UNIVERSITÀ DEGLI STUDI DI MILANO

Muffin vs. Chihuahua Image Classification Using CNN

Course: Machine Learning

Professor: Nicolò Cesa-Bianchi

Student: Emil Soltanov

May 2024

Table of Contents

Introduction

This project aims to build a convolutional neural network (CNN) model to classify images of muffins and chihuahuas. The challenge lies in the visual similarity between muffins and chihuahuas, making it a compelling problem for image classification using deep learning techniques. The project employs the TensorFlow Keras framework to implement and train the CNN model.

Data Preparation

The dataset used for this project consists of 6000 images, divided into two categories: muffins and chihuahuas. The images were sourced from Kaggle's 'Muffin vs. Chihuahua' dataset.

Dataset Breakdown

- Total images: 6000
- Training and validation set: 4733 images
  - Chihuahuas: 2559 images
  - Muffins: 2174 images
- Test set: Remaining images

The images were preprocessed as follows:
1. Resized to 64x64 pixels to ensure uniformity.
2. Converted to RGB pixel values.
3. Split into training, validation, and test sets.
4. Labels assigned (1 for Muffin, 0 for Chihuahua).

Model Architectures

Three CNN architectures were implemented: AlexNet, VGG16, and ResNet. These architectures were chosen for their proven performance in image classification tasks.

AlexNet

AlexNet consists of 5 convolutional layers followed by 3 fully connected layers. ReLU activation functions and batch normalization were used after each convolutional layer, and max pooling was applied to reduce the spatial dimensions.

VGG16

VGG16 employs a simpler structure with only 3x3 convolutions and 2x2 pooling layers throughout the network. This architecture, known for its simplicity and effectiveness, was also implemented and evaluated.

ResNet

ResNet introduces residual connections to mitigate the vanishing gradient problem, allowing the training of very deep networks. This architecture's performance was compared against AlexNet and VGG16.

Training Process

The training process involved the following steps:
1. Data Loading and Preprocessing: Images were loaded and preprocessed into tensors suitable for training the neural network.
2. Training Loop: Implemented a loop to pass the input data through the network, calculate the loss using the Binary Cross-Entropy function, and update the model's parameters using backpropagation.
3. Loss Function: Binary Cross-Entropy was used to measure the difference between predicted probabilities and actual labels.
4. Optimizer: The SGD optimizer was tested with different learning rates (0.005, 0.01, and 0.1).
5. Batch Size: A batch size of 128 was used to train the model.
6. Epochs: The models were trained for 30 epochs.

Model Evaluation

The models were evaluated using 5-fold cross-validation to estimate performance and prevent overfitting. Performance metrics such as accuracy, zero-one loss, and entropy loss were recorded for each fold.

Experimental Results

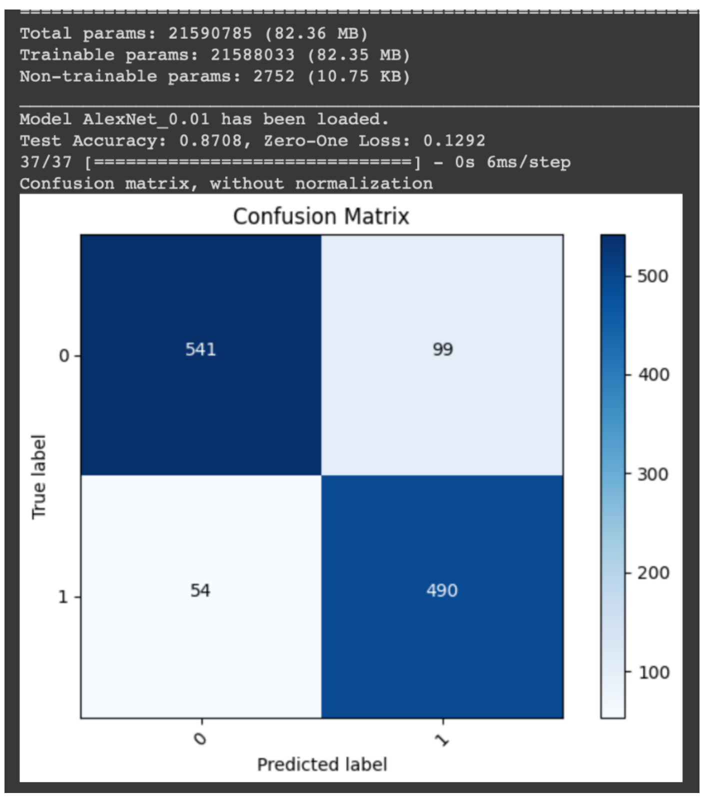
The following sections detail the test results for each CNN architecture using the optimal learning rate and optimizer.

