Effectiveness of Data Synthesized by Generative AI in Supervised Image Classification

CS 539 Group 9

Diego Pena-Stein, Bashir Gulistani, Ehu Shubham Shaw, Steffi Dorothy

Motivation

- Data Scarcity in Image Classification
- Potential of Generative AI
- Can we use it to enhance or replace existing datasets or models?

About

- We chose to test this on supervised image classification because of the ease of access to free, highly realistic models that can run locally.

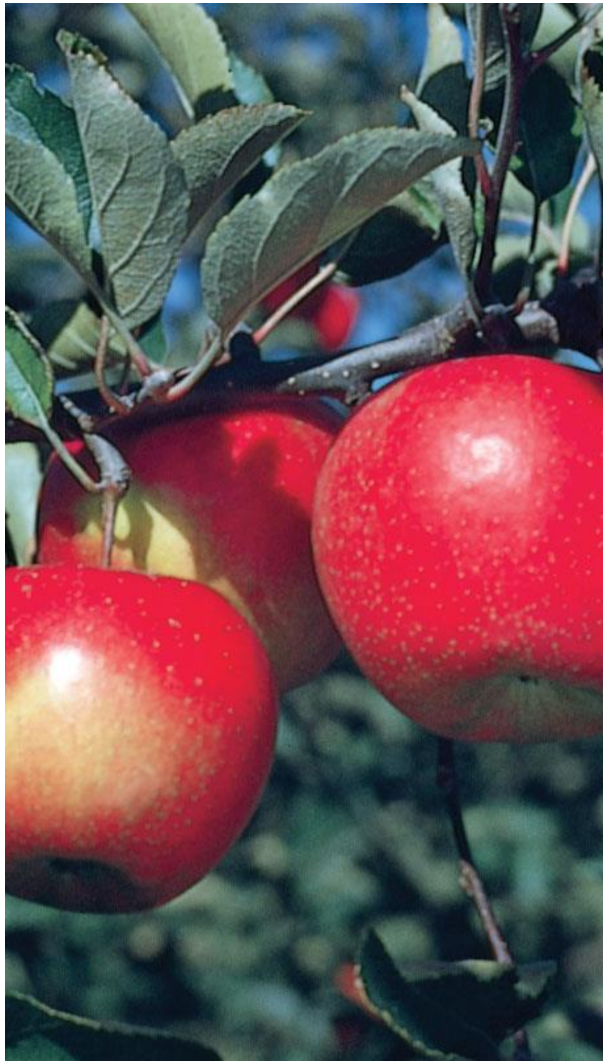
- Stable diffusion: Able to prompt with a text input and get a different but accurate result each time. Perfect for automating the generation of images belonging to specific classes.

What is Stable Diffusion?

Stable Diffusion: A text-to-image diffusion model by Stability AI.

Function: Generates high-quality images by iteratively denoising a random noise image guided by input text.

Synthetic Data Generation: Create diverse, high-quality images.

Which of these images is real?

Datasets

- Real Data:

  Source : Kaggle

  Categories : Fruits

- Synthetic Data:

  Generated Using : Stable Diffusion

  Total Images : 4400

Image Preprocessing

- May be tempted to simply feed the raw images to the classification model
- While it works well for a single unified dataset, the model performs much worse when tasked to generalize to a new dataset
- Possibly because the model learned features unique to each dataset
- Preprocessing the images is useful for having each dataset closely resemble the other

Preprocessing: Aspect Ratio

- While stable diffusion always produces 512x512 square images, real datasets may not have this property
- At first, simply rescaled each image down to a standard 64x64
- This led to an issue where non-square images lost their aspect ratio, 'stretching' or 'squeezing' the image
- Not an issue for datasets where this appeared normally, but models trained on only square images lost a lot of accuracy when presented with stretched or squeezed images.

Preprocessing: Margins

Resize image while maintaining aspect ratio

- Now dataset is unsqueezed – or is it? While the original data now has margins, our synthetic data is still only square.

