VGG-16 Gender Classification

Name: Alper Çamlı

Date: May 6, 2025

Notebook link:
https://drive.google.com/file/d/1YQtea41Jd9IApPdjg6MKlvqj56cCLrgB/view?usp=sharing

1. Introduction

In this assignment, we explored transfer learning using the VGG-16 convolutional neural network to perform binary gender classification on a subset of the CelebA dataset. The main goal was to evaluate how different learning rates and fine-tuning strategies affect the model's performance. By utilizing a pretrained model, we aimed to leverage learned features from a large dataset and adapt them to our task using.

2. Method

We used a subset of the CelebA dataset (CelebA30k) containing 30,000 facial images annotated with multiple attributes, including the binary gender label under the "Male" column. The dataset was split into 80% training, 10% validation, and 10% testing sets.

We adopted the pretrained VGG-16 model and experimented with the following configurations:

- **Learning Rates**: 0.001 and 0.0001
- **Fine-Tuning Strategies**:
    1. **Frozen Convolutional Layers**: All convolutional layers frozen; only the classifier head was trained.
    2. **Fine-Tuning Last Convolutional Block**: The last convolutional block and the classifier head were trainable.

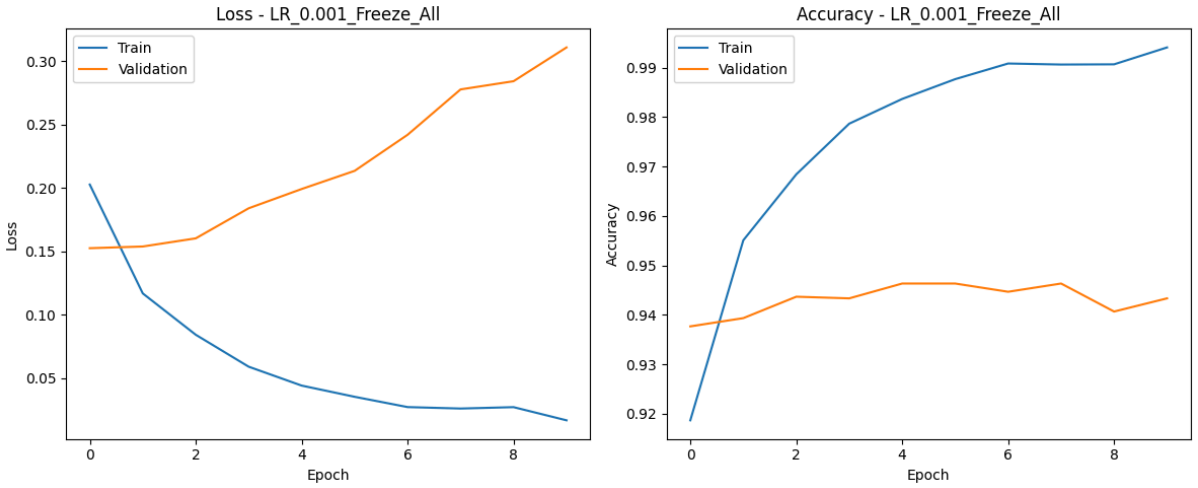
Each configuration was trained for **10 epochs**, and performance was measured using accuracy, F1 score, precision, recall, and confusion matrices. We also tracked loss/accuracy over epochs for both training and validation sets.