AlexNet

- Optimizer: SGD with a learning rate of 0.01
- Test Accuracy: 0.8708
- Zero-One Loss: 0.1292

The model with SGD and a learning rate of 0.01 achieved an accuracy of 87.08%. It correctly classified 541 images of Chihuahuas and 490 images of Muffins. However, it misclassified 99 Chihuahuas as Muffins and 54 Muffins as Chihuahuas. This indicates a considerable number of false positives and false negatives, suggesting that while the model is relatively accurate, there is still room for improvement in reducing these errors.

Confusion Matrix:

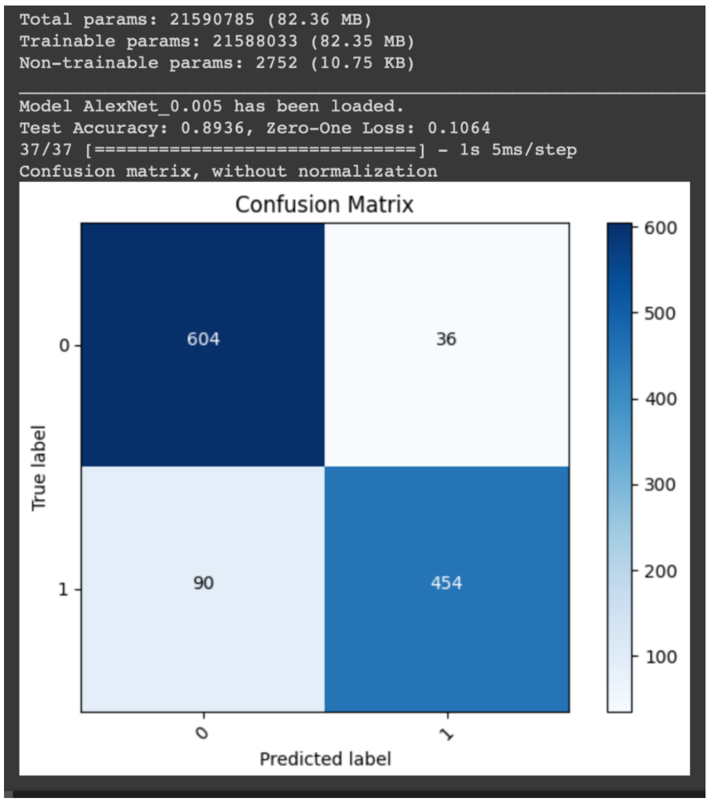


- Optimizer: SGD with a learning rate of 0.005

- Test Accuracy: 0.8936
- Zero-One Loss: 0.1064

With a learning rate of 0.005, the accuracy improved to 89.36%. The model correctly classified 604 images of Chihuahuas and 454 images of Muffins, with only 36 false positives and 90 false negatives. This significant reduction in false positives indicates better performance in distinguishing between the two classes.

