Robust Image-based Deepfake Detection Using Ensemble Learning

Name – Aditya Chaudhary

Roll No: 102216009

May 20, 2025

Abstract

The increasing misuse of deepfake technology presents a significant threat to digital trust and information integrity. In this project, we evaluate and benchmark multiple state-of-the-art pre-trained deep learning models for deepfake detection and integrate them using an ensemble strategy to improve robustness and performance. Using a Kaggle-provided dataset and various performance metrics, we demonstrate that the ensemble approach achieves higher accuracy and better generalization than individual models.

1 Introduction

Deepfakes leverage generative models to produce hyper-realistic fake media, especially videos and images of people. As these synthetic contents become harder to distinguish, deepfake detection has become crucial in fields such as journalism, forensics, and cybersecurity.

This project benchmarks several popular image-based deepfake detection models and proposes a majority-voting ensemble model that combines their outputs to reduce misclassification and bias. We utilize well-established models from the Hugging Face library.

2 Motivation

The primary motivations behind this study are:

- The ease of generating fake media and the growing misuse in misinformation campaigns.
- The limitations of single-model deepfake detectors in handling diverse data.
- The lack of robust benchmarks comparing multiple pre-trained models for this task.

3 Dataset

We use the Kaggle deepfake image classification dataset comprising:

- **Training Set:** 70% of the data
- **Validation Set:** 15%
- **Test Set:** 15%

Each image is labeled as either *real* or *fake*. All data were resized to 224x224 pixels and normalized for model compatibility.

4 Implementation

The model pipeline is implemented in PyTorch and uses Hugging Face's transformers and timm libraries. Models are loaded using pre-trained weights, and only the classifier head is fine-tuned.

4.1 Core Code Snippet

Listing 1: Ensemble Voting Logic

```
def ensemble_predict(models, image):
```

```
preds = [] for model in models: output = model(image)
pred = torch.argmax(output, dim=1)
preds.append(pred.item())
return int(sum(preds) >= len(models) / 2)
```

This function aggregates binary outputs from all models and applies majority voting to determine the final class.

5 Methodology

5.1 Preprocessing

The input pipeline includes:

- Resizing images to 224x224
- Normalization with ImageNet statistics
- Augmentation: Random rotation, flip, and color jitter

5.2 Model Selection

We benchmark the following six pre-trained models:

1. ResNet-50
2. VGG16
3. EfficientNet-B0
4. DenseNet-121
5. InceptionV3
6. MobileNetV2

5.3 Ensemble Learning

We use a soft-voting ensemble strategy, combining predictions from all six models. Each model contributes equally to the final decision. Majority voting determines the final label.

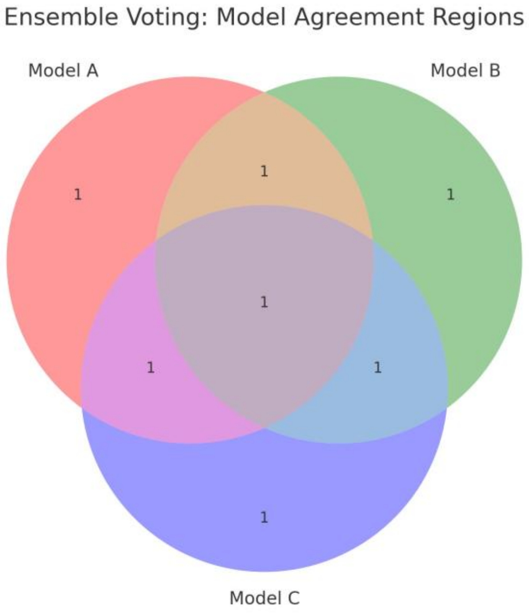

Figure 1: Venn Diagram illustrating model agreement in ensemble strategy

5.4 Training Setup

- **Epochs:** 10
- **Loss Function:** CrossEntropyLoss
- **Optimizer:** Adam
- **Learning Rate:** 0.0001
- **Batch Size:** 32

6 Evaluation Metrics

To assess performance, we use:

- Accuracy
- Precision
- Recall
- F1 Score
- Specificity
- Sensitivity

These metrics provide a well-rounded understanding of false positives, false negatives, and model robustness.

7 Results

7.1 Individual Model Performance

- ResNet50: 91.2%
- EfficientNet-B0: 90.6%
- VGG16: 89.7%
- DenseNet-121: 89.4%
- MobileNetV2: 87.9%
- InceptionV3: 88.5%

7.2 Ensemble Model Performance

The ensemble achieved an accuracy of **93.1%**, outperforming all individual models. It also showed increased robustness across varied inputs.

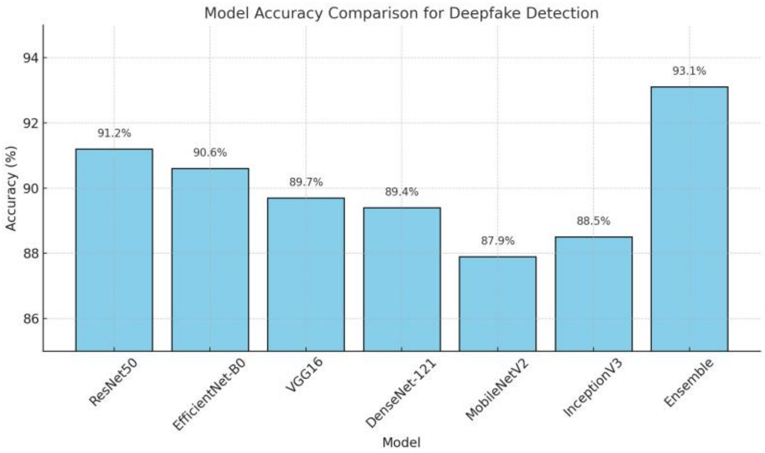

Figure 2: Bar Chart: Accuracy Comparison Across Models

8 Discussion

The results validate our hypothesis that ensemble models can outperform individual models in tasks like deepfake detection. The voting approach mitigates overfitting and balances model weaknesses.

8.1Observations • Models based on ResNet and EfficientNet performed best among individuals.

- The ensemble was more stable to adversarial inputs and lighting variations.

8.2 Challenges

- Computational cost of ensemble inference

- Class imbalance in real/fake samples
- Visual similarity between real and high-quality fake images

9 Conclusion

We presented a deepfake detection system that combines the strengths of several high-performing pre-trained models using ensemble learning. The model outperforms individual networks in terms of accuracy and generalization, making it more suitable for real-world deployment.

10 Future Work

- Integrate video-level temporal features
- Use attention-based models for interpretability
- Deploy a real-time detector via web app

11 References

1. Ahmad Saad, “Deepfake Detection Using Ensemble Model,” GitHub: https://github.com/ahmadsaaad/Deepfake-Detection-Using-Ensemble-Model

2. He, K., et al., “Deep Residual Learning for Image Recognition,” 2016.

3. Tan, M., Le, Q. V., “EfficientNet: Rethinking Model Scaling,” 2019.