss showed a dramatic decline from 1.794 to 0.0047, indicating effective learning and adaptation of the model to the dataset.
- **Accuracy Increase**: The training accuracy improved significantly from approximately 52.47% to 99.86%. This increase reflects the model's ability to recognize and correctly classify the training images over time.

Validation Performance:

- **Loss Trends**: Initially, the validation loss decreased, reaching its lowest point at epoch 14 (1.4218). However, despite fluctuations, it did not significantly deteriorate, which suggests that the model was not overfitting to the training data.

- **Accuracy Improvement**: Validation accuracy started at 46.51% and progressively increased to a peak of 71.18% by epoch 29. This indicates a significant generalization to new, unseen data.

Analysis:

- **Early Performance**: The model struggled with lower accuracy and higher loss in both training and validation. This is typical as the model begins to learn from a relatively unoptimized state.
- **Mid Training Adjustments**: During training, as the model parameters adjusted and the learning rate likely adapted (as managed by the **ReduceLROnPlateau** scheduler), the model began to show more stable and improved validation results.
- **Late Training Observations**: During those last epochs, the training accuracy nearly reached perfection, which typically raises concerns about overfitting. However, the validation accuracy also improved, though more modestly, which implies that while the model is highly tuned to the training data, it still retains a reasonable level of generalization.

Considerations:

- **Potential Overfitting**: While training accuracy is extremely high, there is a discrepancy between training and validation accuracy, suggesting that the model might be overfitting. Techniques such as further increasing dropout rates, adding more data augmentation, or even employing more aggressive early stopping could potentially bridge this gap.
- **Validation Accuracy Plateau**: The fact that validation accuracy plateaus around 70% suggests that additional strategies may be needed to overcome potential biases or limitations in the model's learning capacity. Exploring alternative architectures, or more sophisticated image **preprocessing** methods, could provide further benefits.

Live Result:

```python
urls = {
    "Black footed Albatross": "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT6zzYyaoo-AfrGJYPLi_oQ6-0rY4pYKREPAkMEBktUtQ&s",
    "Eared Grebe": "https://leesbirdblog.files.wordpress.com/2014/07/pod-podi-horned-grebe-podiceps-auritus-c2a9wiki.jpg",
    "Indigo Bunting": "https://encrypted-tbn0.gstatic.com/licensed-image?q=tbn:ANd9GcR3JSKD9bUrn8SFlTx2IXhNYxkPpaoDz6S7fMdh2EE33UoMEf3E_5xcWrC7Bn_fNYVWirt-U7dkPdZwxKM",
    "White breasted Nuthatch": "https://holdenfg.org/wp-content/uploads/2021/03/nuthatch-351395-1920x1153.jpg",
    "Louisiana Waterthrush": "https://encrypted-tbn1.gstatic.com/images?q=tbn:ANd9GcT4xKBehNcCiOSy2zp9M6ZBDfok3F4aZPIalsuqqEf90bQnBmvs"
}

for label, url in urls.items():
    predicted, img = predict_online(model, url, label_map)
    print(f'Correct label: {label}, Predicted label: {predicted}\n | Correct: {bool(label == predicted)}\n | URL: {url}\n')
    # plt.imshow(img)
    # plt.show()
```

✓ 2.1s

```
Correct label: Black footed Albatross, Predicted label: Black footed Albatross
 | Correct: True
 | URL: https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT6zzYyaoo-AfrGJYPLi_oQ6-0rY4pYKREPAkMEBktUtQ&s

Correct label: Eared Grebe, Predicted label: Eared Grebe
 | Correct: True
 | URL: https://leesbirdblog.files.wordpress.com/2014/07/pod-podi-horned-grebe-podiceps-auritus-c2a9wiki.jpg

Correct label: Indigo Bunting, Predicted label: Indigo Bunting
 | Correct: True
 | URL: https://encrypted-tbn0.gstatic.com/licensed-image?q=tbn:ANd9GcR3JSKD9bUrn8SFlTx2IXhNYxkPpaoDz6S7fMdh2EE33UoMEf3E_5xcWrC7Bn_fNYVWirt-U7dkPdZwxKM

Correct label: White breasted Nuthatch, Predicted label: White breasted Nuthatch
 | Correct: True
 | URL: https://holdenfg.org/wp-content/uploads/2021/03/nuthatch-351395-1920x1153.jpg

Correct label: Louisiana Waterthrush, Predicted label: Louisiana Waterthrush
 | Correct: True
 | URL: https://encrypted-tbn1.gstatic.com/images?q=tbn:ANd9GcT4xKBehNcCiOSy2zp9M6ZBDfok3F4aZPIalsuqqEf90bQnBmvs
```