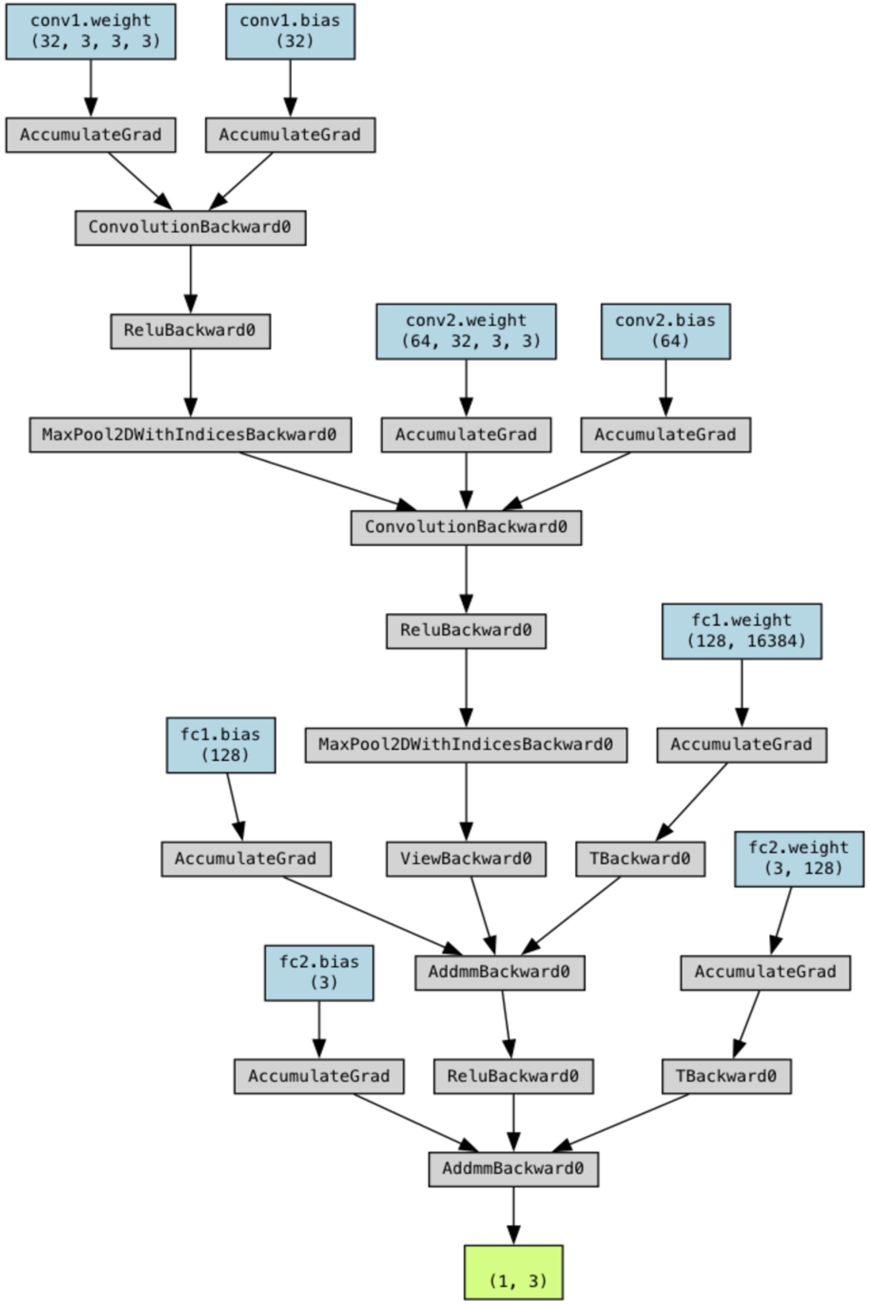
Randomly remove a fixed margin

- Solution: Introduce a randomly sized margin into the synthetic images to better resemble the original datasets

