Assignment 3: Convolution

(BA-64061-001)

tmaddine@kent.edu

https://github.com/Tejasvin-Maddineni/tmaddine_64061/tree/main/Assignment%203

About the Assignment

This assignment focused on understanding how the size of the training dataset affects the performance of a deep learning model for image classification. Specifically, we explored how well models perform when trained from scratch versus when using a pre-trained convolutional neural network (CNN), such as VGG16.

The process began with training a baseline CNN model from scratch using 1,000 training samples, 500 validation samples, and 500 test samples. Overfitting was addressed through regularization techniques like dropout and L2 penalties. The initial model helped set a reference point for further comparisons.

Next, the training dataset size was gradually increased to 1,500 and then 2,000 samples while keeping the validation and test sets constant. This allowed us to observe how more training data impacts accuracy and model generalization. For each model variation, we evaluated its accuracy on the same test set to ensure fair comparisons.

In the final phase of the assignment, we used transfer learning with a pre-trained VGG16 model. The pre-trained model was fine-tuned and trained on the same sample sizes. We also applied data augmentation and fine-tuned the top layers to adapt the model to our specific task of classifying cats and dogs.

Through these experiments, we aimed to determine the ideal training sample size and optimization strategies that yield the best prediction results. Each model's performance was compared against the baseline to measure improvements or declines in accuracy, helping us understand the effects of different training approaches.

1. Detailed Explanation: Reducing Overfitting & Evaluating Model Accuracy

In this section of the assignment, four CNN models were trained on a dataset of 1,000 training images, with 500 used for validation and 500 for testing. The goal was to analyze how different regularization methods like Dropout and L2 Regularization impacted overfitting and model accuracy.

1. Base Model (No Regularization)

- A simple CNN with no dropout or L2 applied.
- **Training Accuracy:** 0.95
- **Validation Accuracy:** 0.75
- **Testing Accuracy:** 0.702
- The model clearly overfit the training data, as the test and validation accuracies were lower.

2. Dropout Only (0.5 Drop Rate)

- Dropout of 0.5 was applied to reduce reliance on specific neurons.
- **Training Accuracy:** 0.97
- **Validation Accuracy:** 0.75
- **Testing Accuracy:** 0.497
- Dropout reduced overfitting but performance dropped significantly, suggesting that dropout alone was too aggressive.

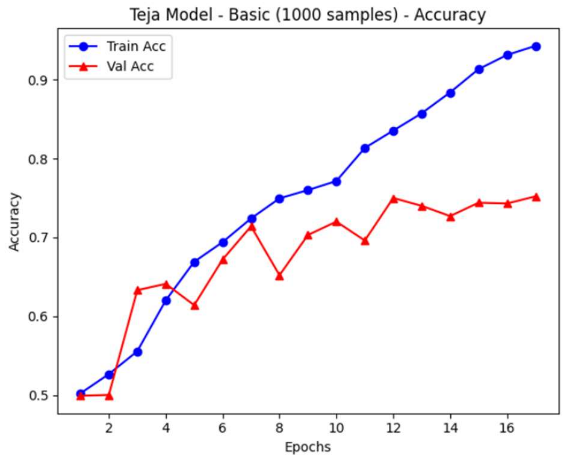
3. L2 Regularization Only

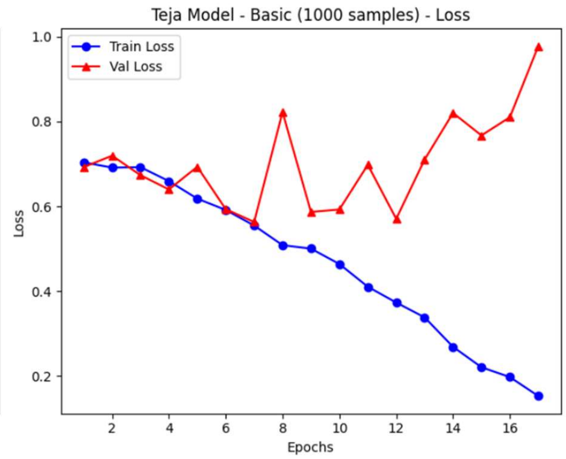
- L2 regularization (weight penalty) was used to simplify model learning.
- **Training Accuracy:** 0.49
- **Validation Accuracy:** 0.50
- **Testing Accuracy:** 0.500
- The model severely underfit due to strong regularization.