Confusion Matrix:

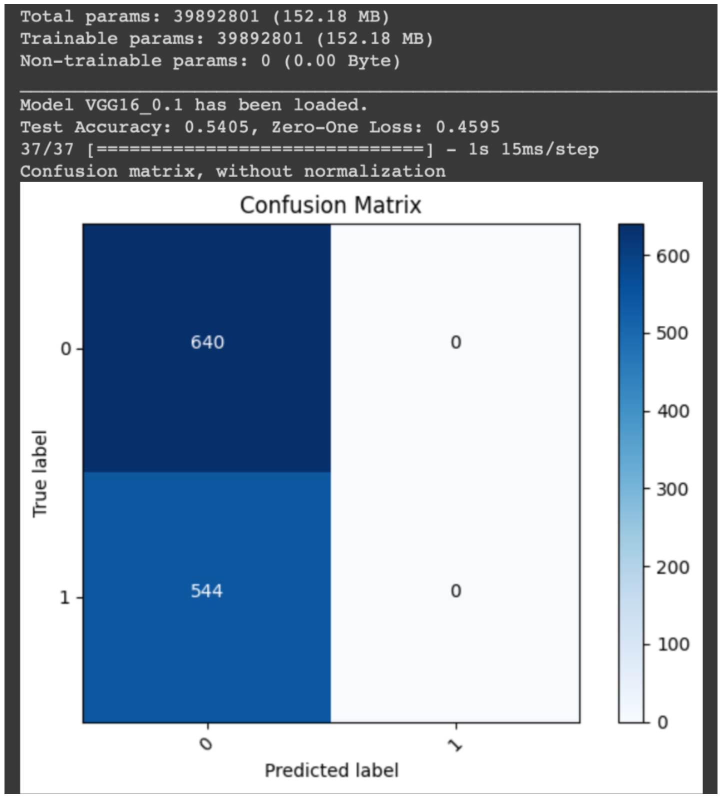


VGG16

- Optimizer: SGD with a learning rate of 0.1
- Test Accuracy: 0.5405
- Zero-One Loss: 0.4595

The model with a learning rate of 0.1 showed poor performance with an accuracy of 54.05%. It failed to classify any positive cases of Muffins, indicating severe overfitting or an excessively high learning rate, resulting in a total misclassification of Muffins (544 false negatives).

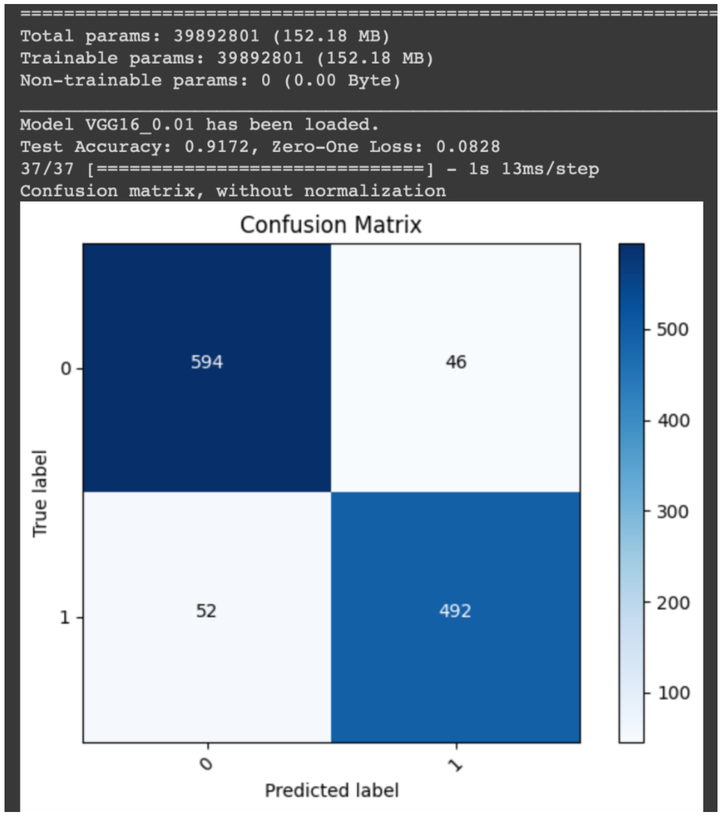
Confusion Matrix:



- Optimizer: SGD with a learning rate of 0.01
- Test Accuracy: 0.9172
- Zero-One Loss: 0.0828

At a learning rate of 0.01, the VGG16 model achieved the highest accuracy of 91.72%. It correctly classified 594 Chihuahuas and 492 Muffins, with only 46 false positives and 52 false negatives. This balanced and low number of misclassifications indicates excellent performance.