Preprocessing pipeline

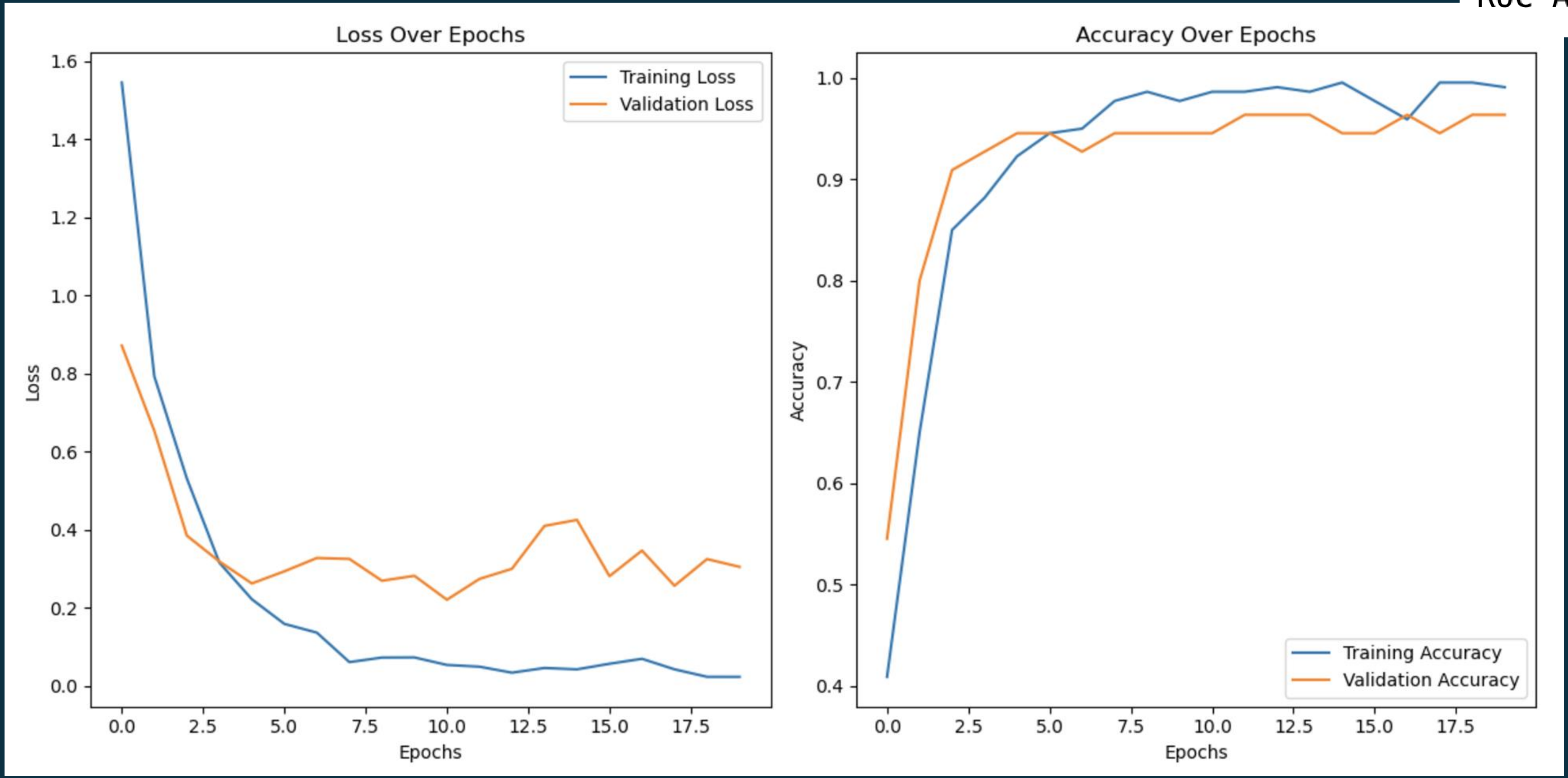
Model Architecture

- Input Tensor (batch size:1, RGB, and image: 64x64)
- Convolution Layer 1 (32 filters, each size 3x3, applied to 3 input channels.
  - produces a feature map
- ReLU Activation (ReLU) -> Introduce Non-Linearity
- Max Pooling Layer 1 (Downsizing to half)
- Convolution Layer 2 (64 filters)
- ReLU Activation
- Max Pooling Layer 2
- Flattening (4D tensor (batch, channels, height, width) is flattened into a 2D)
- Fully Connected Layer 1
- Dropout
- Fully Connected Layer 2
- Output