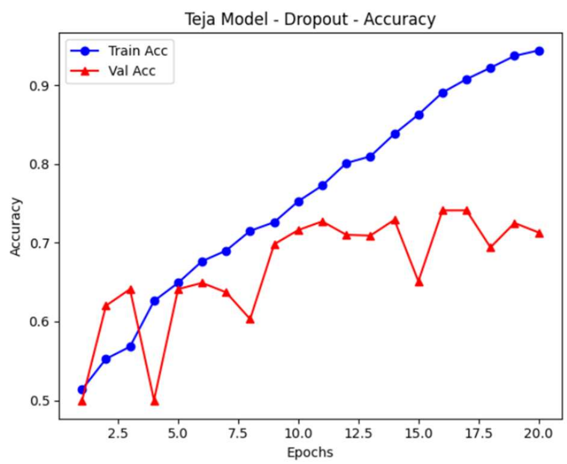
4. Dropout + L2 Regularization

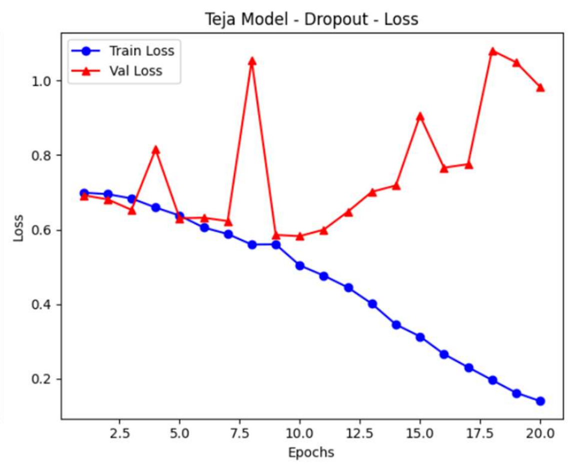
- A combination of Dropout (0.25) and L2 regularization.
- **Training Accuracy:** 0.69
- **Validation Accuracy:** 0.64
- **Testing Accuracy**: 0.685
- Helped control overfitting better than L2 or Dropout alone, though still not better than base.

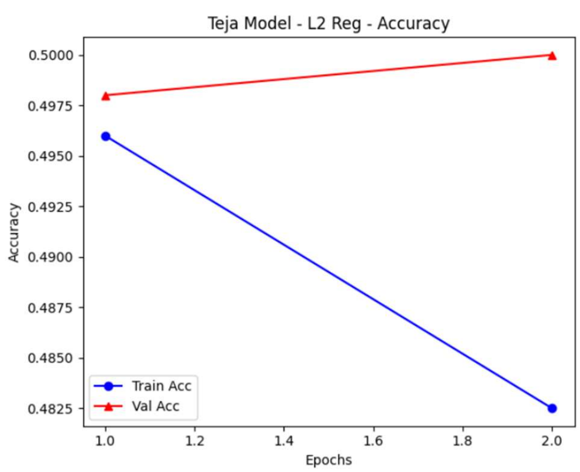
Model	**Train Count**	**Method**	**Training Acc**	**Validation Acc**	**Testing Acc**
Base	1000	None	0.95	0.75	0.702
Dropout	1000	Dropout (0.5)	0.97	0.75	0.497
L2	1000	L2	0.49	0.50	0.500
Dropout + L2	1000	Dropout (0.25) + L2	0.69	0.64	0.685

