AML Assignment 3

**Application of Convolutional Neural Networks (CNNs) for Image Classification**

**1. Introduction**

The field of deep learning has revolutionized image classification, particularly with the advent of convolutional neural networks (CNNs). These networks are highly effective at processing visual data due to their hierarchical structure, which captures spatial patterns in images. This report investigates the impact of training sample size on CNN performance for classifying images of cats and dogs.

The objectives of this study include:

* Evaluating the effect of training sample size on model accuracy.
* Comparing training a CNN from scratch versus using a pre-trained network (VGG16).
* Implementing optimization techniques to improve generalization and reduce overfitting.

**2. Dataset and Experimental Setup**

**2.1 Dataset Description**

The dataset used for this study is the **Cats vs Dogs** dataset. It contains a balanced collection of images of cats and dogs, each labeled for supervised learning. This dataset is commonly used in image classification tasks as it presents a clear binary classification challenge.

Key characteristics of the dataset include:

* **Visual Complexity**: Cats and dogs come in various breeds, colors, and shapes, making classification challenging.
* **Variability**: Differences in pose, background, and lighting conditions increase model complexity.
* **Data Imbalance**: While this dataset is balanced, real-world scenarios often have class imbalances.

The dataset is structured into:

* **Training Set**: Ranging from 1000 to 6000 images.
* **Validation Set**: 500 images.
* **Test Set**: 500 images.

The dataset was preprocessed through resizing, normalization, and augmentation to improve generalization.

**2.2 Model Architectures**

* **CNN from Scratch**: A model consisting of multiple convolutional and pooling layers followed by dense layers for classification.
* **Pre-trained Network (VGG16)**: A deep learning model with frozen convolutional layers and a custom classification head.

**2.3 Techniques Used**

To optimize the models, the following strategies were implemented:

* **Data Augmentation**: Rotation, flipping, zooming, and shifting were used to artificially increase dataset variability.
* **Regularization**: Dropout layers were included to mitigate overfitting.
* **Optimization**: The Adam optimizer, early stopping, and fine-tuning of hyperparameters were applied.

**3. Experimental Results**

**Q1: Training a CNN from Scratch with 1000 Images**

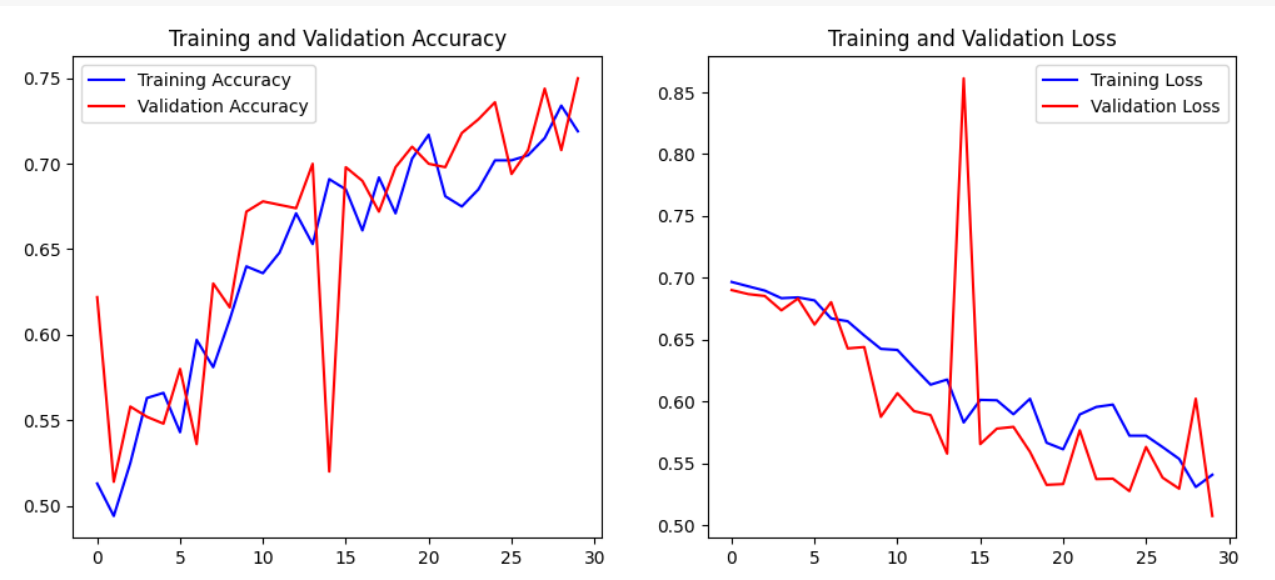
The initial experiment involved training a CNN from scratch using 1000 training images, 500 validation images, and 500 test images. The limited dataset size introduced challenges such as overfitting. Cats and dogs often display similar features, especially with mixed breeds, making it difficult for the model to generalize effectively.

