Analysis of Convolutional Neural Networks for Dog vs. Cat Classification

Yunzhen Wu

Date: Mar. 2, 2025

Management/Research Question

The goal of this task is to properly identify photos of dogs and cats utilizing deep understanding models. This capability is useful in various real-world applications, such as e-commerce, pet adoption platforms, and veterinary care.

In online marketplaces and pet adoption sites, automated classification helps categorize pets correctly, making it easier for users to find what they are looking for. In veterinary medicine, identifying breeds from images can provide quick insights into potential health risks associated with specific breeds. Additionally, smart home security systems can use similar technology to distinguish between pets and intruders, reducing false alarms.

Beyond classifying pets, the same deep learning techniques can be applied to other areas, such as wildlife monitoring, livestock management, and even medical imaging. Developing accurate models for dog vs. cat classification is a step toward improving broader image recognition tasks.

Introduction

Image classification is an essential aspect of computer vision that has applications for autonomous systems, medical imaging and even e-commerce. This study is targeted to evaluate the performance of three different Convolutional Neural Networks (CNNs) in classifying images of dogs and cats. The models implemented include a Simple CNN, a Deeper CNN, and a VGG16 Transfer Learning model. The Simple CNN consists of a basic architecture with convolutional and pooling layers. The Deeper CNN incorporates additional convolutional layers, batch normalization, and dropout for improved generalization. The

VGG16 Transfer Learning model leverages a pre-trained VGG16 network with fine-tuning tailored for dog vs. cat classification. This study evaluates model performance using training and validation accuracy, loss curves, confusion matrices, and ROC curves.

Data Presentation

The dataset in this study includes labeled images of dogs and cats. The images are resized to 150x150 pixels for processing. The training set contains 25,000 images (12,500 per class). The test set, containing unlabeled images, is used for final predictions and submission to Kaggle. From the figure 1 below, it shows that the training dataset is balanced, containing an equal number of cat and dog images, which allows effective training.

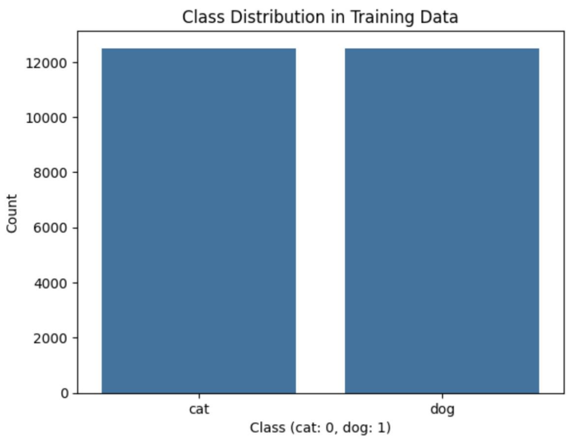

Figure 1: Class Distribution in Training Data

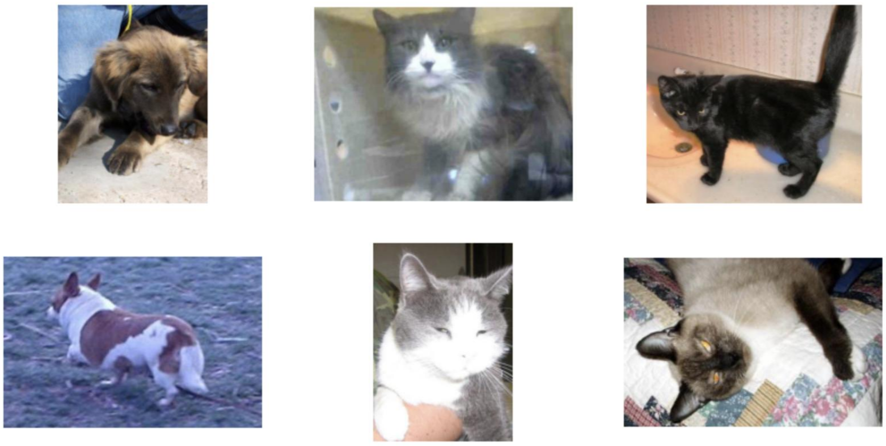

Figure 2: Example images of cats and dogs from the dataset.