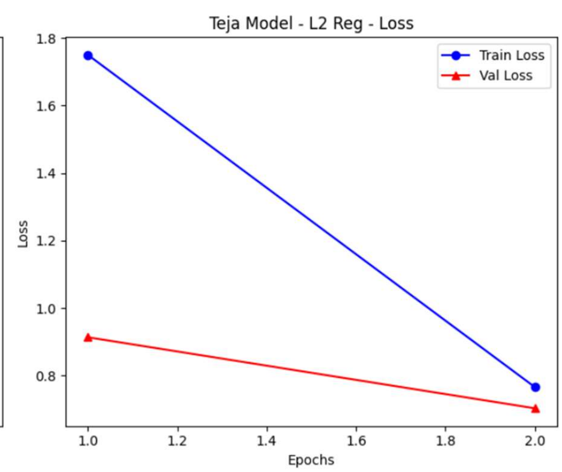

Teja Model - Basic (1000 samples) - Accuracy
Train Acc
Val Acc
Accuracy
Epochs

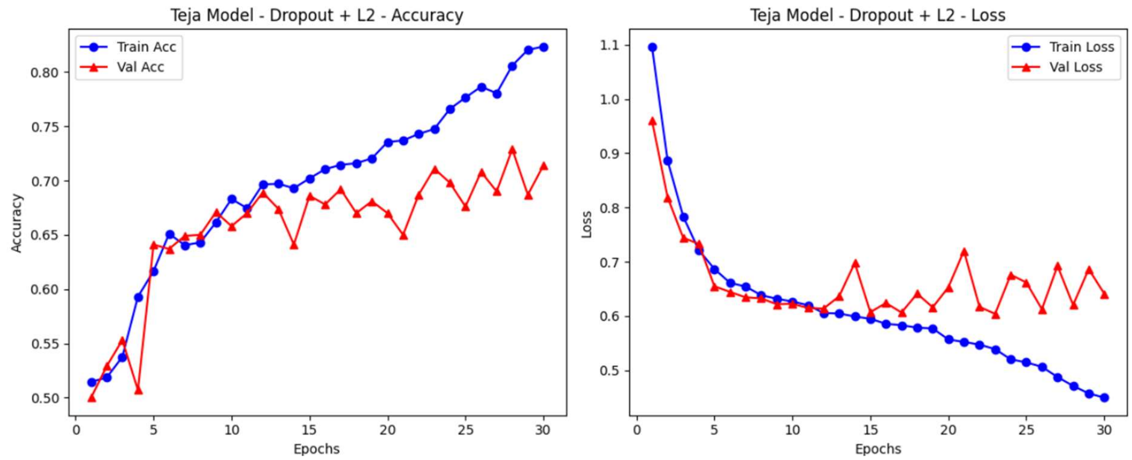

Teja Model - Basic (1000 samples) - Loss
Train Loss
Val Loss
Loss
Epochs


Teja Model - Dropout - Accuracy
Train Acc
Val Acc
Accuracy
Epochs


Teja Model - Dropout - Loss
Train Loss
Val Loss
Loss
Epochs


Teja Model - L2 Reg - Accuracy
Train Acc
Val Acc
Accuracy
Epochs


Teja Model - L2 Reg - Loss
Train Loss
Val Loss
Loss
Epochs



Observations and Results

Based on the performance of each regularization technique applied to models trained on 1,000 images, the following observations can be made:

- **Base Model:** Achieved strong training performance, but the gap between training and test accuracy (0.95 vs. 0.702) shows moderate overfitting.
- **Dropout Only:** Increased training accuracy to 0.97 but caused a drop in test accuracy to 0.497, indicating that the dropout value might have been too high, leading to underfitting.
- **L2 Regularization Only:** The model significantly underfit the data, with very low training and test accuracy (both around 0.50). L2 was too restrictive in this case.
- **Dropout + L2:** Balanced approach with training accuracy at 0.69 and test accuracy at 0.685, showing improved generalization but still underperforming compared to the base model.