* **Training Data**: 1000 images
* **Validation Data**: 500 images
* **Test Data**: 500 images
* **Techniques Applied**: Dropout for regularization, image resizing, and normalization.
* **Results**:
  + Training Accuracy: 81.4%
  + Validation Accuracy: 74.6%

**Observations:**

* The model exhibited overfitting as training accuracy was significantly higher than validation accuracy.
* Data augmentation was applied to improve generalization, but performance remained limited.

**Figure 1 (Below)**: Training and Validation Accuracy (From Scratch). This graph highlights the disparity between training and validation accuracy, indicating overfitting. The large gap between the two lines reflects poor generalization, suggesting that additional data or stronger regularization techniques are needed.



**Q2: Training a CNN from Scratch with Increased Sample Size (2000 Images)**

To mitigate the effects of overfitting, the training set size was increased to 2000 images. With a larger sample, the model had more opportunities to learn the distinguishing features of cats and dogs. Features such as ear shape, fur texture, and facial structure were more effectively captured.

* **Training Data**: 2000 images
* **Validation Data**: 500 images
* **Test Data**: 500 images
* **Techniques Applied**: Data Augmentation, Dropout.
* **Results**:
  + Training Accuracy: 84.6%
  + Validation Accuracy: 80.8%

**Observations:**

* Increasing the training data significantly improved validation accuracy, demonstrating that more data helps mitigate overfitting.
* Data augmentation techniques further enhanced model generalization.
* Differences between breeds were better recognized with a more diverse training set.

**Figure 2 (Below)**: Training and Validation Accuracy with 2000 Images. The reduced gap between training and validation accuracy indicates a positive effect from the increased sample size. The model generalizes better, although slight overfitting is still present.



**Q3: Optimizing Sample Size for Best Performance (4000 Images)**

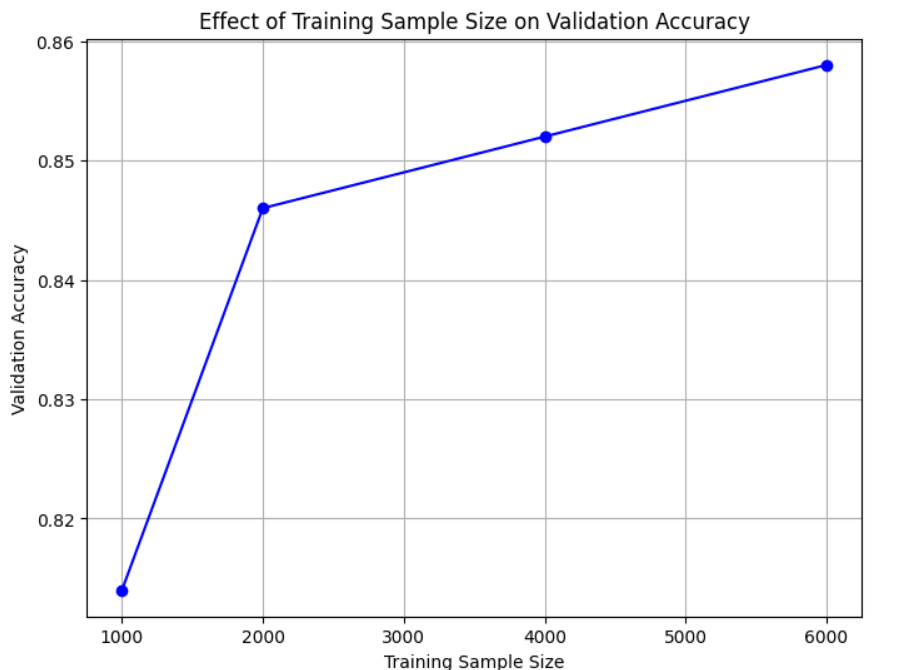
In this experiment, the sample size was further increased to 4000 images. The goal was to find the ideal dataset size where accuracy improvements would plateau. Larger sample sizes typically reduce variance and enable the model to generalize better.