Preprocessing

In order to prepare the data to be used for training models, a number of processing steps were performed. The data was then divided into validation and training sets with an 80/20 split. This ensures that the models can learn effectively while also being evaluated on unseen data.

Each image was resized to 150 * 150 pixels to maintain consistency in input dimensions. To enhance model generalization, pixel values were rescaled to the range [0,1] by dividing by 255. Additionally, horizontal flipping was applied to the training set, which helps introduce variability and reduce overfitting.

The processed data was then loaded using ImageDataGenerator and flow_from_dataframe, allowing for efficient batch loading and real-time preprocessing. The training dataset has 20,000 images, while the validation set contains 5,000 images. These preprocessing techniques were designed to optimize model training while ensuring stability and consistency across different datasets.

Methodology

Simple CNN:

The Simple CNN contains two convolutional layers complied with max-pooling layers and fully connected layers. The simplicity of this model allows for quick experimentation and provides insight into how well a basic CNN can perform on the dataset without advanced regularization or pretraining. It was chosen to establish a performance baseline and understand the impact of deeper architectures. The model was trained with the Adam optimizer, with a learning rate of 0.0001 and a dropout of 0.6 to avoid extreme overfitting.

Deeper CNN:

The Deeper CNN was selected to explore the benefits of deeper architectures with additional convolutional layers and batch normalization. These features help improve generalization and prevent overfitting, enhancing model performance beyond the simple CNN. The model was trained with the Adam optimizer, with a learning rate of 0.0001and a dropout of 0.5 to avoid overfitting.

VGG16 Transfer Learning:

The VGG16 model, introduced by Simonyan and Zisserman (2015) is a deep convolutional neural network that accomplished significant efficiency renovations in image classification tasks. The architecture consists of 16 weight layers, including 13 convolutional layers and 3 fully connected layers, utilizing small 3x3 convolution filters throughout. This design highlights depth and simplicity, allowing the model to find out complex attributes from large datasets like ImageNet. The success of VGG16 has led to its widespread adoption and adaptation in various computer vision applications.

By fine-tuning this architecture for dog vs. cat classification, we aimed to take advantage of transfer learning's ability to generalize better on smaller datasets with fewer training epochs. The model was trained with the Adam optimizer, with a learning rate of 0.0001.

All models were trained for 10 epochs with a batch size of 32.

Results & Evaluation

The performance of each model was analyzed using several metrics, including loss curves, accuracy curves, confusion matrices, and ROC curves.

The Simple CNN achieved a training accuracy of 86.9% and a validation accuracy of 79.3%. However, a significant gap between training and validation accuracy shows that the model overfit the training data.

The Deeper CNN demonstrated improved generalization, achieving a training accuracy of 87.8% and a validation accuracy of 84.8%. The additional layers and regularization techniques added to better efficiency compared to the Simple CNN.

The VGG16 Transfer Learning model outperformed the other two, achieving a training accuracy of 96.4% and a validation accuracy of 92.2%. Using pre-trained features from VGG16 significantly enhanced classification performance, enabling the model to generalize better to unseen data.

To further evaluate model performance, confusion matrices (Figure 4, 7, 10) were examined, highlighting classification errors. The ROC curve (Figure 5, 8, 11) analysis indicated that VGG16 achieved the highest area under the curve (AUC), reinforcing its superior classification ability.

After comparing the models, the Deeper CNN and VGG16 Transfer Learning models were selected for Kaggle submission due to their better validation performance. The final Kaggle scores were 0.43822 for the Deeper CNN and 0.24155 for the VGG16 Transfer Learning model, confirming that transfer learning provided the best results.