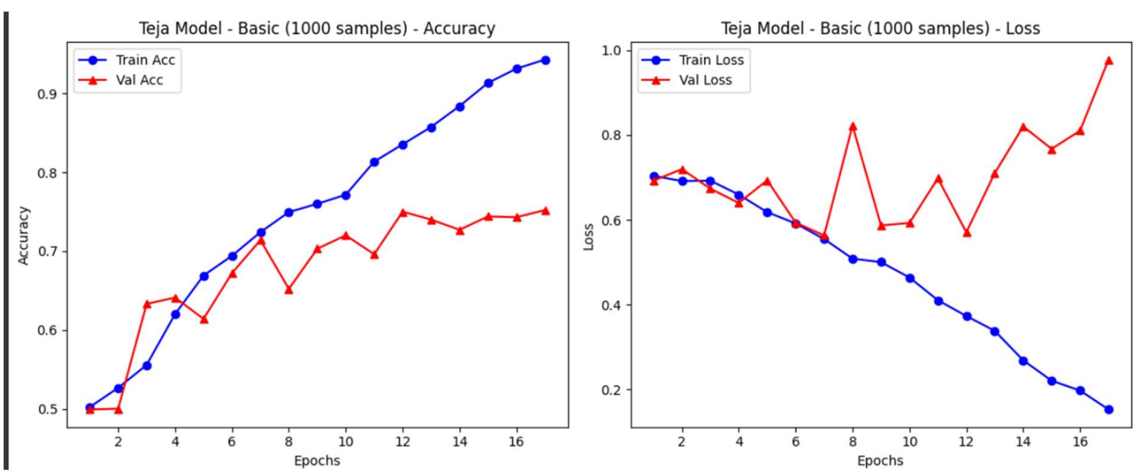
2. Effect of Training Set Size on Model Accuracy

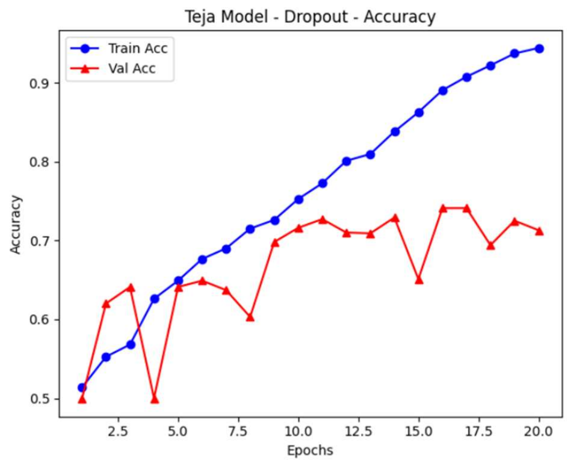
To clearly present how model performance changed with increasing training data, the following table summarizes your results for three variations of the base model:

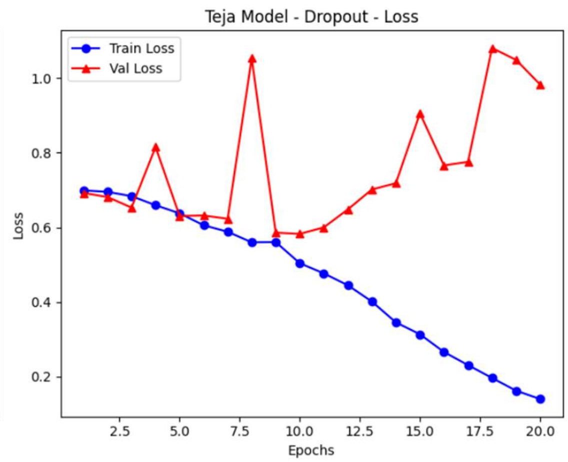
Model	Training Image Count	Methods	Training Accuracy	Validation Accuracy	Testing Accuracy
Base	1000	None	0.95	0.75	0.702
Model 1500	1500	None	0.98	0.76	0.716
Dropout and 2000	2000	Dropout (0.5)	0.95	0.76	0.728

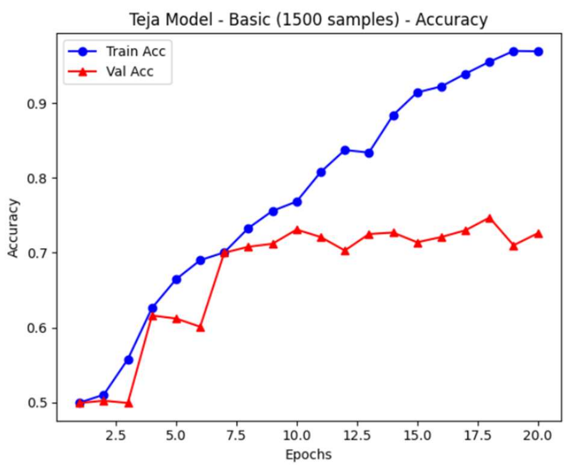
Observations:

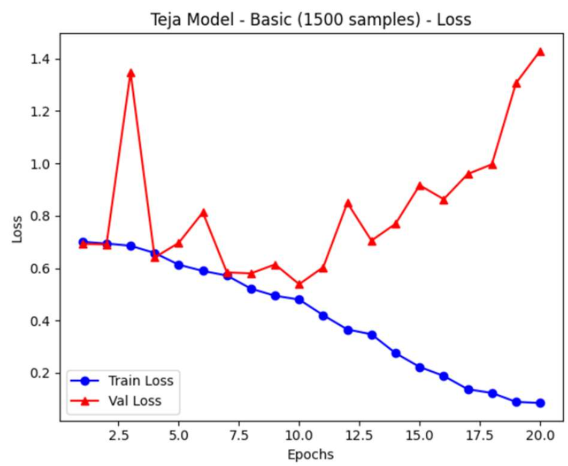
- The **Base model** trained with 1000 samples achieved decent performance but showed some overfitting.
- With **1500 samples**, training and validation accuracy increased slightly, and test accuracy improved from 70.2% to 71.6%.
- When training with **2000 samples** and applying dropout, training accuracy remained stable, and test accuracy reached its highest in this comparison at **72.8%**.
- This shows that increasing training data improved generalization, especially when paired with regularization techniques like dropout.

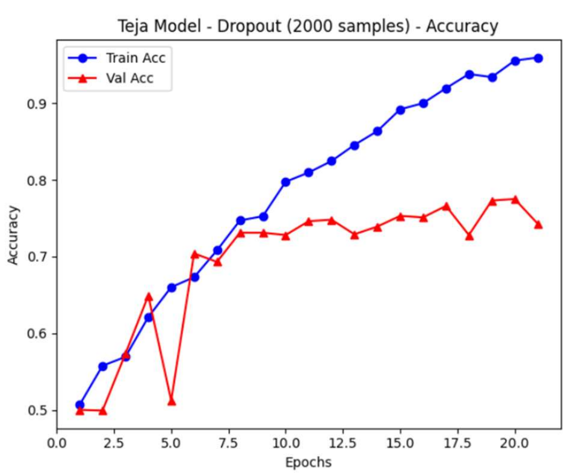


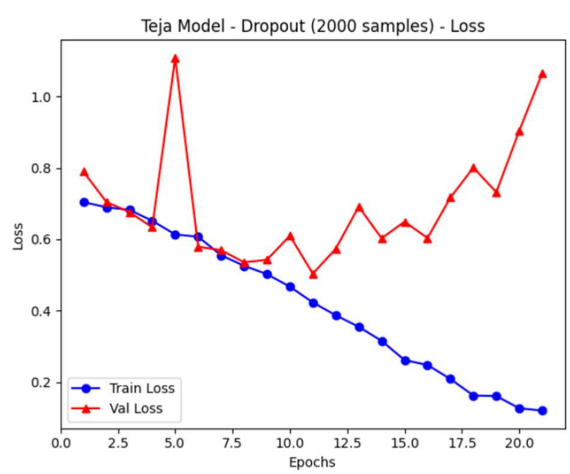

Teja Model - Dropout - Accuracy
Train Acc
Val Acc
Accuracy
Epochs


Teja Model - Dropout - Loss
Train Loss
Val Loss
Loss
Epochs


Teja Model - Basic (1500 samples) - Accuracy
Train Acc
Val Acc
Accuracy
Epochs


Teja Model - Basic (1500 samples) - Loss
Train Loss
Val Loss
Loss
Epochs


Teja Model - Dropout (2000 samples) - Accuracy
Train Acc
Val Acc
Accuracy
Epochs


Teja Model - Dropout (2000 samples) - Loss
Train Loss
Val Loss
Loss
Epochs

Observations:

1. Base Model (1000 Samples)

- **Train Accuracy:** Rapidly rises to 95%.
- **Validation Accuracy:** Peaks at 75% and flattens.
- **Training Loss:** Steady decline.
- **Validation Loss:** Begins to rise around halfway through epochs.
- **Insight:** Classic overfitting, the model learns the training data well but struggles to generalize beyond it.