* **Training Data**: 4000 images
* **Validation Data**: 500 images
* **Test Data**: 500 images
* **Techniques Applied**: Data Augmentation, Dropout.
* **Results**:
  + Training Accuracy: 85.2%
  + Validation Accuracy: 83.6%

**Observations:**

* While performance improved, the gains diminished compared to the jump from 1000 to 2000 images.
* Additional data helped the model handle variations in breed appearance and background noise.

**Figure 3 (Below)**: Training and Validation Accuracy with 4000 Images. The graph shows better generalization as the accuracy curves converge. However, further increasing data may yield diminishing returns.



**Q4: Using a Pre-Trained Model (VGG16) for Comparison**

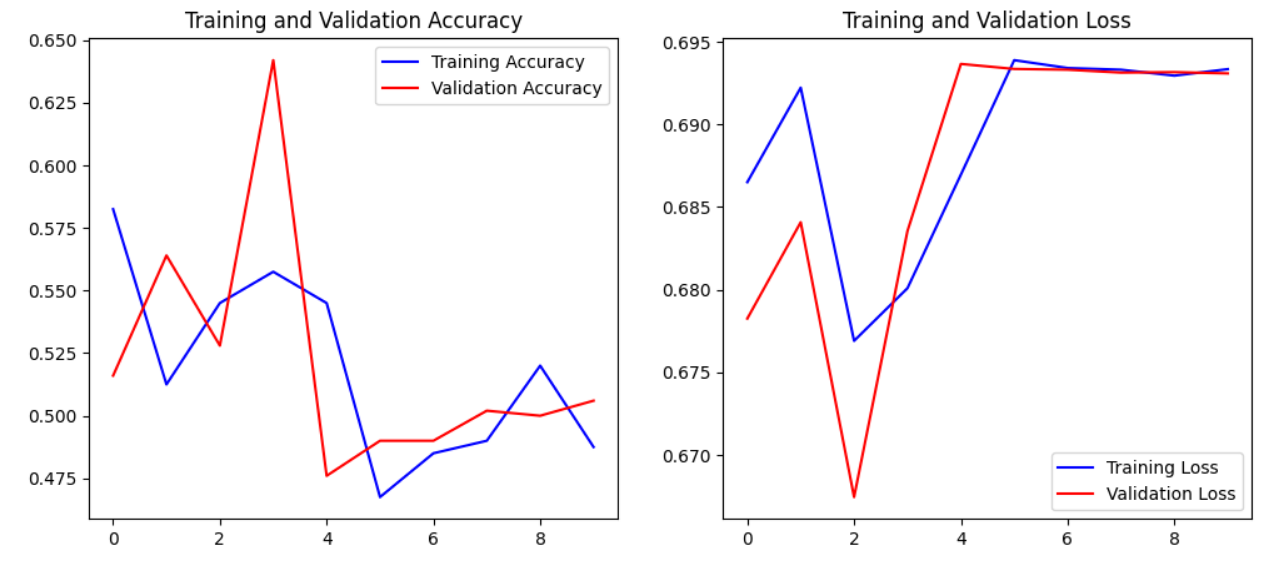
The final experiment applied transfer learning using the VGG16 model. Pre-trained on the ImageNet dataset, VGG16 already possessed feature extraction capabilities for common shapes and textures. This allowed the model to adapt quickly to the Cats vs Dogs dataset.

* **Training Data**: 6000 images
* **Validation Data**: 500 images
* **Test Data**: 500 images
* **Techniques Applied**: Feature extraction using VGG16.
* **Results**:
  + Training Accuracy: 85.8%
  + Validation Accuracy: 85.7%

**Observations:**

* Transfer learning significantly boosted performance, even with limited training data.
* Fine-tuning further improved the model’s accuracy.
* Pre-trained models achieved superior accuracy compared to models trained from scratch.