3. Results

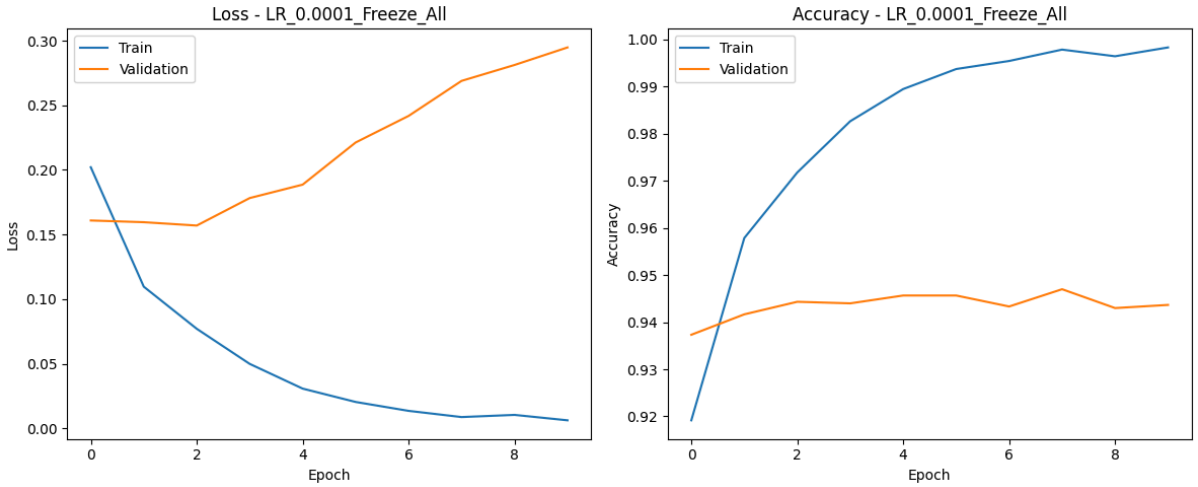
In this section performance training and test results are presented. Each model variant is evaluated based on their Accuracy and Loss over epochs for training and validation sets are compared via plot graphs. Additionally, Accuracy, F1 Score, Precision, Recall for Testing.

Loss/Accuracy vs. Epochs Graphs

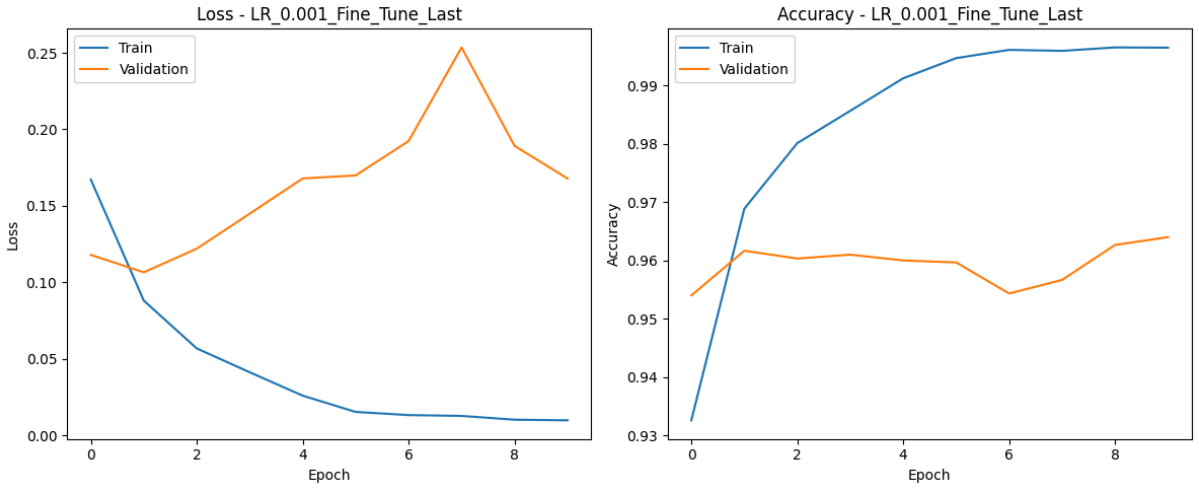
1. Here is a graph of Loss and Accuracy over epochs for a model by freezing all convolutional layers and training only the classifier head with **Learning Rate: 0.001. (Model 1)**