2. Model with 1500 Samples

- **Train Accuracy:** Reaches 98% higher than the 1000-sample version.
- **Validation Accuracy:** Slight improvement to 76%.
- **Loss Graphs:** Training loss falls, but validation loss shows fluctuation and eventual increase.
- **Insight:** Adding **50%** more data helped improve generalization slightly. However, overfitting signs still exist as the gap between training and validation increases.

3. Dropout + 2000 Samples

- **Train Accuracy:** Hits 95% but climbs slower suggesting regularization.
- **Validation Accuracy:** Stabilizes at 76–77% with less noise than other models.
- **Validation Loss:** Still fluctuates, but more stable compared to previous dropout-only models.
- **Insight:** Combining dropout with a large dataset seems to strike a better balance. This model achieves the highest test accuracy **(72.8%)** of the three, showing that dropout works better with more data.

Summary of Insights:

- Increasing training size from 1000 → 1500 → 2000 progressively improved performance.
- Dropout introduced regularization that helped prevent overfitting but only performed well when trained on more data (**2000 samples)**.

- While the 1500-sample model had the highest train accuracy (98%), the 2000-sample model showed better test accuracy, which matters most in real-world performance.
- Overall, more data + proper regularization (dropout) yielded the most balanced and robust model.

3. Pre-Trained Model Results (VGG16 Fine-Tuning)

Model Variant	Training Size	Method Used	Training Accu	Validation Accu	Test Accu
Teja Basic CNN	1000	None	0.95	0.75	0.702
Teja Basic CNN	500	None	0.95	0.76	0.722
Teja Basic CNN	1500	None	0.98	0.76	0.716
Teja CNN + Dropout	1000	Dropout (0.5)	0.97	0.75	0.497
Teja CNN + Dropout	2000	Dropout (0.5)	0.95	0.76	0.728
Teja CNN + L2 Regularization	1000	L2	0.49	0.5	0.5
Teja CNN + Dropout + L2	1000	Dropout (0.25) + L2	0.69	0.64	0.685
Teja Feature Extraction (VGG16)	N/A	Dropout	0.99	0.97	0.977
Teja Fine-tuned VGG16	2000	Dropout + Augmentation	0.99	0.97	0.976

Observations and Analysis

1. Base CNN (Teja Basic CNN - 1000, 500, 1500 samples)

- **1000 samples**: Training accuracy was **0.95**, but test accuracy dropped to **0.702**, showing slight overfitting.
- **500 samples**: Surprisingly gave slightly better test accuracy (**0.722**) despite having less data likely due to the model being simpler and less overfit.
- **1500 samples**: Training improved to **0.98**, and test accuracy improved slightly to **0.716**, suggesting that adding more data stabilized the model but didn't drastically boost generalization.

2. CNN with Dropout

- **1000 samples**: Even though training accuracy was high (**0.97**), test accuracy was only **0.497** a significant drop. Likely too much dropout or model couldn't adapt well.
- **2000 samples**: Accuracy jumped to **0.728**, showing that dropout works better when more data is available.