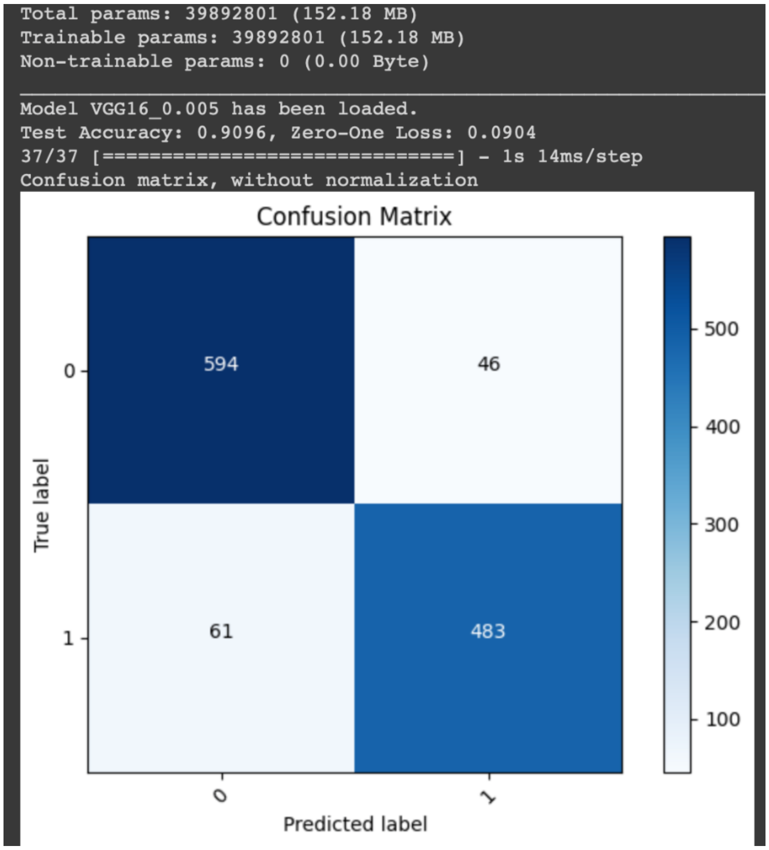
Confusion Matrix:



- Optimizer: SGD with a learning rate of 0.005
- Test Accuracy: 0.9096
- Zero-One Loss: 0.0904

The model with a learning rate of 0.005 showed a slightly lower accuracy of 90.96% but still performed well. It correctly classified 594 Chihuahuas and 483 Muffins, with 46 false positives and 61 false negatives. This shows good generalization with minimal misclassifications.

Confusion Matrix:

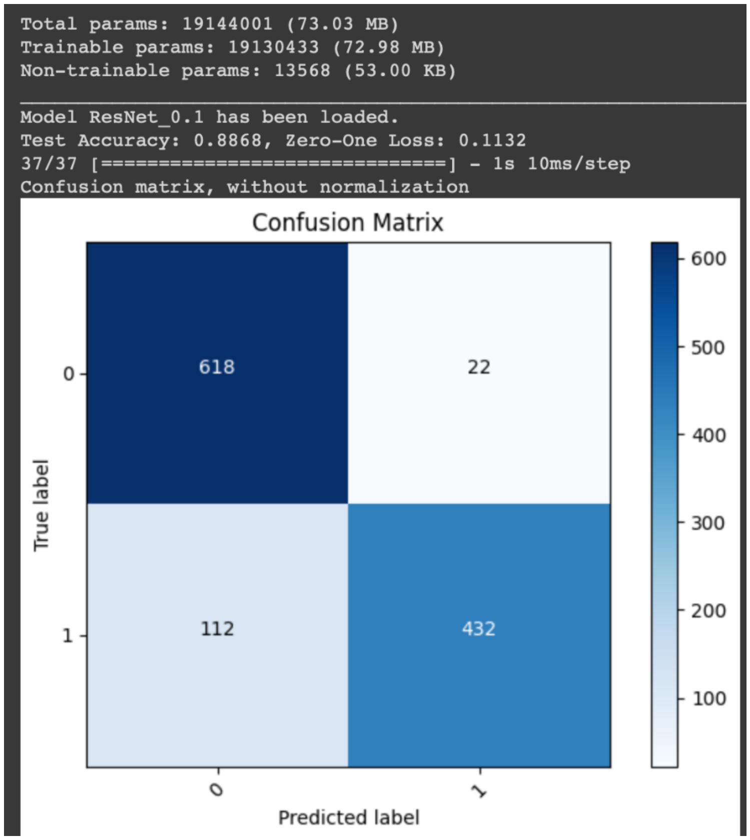


ResNet

- Optimizer: SGD with a learning rate of 0.1
- Test Accuracy: 0.8868
- Zero-One Loss: 0.1132

Using a learning rate of 0.1, ResNet achieved an accuracy of 88.68%. It correctly classified 618 Chihuahuas and 432 Muffins, with 22 false positives and 112 false negatives. This indicates the model is more conservative, leading to fewer false positives but a higher number of false negatives.

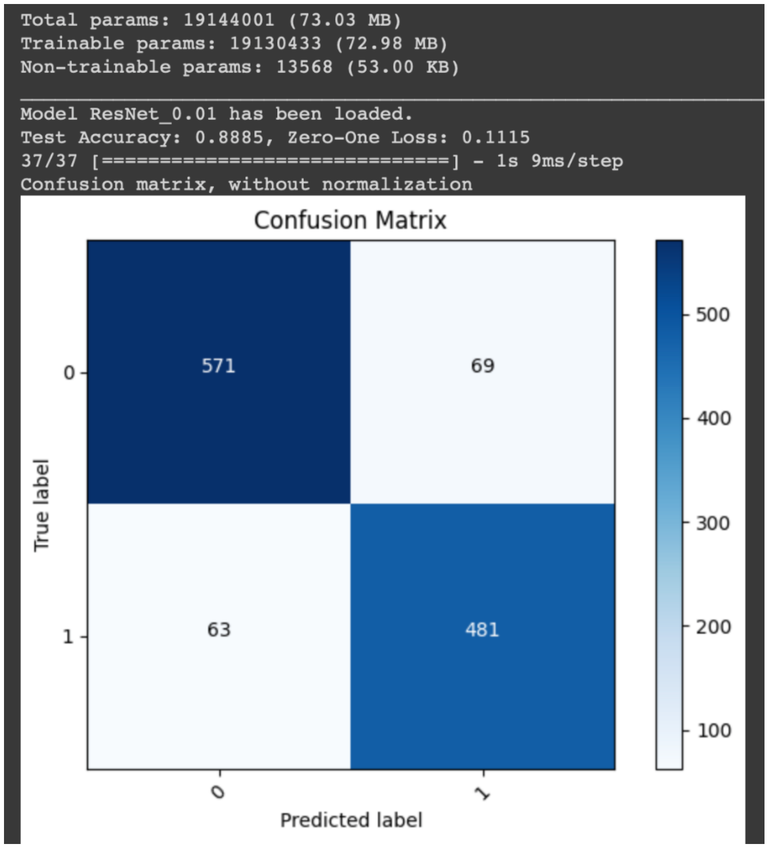
Confusion Matrix:



- Optimizer: SGD with a learning rate of 0.01
- Test Accuracy: 0.8885
- Zero-One Loss: 0.1115

At a learning rate of 0.01, the ResNet model achieved an accuracy of 88.85%. It correctly classified 571 Chihuahuas and 481 Muffins, with 69 false positives and 63 false negatives. This more even balance between false positives and false negatives suggests improved performance.