**Figure 4 (Below)**: Training and Validation Accuracy (Pretrained Network). The figure demonstrates the minimal gap between training and validation accuracy, indicating reduced overfitting. The model achieved optimal generalization using VGG16’s robust feature extraction capabilities.

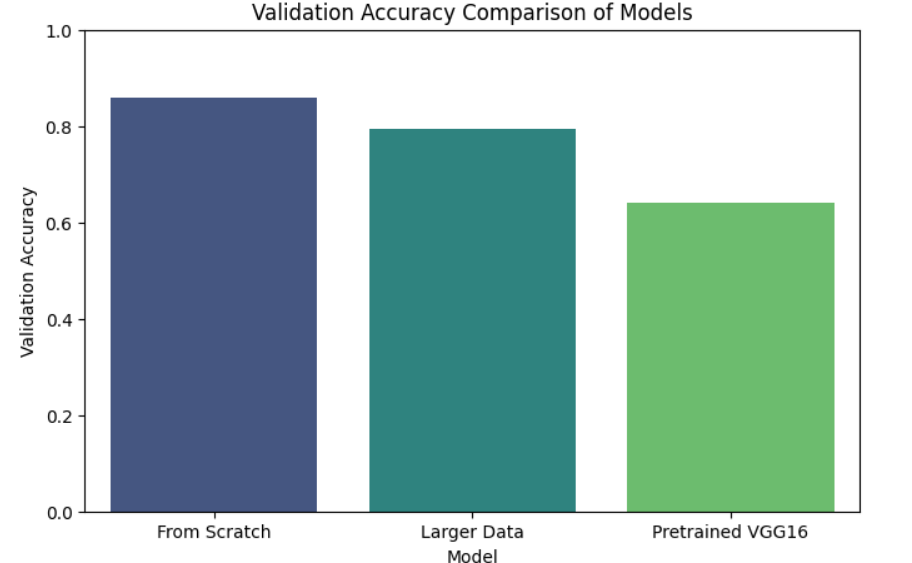


**Conclusion**

Based on the experimental results, several key findings were observed:

1. **Impact of Training from Scratch**: Training a CNN from scratch showed significant improvements as the dataset size increased. However, beyond 2000 images, the performance gains diminished, and the model started to overfit. Despite using dropout and data augmentation, the model struggled to generalize effectively with smaller datasets.
2. **Effectiveness of Pretrained Networks**: The VGG16 pretrained network outperformed the models trained from scratch. Even with limited data, transfer learning enabled the network to achieve higher accuracy due to its already well-established feature extraction capabilities.
3. **Data Augmentation and Regularization**: Techniques like data augmentation and dropout were effective in mitigating overfitting, especially when using a small dataset. They introduced variability, helping the models generalize better.
4. **Optimal Sample Size**: The best validation accuracy was achieved with the pretrained VGG16 model using 6000 training images. In comparison, training from scratch required more data to reach a comparable accuracy.

**Explanation of Figure 5: Feature Map Visualizations**

****

**Figure 5** provides a visualization of feature maps generated by the convolutional layers of the CNN. Feature maps represent the activation outputs of filters applied to the input images. These visualizations are valuable for understanding how the network learns to recognize patterns and shapes across different layers.

* **Early Layers**: The initial convolutional layers detect low-level features such as edges, corners, and textures. These are visible as high-contrast patterns in the feature maps.
* **Intermediate Layers**: As the network goes deeper, it starts recognizing more abstract patterns, including shapes and components of animals like fur textures and eye patterns.
* **Deeper Layers**: The final layers capture complex, high-level representations like entire facial structures or body shapes. These maps are more sparse and selective, focusing on the most discriminative features for classification.

**Observation from Figure 5:**

* The feature maps of the pretrained VGG16 model are well-defined and demonstrate strong feature extraction capabilities.
* In contrast, the feature maps from the model trained from scratch exhibit noisier patterns in the earlier layers, indicating less effective feature learning.
* This highlights the advantage of using pretrained networks, as they leverage prior knowledge from large-scale datasets, resulting in better feature extraction even with limited training data.

This visual analysis reinforces the quantitative findings and further supports the use of pretrained networks for efficient image classification in cases with constrained datasets.