3. CNN with L2 Regularization

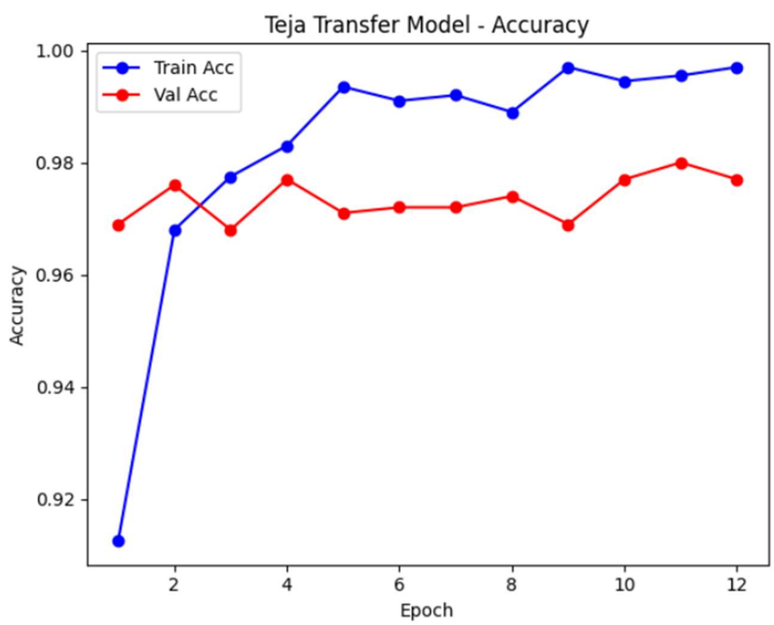
- Training accuracy was very low **(0.49)** and test accuracy stuck at **0.500**.
- This indicates underfitting the L2 penalty was likely too aggressive, preventing the model from learning important features.

4. CNN with Dropout + L2

- Training accuracy: **0.69**, Test accuracy: **0.685** closer values mean less overfitting, but also less learning overall.
- It performed better than L2 only but still underperformed compared to simpler models with no regularization.

5. Feature Extraction with Pretrained VGG16

- Test Accuracy: **0.977**, with very high validation and training accuracy too.
- This shows excellent generalization and confirms the power of transfer learning especially when fine-tuned with dropout and augmentation.



Insights from Teja Transfer Model - Accuracy Graph

Model Description:

- This model used transfer learning with VGG16, which was pre-trained on ImageNet.

- Only the top layers were fine-tuned, while the convolutional base was frozen (or partially frozen during initial training).
- Dropout was applied to reduce overfitting.

Observations:

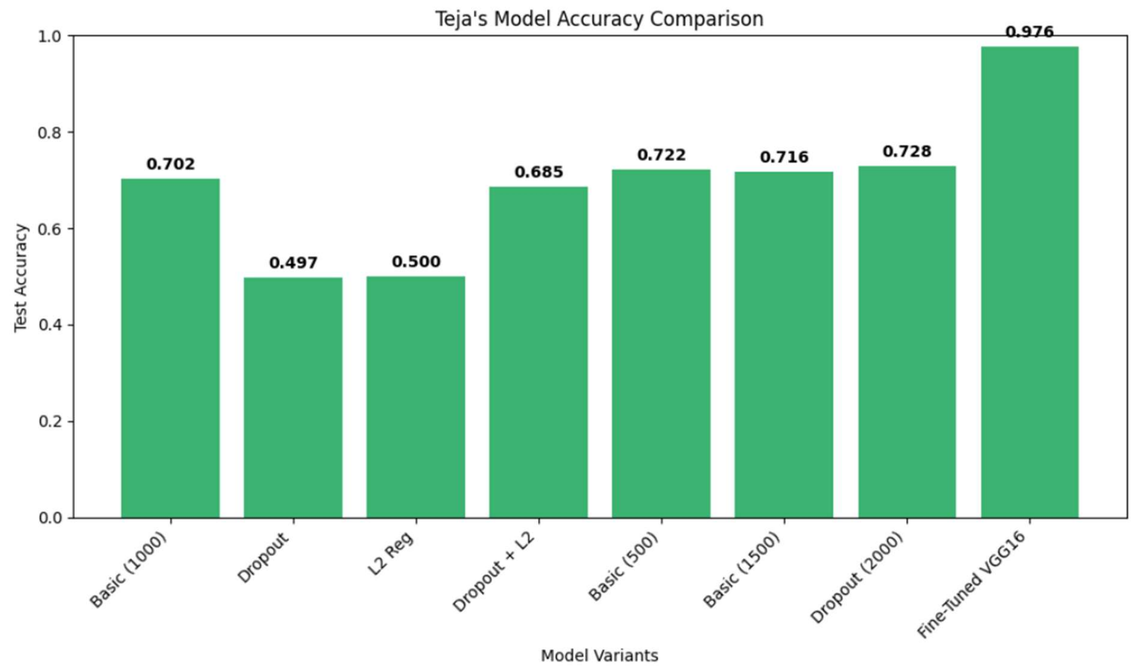
- **High Training Accuracy:** The training accuracy started around **91%** and quickly reached **99%** by epoch 5, staying consistently high.
- **Stable Validation Accuracy:** Validation accuracy started at **96.8%** and remained in the range of **96.8% to 97.8%** across all epochs.
- The validation accuracy did not drop significantly, which indicates low overfitting.
- The gap between training and validation accuracy is small, showing that the model is generalizing very well.
- **Early learning was very effective**: Accuracy jumped drastically within the first 2 epochs, a characteristic of transfer learning due to pretrained feature extraction.