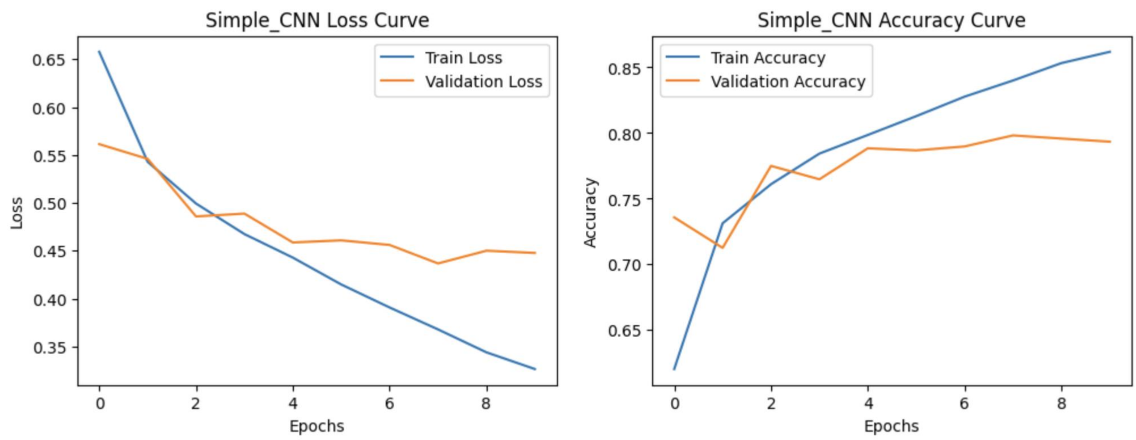

Figure 3: Training loss and accuracy curves for the Simple CNN model.

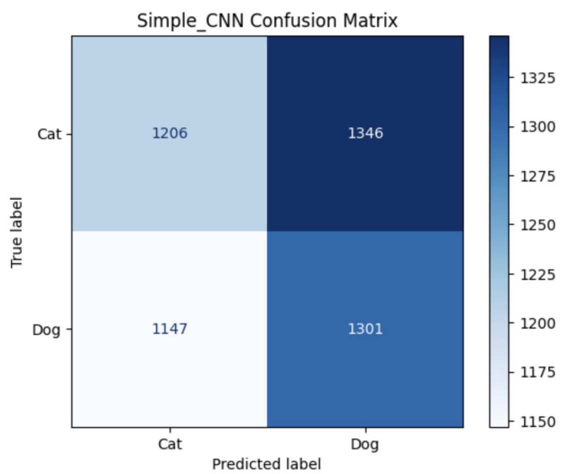

Figure 4: Confusion matrix for the Simple CNN model.

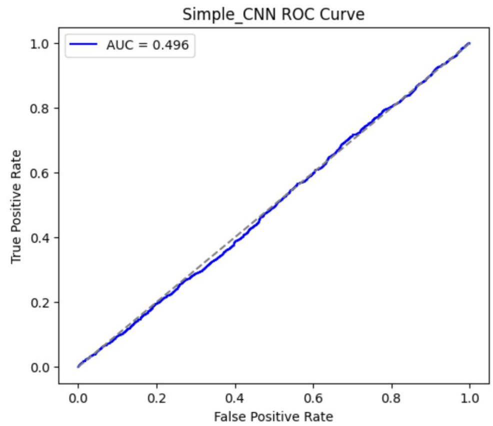

Figure 5: ROC curve for the Simple CNN model.