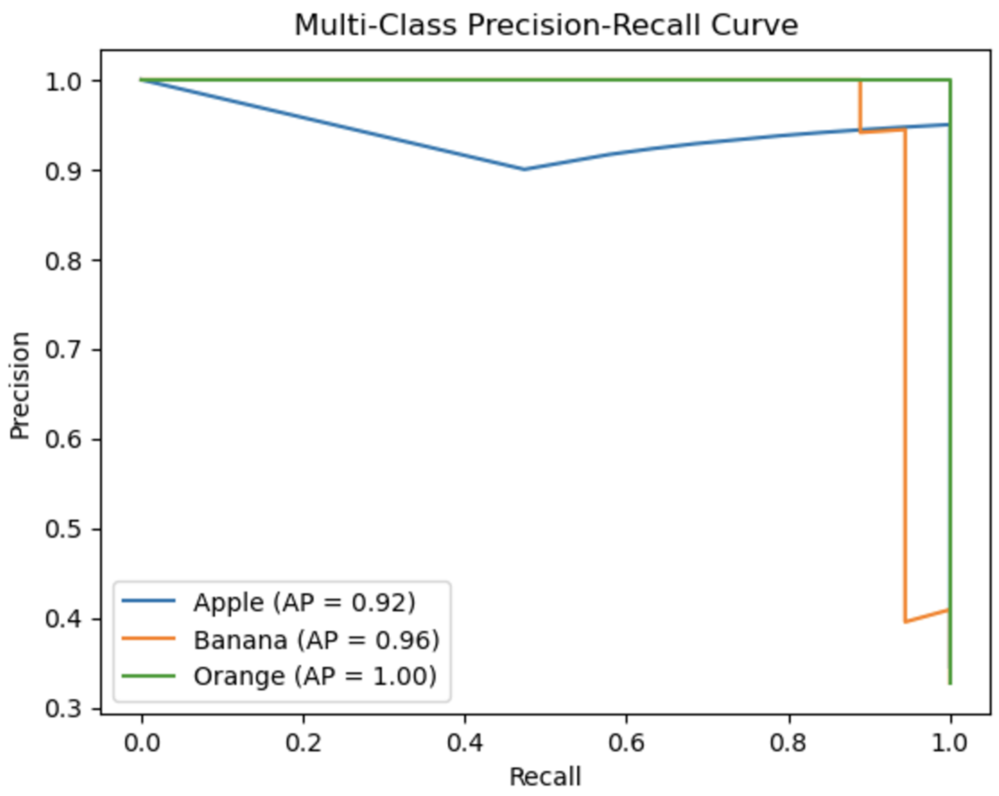


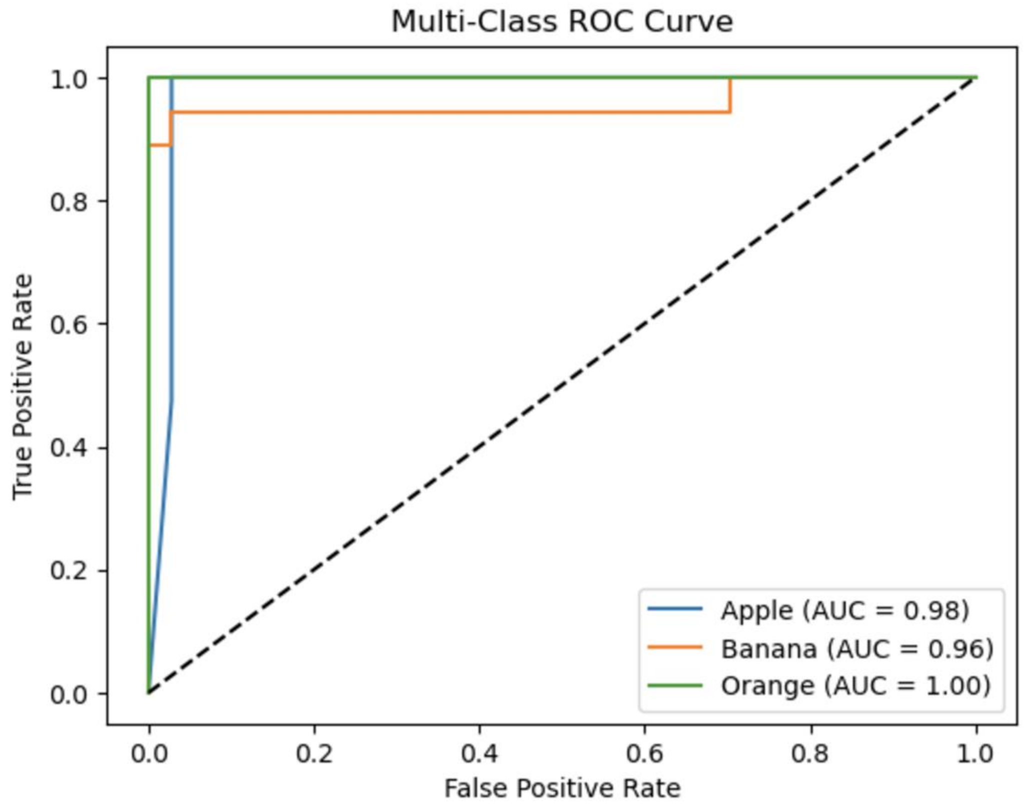
Model 1: Results (Original Images)