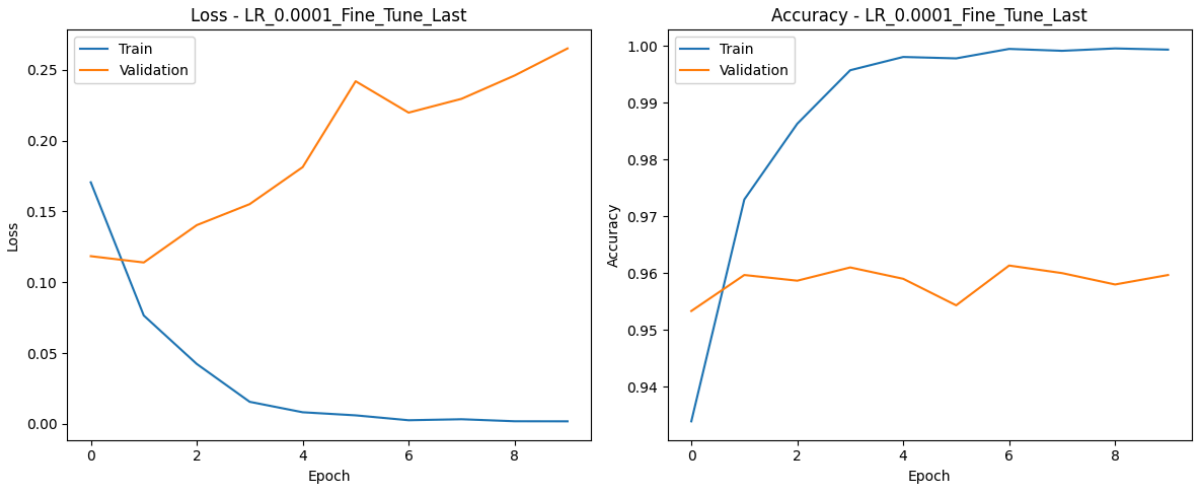
2. Here is a graph of Loss and Accuracy over epochs for a model by freezing all convolutional layers and training only the classifier head with **Learning Rate: 0.0001. (Model 2)**



3. Here is a graph of Loss and Accuracy over epochs for a model by freezing all weights, but **fine-tuning** the last convolutional block along with the classifier head. with **Learning Rate: 0.001. (Model 3)**



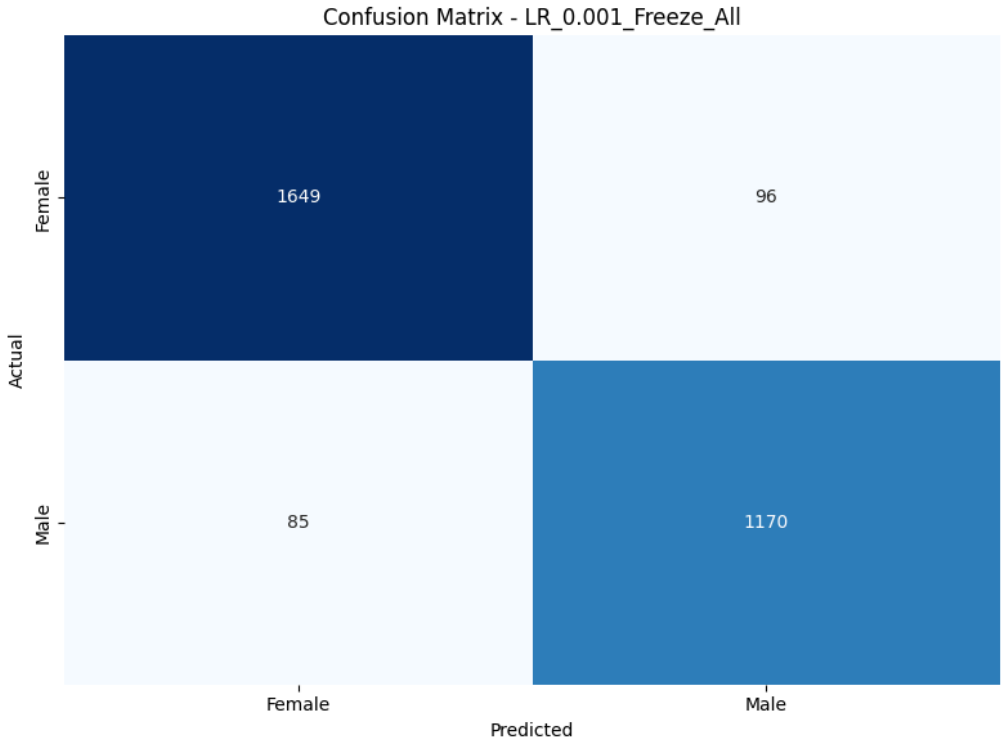
4. Here is a graph of Loss and Accuracy over epochs for a model by freezing all weights, but **fine-tuning** the last convolutional block along with the classifier head. with **Learning Rate: 0.0001. (Model 4)**