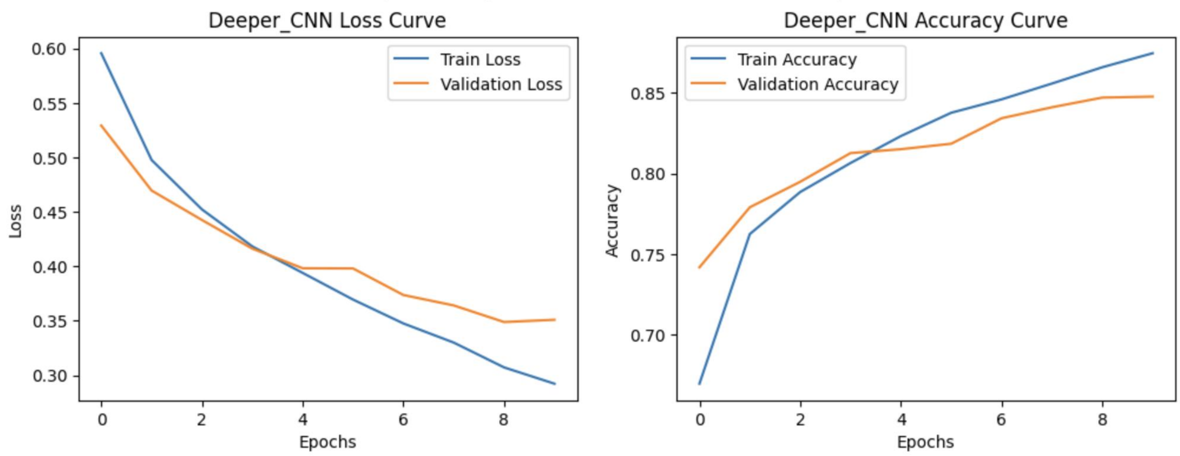

Figure 6: Training loss and accuracy curves for the Deeper CNN model.

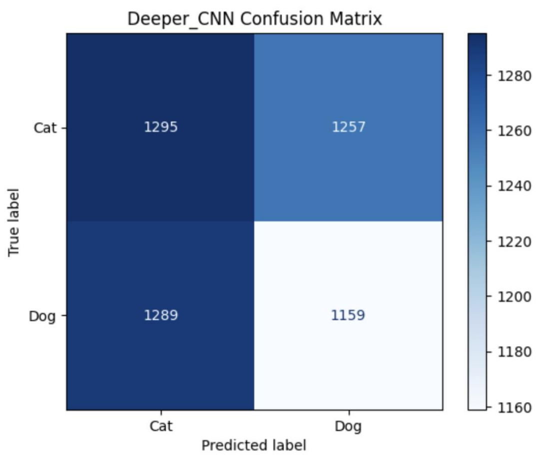

Figure 7: Confusion matrix for the Deeper CNN model.

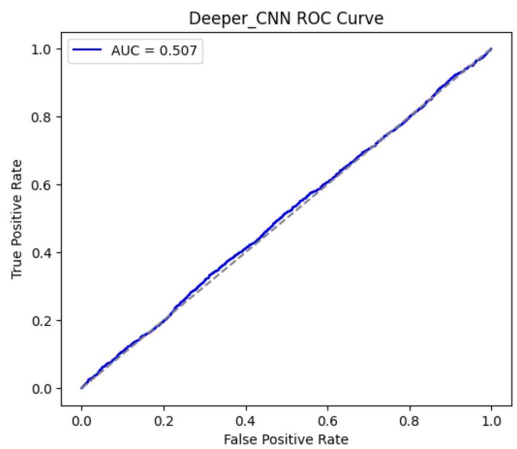

Figure 8: ROC curve for the Deeper CNN model.