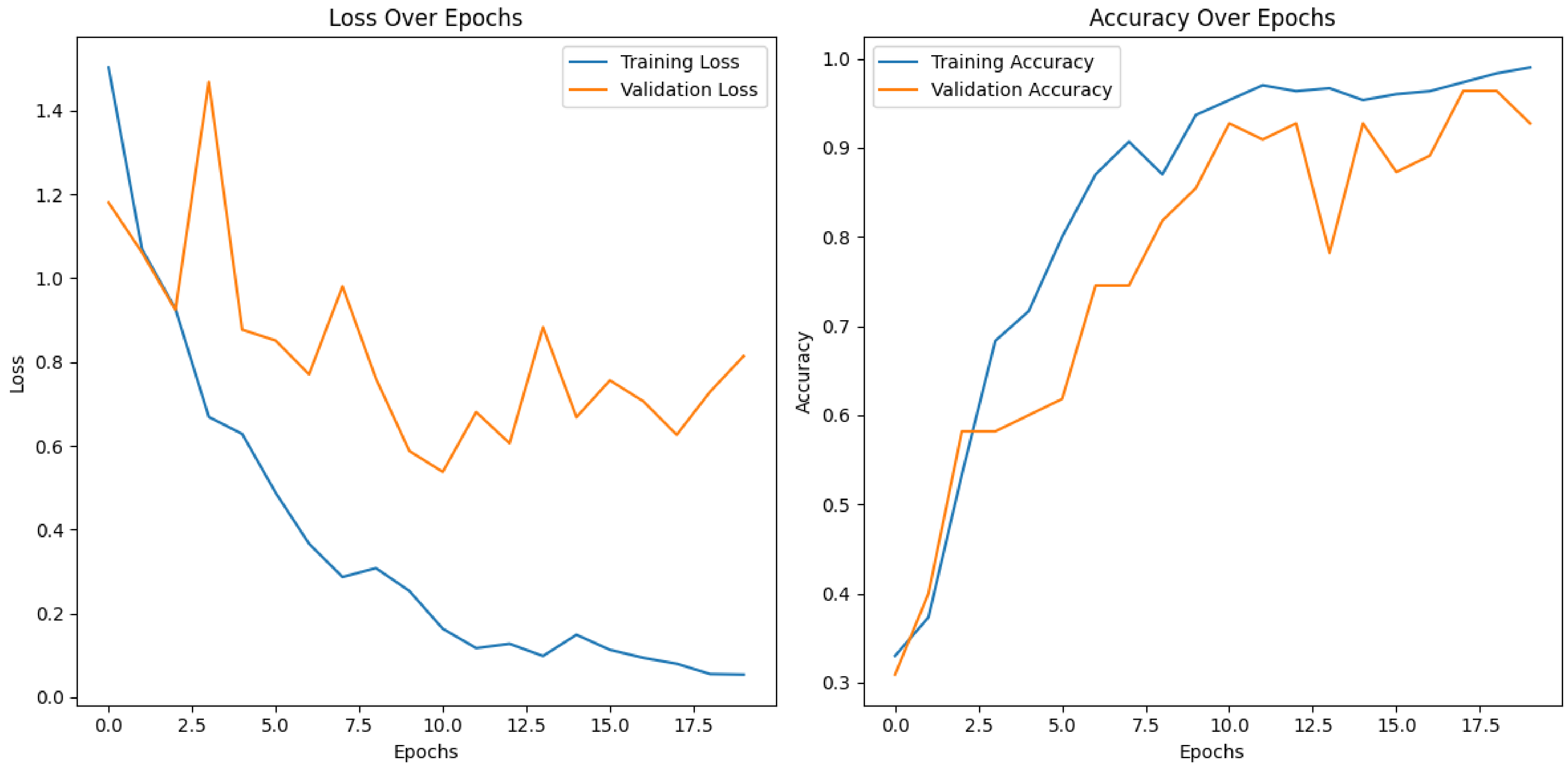
```
Evaluation Metrics:
Accuracy: 0.9636
Precision: 0.9655
Recall: 0.9636
F1-score: 0.9630
ROC-AUC: 0.9865
```



ROC Curve & Precision-Recall Curve (Original)