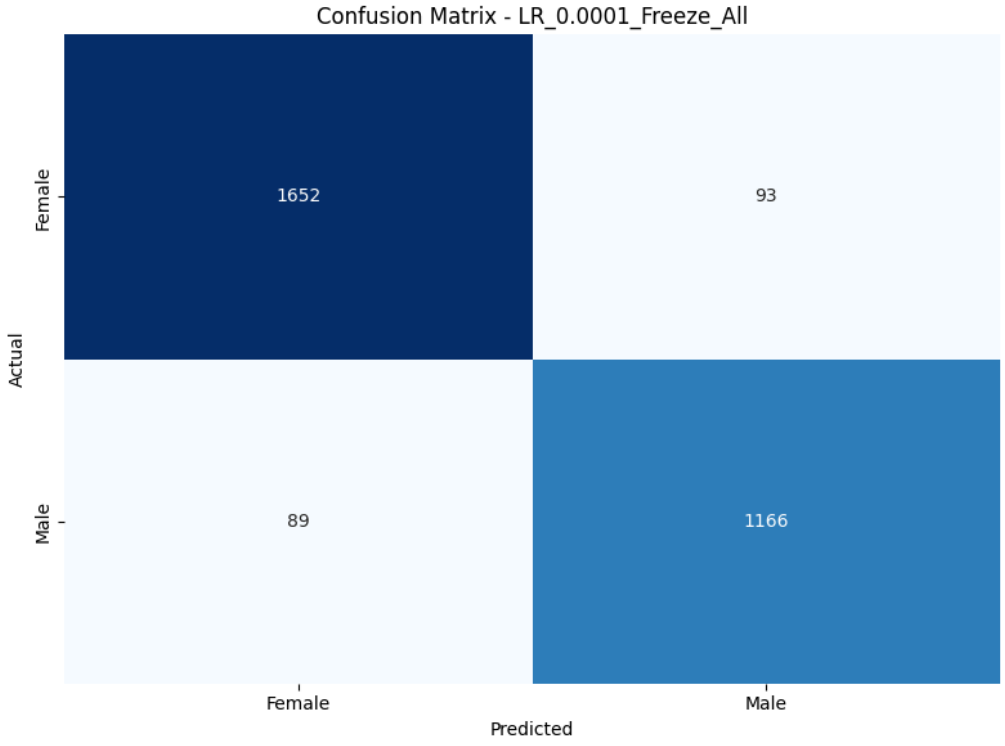
While the accuracy of the validation sets of the models without fine-tuning is around 0.94, the models with fine-tuning show accuracy around 0.96 with less loss. Additionally, changing the learning rate didn't make a major difference in Accuracy nor Loss, without fine-tuning. However in the fine-tuned model changing the learning rate creates a little deviation on the loss. Also, deviation of metrics throughout the epochs is higher on the fine-tuned models, compared to Model 1 and Model 2.