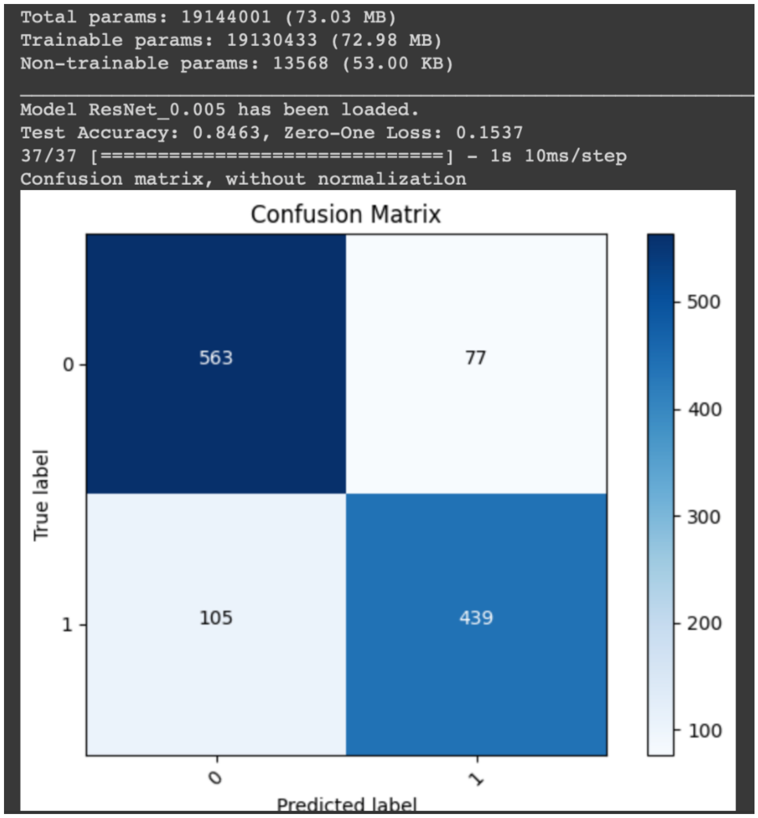
Confusion Matrix:



- Optimizer: SGD with a learning rate of 0.005
- Test Accuracy: 0.8463
- Zero-One Loss: 0.1537

The model with a learning rate of 0.005 showed the lowest accuracy of 84.63%. It correctly clasified 563 Chihuahuas and 439 Muffins, with 77 false positives and 105 false negatives. The increase in both false positives and false negatives indicates potential overfitting or an inappropriate learning rate.

Confusion Matrix:



Summary and Comparison

Comparing the results of the three models:
- Accuracy: VGG16 achieved the highest test accuracy at 91.72%, followed by ResNet at 88.85%, and AlexNet at 87.08%.
- Zero-One Loss: VGG16 also had the lowest zero-one loss at 0.0828, indicating fewer misclassifications compared to ResNet (0.1115) and AlexNet (0.1292).
- Confusion Matrix Analysis:
  - VGG16 showed the best performance in minimizing false negatives (46) and maintaining a high number of true positives (594).
  - AlexNet had a balanced performance but slightly higher false negatives (99) compared to VGG16.
  - ResNet, while still performing well, had the highest number of false positives (112) and the lowest true positives (571).

Conclusion

The VGG16 architecture, optimized with SGD at a learning rate of 0.01, outperformed both AlexNet and ResNet in classifying muffins and chihuahuas. This project demonstrates the importance of selecting the right model architecture and hyperparameters for achieving optimal performance in image classification tasks.

References

1. Kaggle Dataset: Muffin vs. Chihuahua Image Classification
   https://www.kaggle.com/datasets/samuelcortinhas/muffin-vs-chihuahua-image-classification
2. TensorFlow Keras Documentation: TensorFlow Keras
   https://www.tensorflow.org/api_docs/python/tf/keras