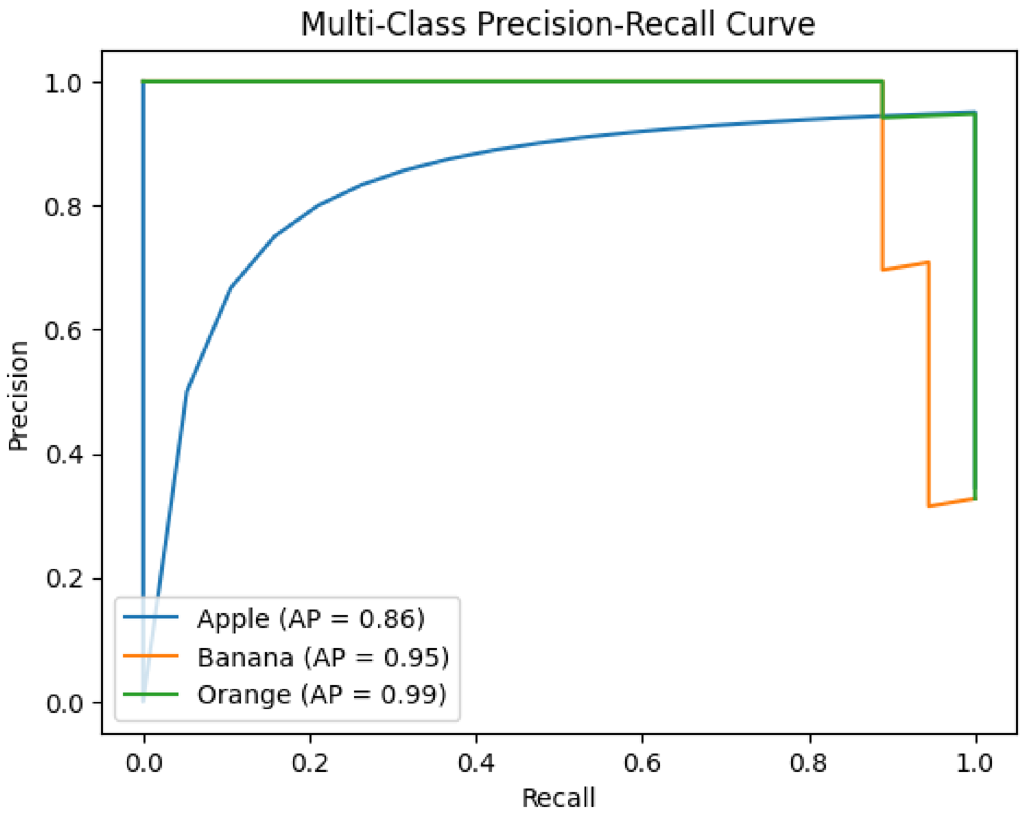


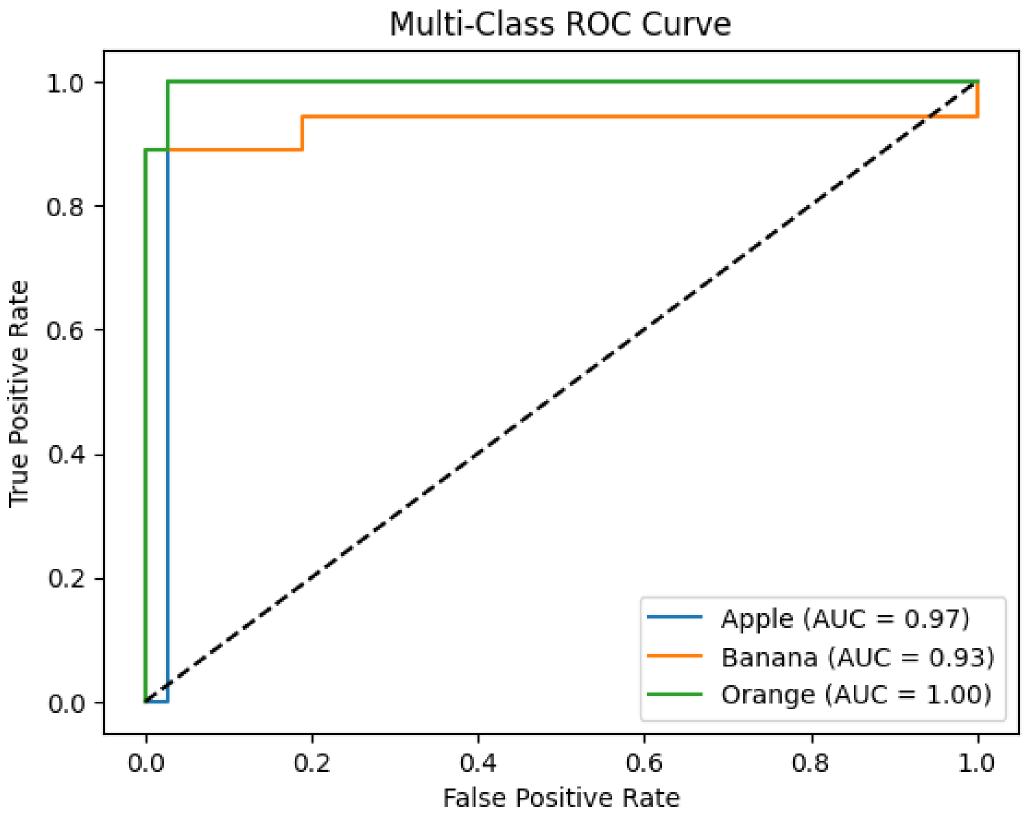




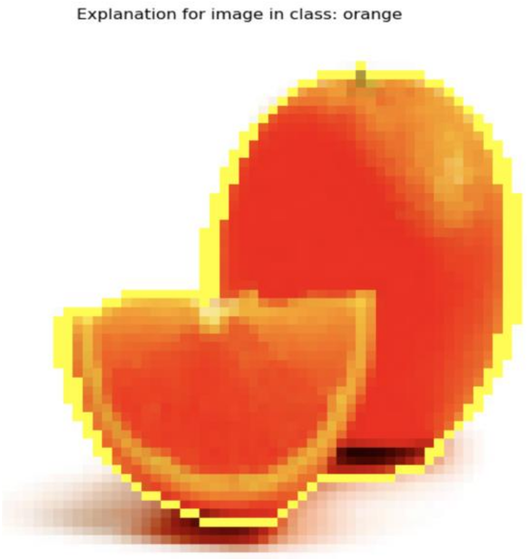
Model 2:
All Synthetic Images Results

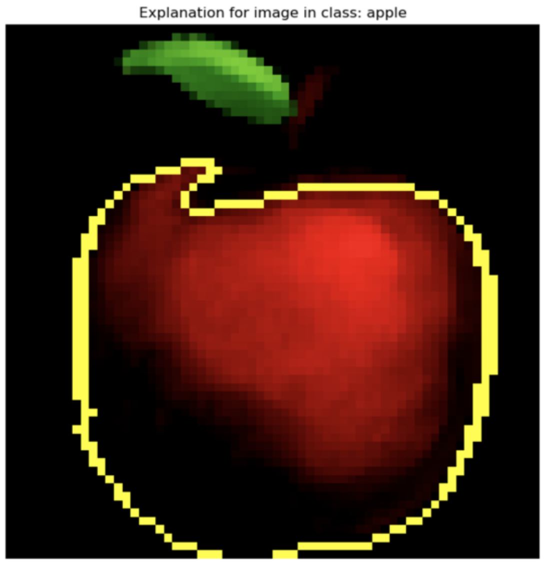
ROC Curve & Precision-Recall Curve (Synthetic)