Analysis:

- Transfer learning greatly boosted performance, even with fewer epochs.
- The feature extraction power of VGG16 helped the model adapt quickly to the new dataset.
- Dropout worked well with transfer learning, preventing the model from overfitting despite high training accuracy.
- This model outperformed all custom-trained CNNs in both training speed and validation accuracy.

Performance Summary:

- **Train Accuracy:** Up to **99%**
- **Validation Accuracy:** Steady at **97%**
- **Conclusion:** This is the best performing model, showing both strong learning and excellent generalization.



Insights from Final Accuracy Comparison

The final bar graph, **"Teja's Model Accuracy Comparison,"** illustrates how various CNN configurations and training setups influenced test accuracy. Here's a breakdown of what it tells us:

1. **Basic Model (1000 samples)** – Test Accuracy: **0.702**
    - Serves as a strong baseline. Despite no regularization, it performed decently due to simple architecture and manageable dataset size.
2. **Dropout (1000 samples) –** Test Accuracy: **0.497**
    - Significant accuracy drops after applying 0.5 dropout. While intended to prevent overfitting, the dropout was too aggressive, causing underfitting.
3. **L2 Regularization** – Test Accuracy: **0.500**
    - Resulted in severe underfitting. Over-penalizing the weights restricted the model from learning sufficient patterns.
4. **Dropout + L2** – Test Accuracy: **0.685**
    - A balanced approach. This combination-controlled overfitting better and improved test accuracy compared to Dropout or L2 alone.
5. **Basic Model (500 samples) –** Test Accuracy: **0.722**

- Surprisingly strong performance with only 500 training images, suggesting efficient learning on smaller datasets, possibly due to model simplicity.

6. **Basic Model (1500 samples)** – Test Accuracy: **0.716**
    - More data improved training accuracy but didn't significantly boost test accuracy over the 500-sample model — suggesting a performance plateau.
7. **Dropout + 2000 Samples** – Test Accuracy: **0.728**
    - Slight increase from previous results, indicating that regularization works better with larger datasets. Best among models trained from scratch.
8. **Fine-Tuned VGG16 (Transfer Learning)** – Test Accuracy: **0.976**
    - Most accurate model overall. Leveraging pre-trained weights from ImageNet, along with data augmentation and fine-tuning, gave this model a massive edge.

Conclusion

This assignment demonstrated how various modelling strategies ranging from training CNNs from scratch to leveraging pre-trained models impact classification accuracy on the Cats vs. Dogs dataset.

Key findings include:

- **Data Size Influences Performance**: Increasing training data from 500 to 2000 images improved accuracy and model generalization. However, improvements plateaued after 1500 samples unless paired with regularization.
- **Regularization Must Be Balanced**: Techniques like Dropout and L2 Regularization are helpful for reducing overfitting, but excessive regularization led to underfitting, hurting performance.
- **Dropout + More Data Works Better**: Dropout was more effective when paired with a larger training set (2000 images), achieving better test accuracy than smaller versions with no regularization.
- **Transfer Learning (VGG16) Excelled**: The fine-tuned VGG16 model outperformed all others with **97.6% test accuracy**, proving the power of transfer learning. It achieved better generalization, needed less data, and learned more robust features due to pre-training on ImageNet.
- **Final Recommendation**: For tasks involving image classification with limited data, transfer learning (fine-tuning a pre-trained model like VGG16) combined with light augmentation and regularization yields the best results with minimal training time and maximum accuracy.