Confusion Matrices and Test Results

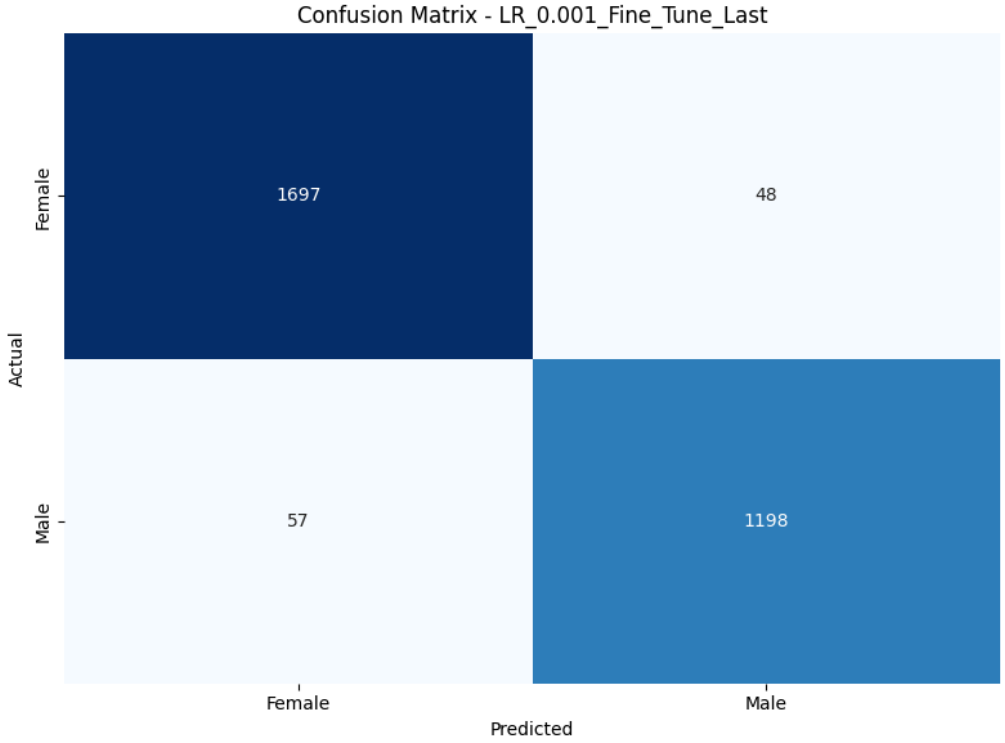
Model 1:



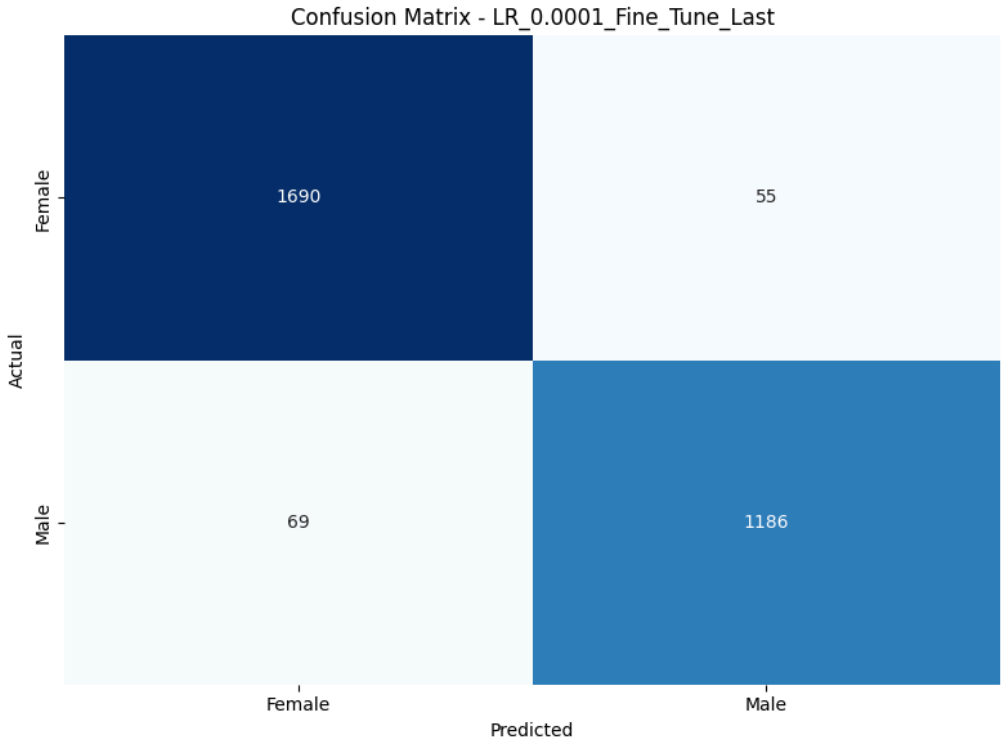
Model 2:



Model 3:

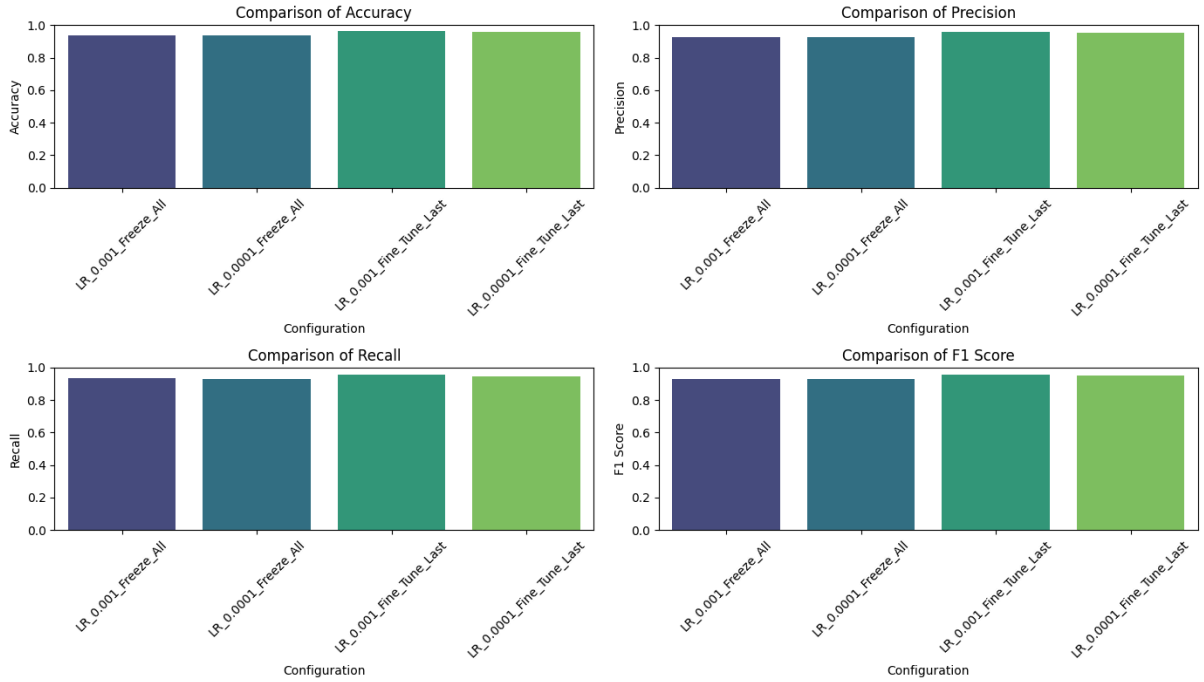


Model 4:



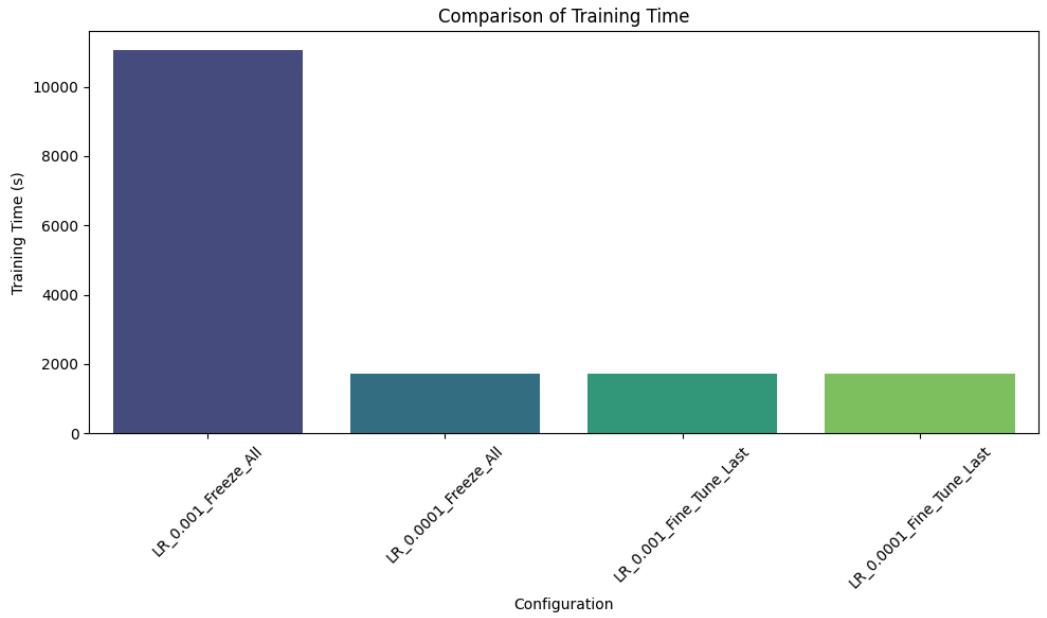
Testing is done with isolated 1745 Female and 1250 Male in total 3000 images which is 10% of the initial dataset. Confusion matrices show that fine-tuning resulted slightly better in accuracy. However all four models gave high accuracy in testing of 3000 images. Also we don't see any anomalies in Confusion matrices which means the models were successful in detecting both Male and Female.

Accuracy, Precision, Recall, F1 Score Comparison Graphs



In terms of comparison, fine-tuning resulted in slightly better performance scores in accuracy. However all four models gave high accuracy in testing of 3000 images. Also we don't see huge differences in the F1-Scores and the Accuracy score which means the models were successful in correct classification.

Time Comparison Graph



In the training, each epoch took around 170 seconds and each model took 1700 seconds to train, However the first ever epoch of the training process took 2 hours 38 minutes due to preprocessing time of the data. Except for the first one, there are not much time differences among the epochs.

Table: Final Comparison of All Configurations

Configuration	Accuracy	Precision	Recall	F1 Score	Training Time (s)
LR = 0.001, Freeze All	0.9397	0.9242	0.9323	0.9282	11047.56
LR = 0.0001, Freeze All	0.9393	0.9261	0.9291	0.9276	1713.94
LR = 0.001, Fine-Tune Last	0.9650	0.9615	0.9546	0.9580	1719.30
LR = 0.0001, Fine-Tune Last	0.9587	0.9557	0.9450	0.9503	1721.22

4. Discussion

All four VGG-16 configurations performed well on the binary gender classification task, with models that included fine-tuning achieving slightly higher validation accuracy (around 0.96) compared to those with all layers frozen (around 0.94). While changing the learning rate had minimal impact on the frozen models, it caused slight variations in loss for the fine-tuned models, which also showed more fluctuation in metrics across epochs due to greater flexibility.

Each model trained for 10 epochs, averaging about 170 seconds per epoch. The first training session took significantly longer (2 hours 38 minutes) due to preprocessing. Testing on a separate set of 3,000 images (1,745 female and 1,250 male) confirmed high classification accuracy across all models, with confusion matrices showing balanced performance and no significant misclassification issues.

Confusion matrices also indicate that the model with fine-tuning and LR=0.001 had the **lowest false negative rate**, which is valuable in classification tasks where missing a positive instance is more costly than a false positive.

Comparative analysis revealed that fine-tuning had a greater effect on performance than learning rate adjustments. The best results came from the **fine-tuned model with a learning rate of 0.001**, which achieved the highest accuracy (0.9650) and overall performance, demonstrating the advantage of adapting deeper layers when using transfer learning.