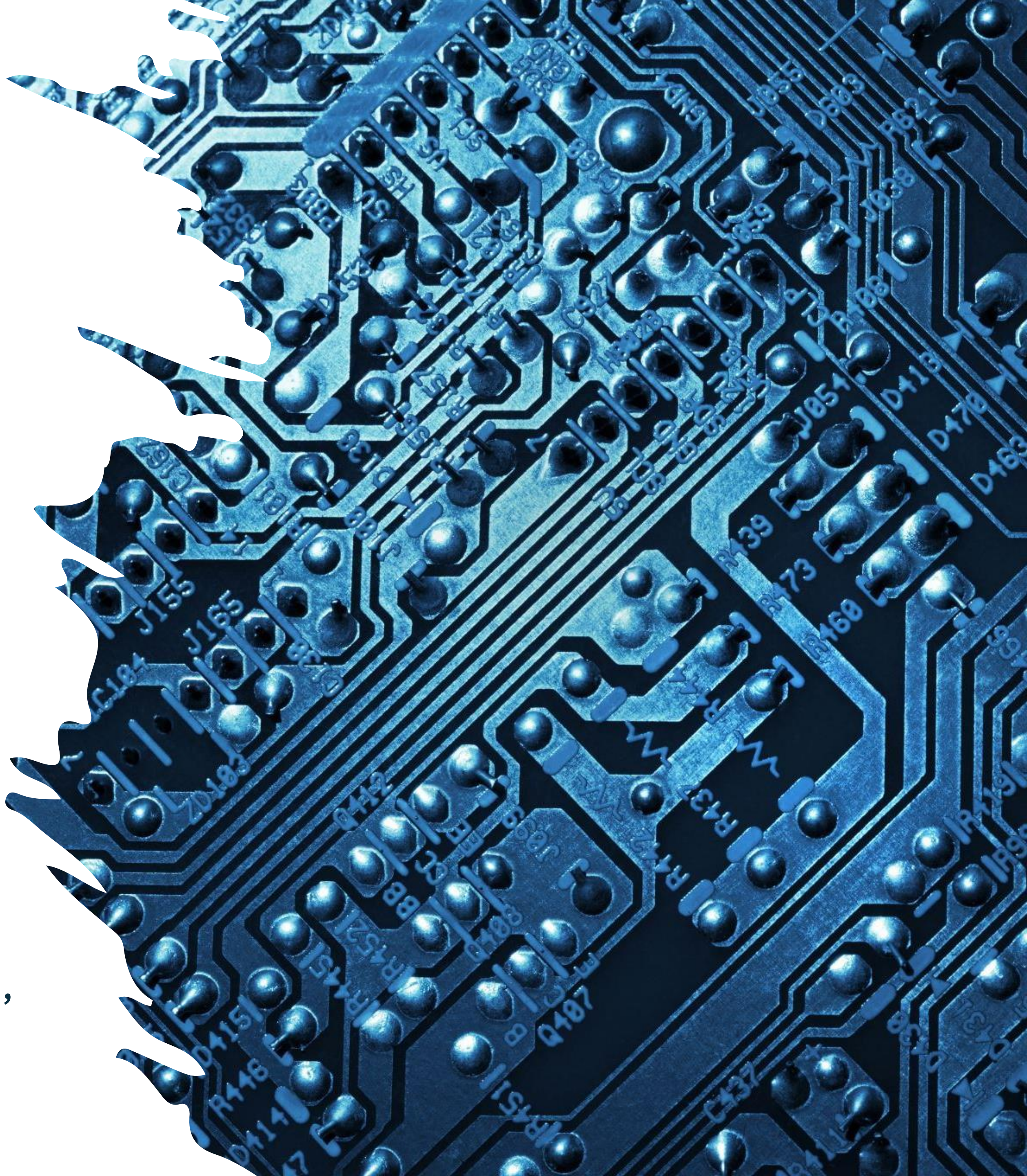
Model Explainability





Key Takeaways

- Stable diffusion takes ~30 seconds to generate a single image on our hardware (Low end CUDA compatible GPU)
  - This may be sped up with parallel computing
  - Or a GPU with enough memory to generate multiple images at once
- An understanding of the nature of the data to be used with the model is required to build a good preprocessing pipeline
  - Can explore more advanced preprocessing techniques in the future, which could further expand the dataset
  - Black and white images, random noise, dropout, colored margins, etc.

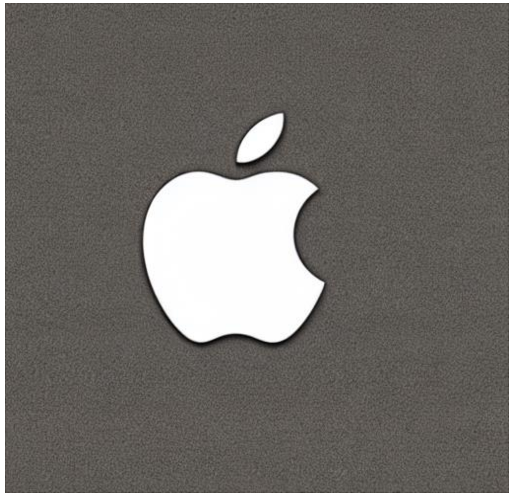


Key Takeaways

- Diffusion models may have inherent bias with certain prompts, and vague class names produce undesirable images
- Can try other models like Flux, or data mediums other than images (Time series, textual, etc.)
- Instead of training on synthetic or real data in a single step, Pre-train on synthetic data and fine-tune on real data

Prompt: Ginger

Prompt: Apple



Thank you

Code:
https://github.com/DPS100/DataSynthesizer