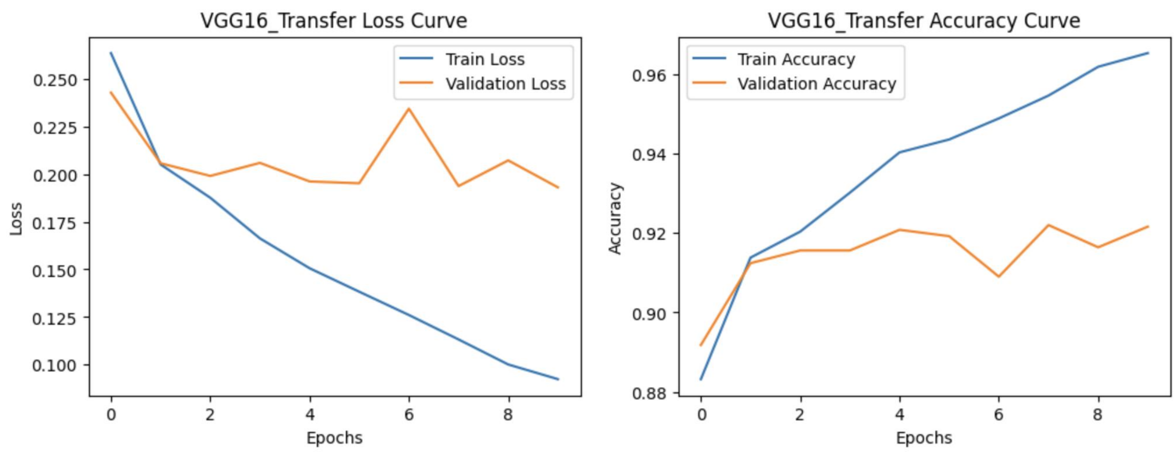

Figure 9: Training loss and accuracy curves for the VGG16 Transfer Learning model.

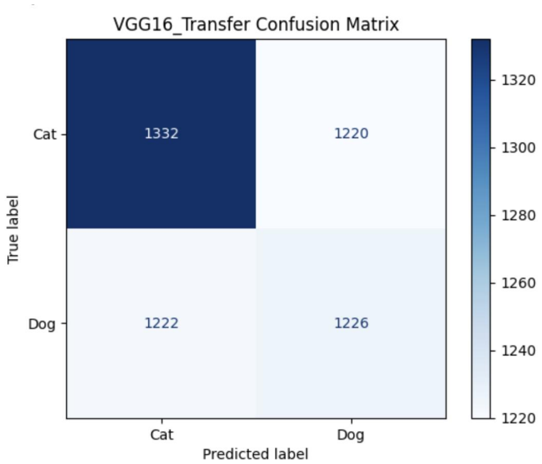

Figure 10: Confusion matrix for the VGG16 Transfer Learning model.

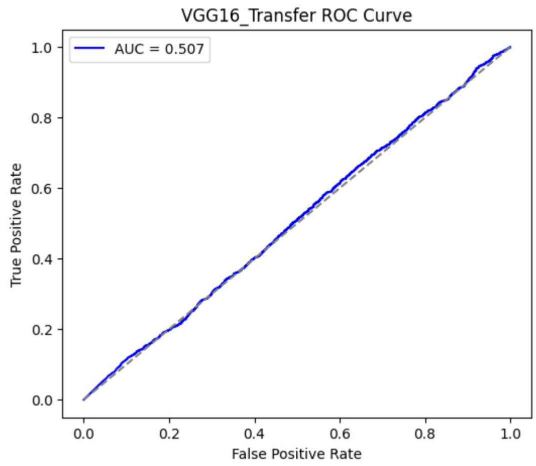

Figure 11: ROC curve for the VGG16 Transfer Learning model.

Conclusion

This study demonstrated that increasing model complexity and leveraging transfer learning can significantly improve classification performance. The Simple CNN, despite being a functional baseline, exhibited overfitting and suboptimal generalization. The Deeper CNN provided noticeable improvements, benefiting from additional layers and regularization techniques. However, the VGG16 Transfer Learning model emerged as the best-performing architecture, achieving the highest validation accuracy and Kaggle score.

Future work may focus on further fine-tuning the VGG16 model by allowing more layers to be trainable, testing different optimization strategies such as SGD with momentum, and increasing dataset size to improve generalization. The results underscore the effectiveness of CNNs for image classification and highlight the advantages of transfer learning in enhancing model performance for real-world applications.

References

Cukierski, W. (2016). "Dogs vs. Cats Redux: Kernels Edition." *Kaggle*. Available: https://kaggle.com/competitions/dogs-vs-cats-redux-kernels-edition.

Simonyan, K., and Andrew Zisserman. (2015). "Very Deep Convolutional Networks for Large-Scale Image Recognition." *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1409.1556.