UNIVERSITY OF PASSAU

Data Science Lab

Topic 3: Data Synthesis

Group 5:
Hai Dang Do
Amandeep Singh Gill
Tarek Al Bouhairi
Mohamad Yehya

Winter Semester 2023/2024

AGENDA

UNIVERSITY OF PASSAU

1 INTRODUCTION

2 WORKFLOW AND SETUP

3 IMPLEMENTATION

4 EVALUATION AND CONCLUSION

Data Augmentation and Data Synthesis

- **Goal**: Evaluate the impact of adding synthetic data generated through Generative Adversarial Networks (GANs) on the performance of a classification model.
- **Subtasks**:

1. Development of Classification Neural Network Model
2. Training of GAN for Synthetic Image Generation
3. Integration of Original and Synthetic Images
4. Performance Evaluation of the Classification Model

- **Reason**: Lack of quality open-source data in breast mammography domain.

Generative AI models

- **Variational Autoencoders (VAE):** Neural networks for learning and generating data by capturing underlying patterns in the data.
- **Diffusion models:** Framework describing gradual data transformations over time, useful for simulating complex processes.
- **Generative Neural Network (GAN):** Neural network pair - a generator creates data, and a discriminator distinguishes real from generated data, producing high-quality synthetic content.

Why experiment with synthetic data?

- **Addressing Data Scarcity:** GANs, VAEs, and diffusion models can generate extra data to augment training sets, especially in data-limited domains.
- **Balancing Distributions:** These models aid in balancing class distributions and reducing overfitting risks through the generation of synthetic data.
- **Improving Generalization with Diverse Data:** These models produce diverse and realistic data, enhancing the generalization and robustness of machine learning models.

About the dataset

- **Domain:** Healthcare (Breast Cancer Mammography images)
- **Dataset:** CBIS-DDSM from Kaggle
- **CBIS-DDSM** is an updated and standardized version of the DDSM dataset.
- The **dataset** includes decompressed **DICOM images**, updated **Region of Interesting** (ROI) segmentation, and **pathology** information.
- **Classes** within the **dataset**:
  - “Malignant”: 0
  - “Benign” and “Benign without callback”: 1

Two main stages of Implementation

Stage 1:

- Collect and preprocess CBIS-DDSM dataset.
- Train CNN using dataset.
- Evaluate CNN performance.

Stage 2:

- Generate new images using a conditional GAN.
- Retrain CNN with generated images.
- Reevaluate CNN performance metrics for comparison.

Data Collection and Feature Engineering

Data Collection:

- Integrated CBIS-DDSM dataset, curated for Breast Cancer Detection.

Feature Engineering:

- Corrected image paths using dictionaries
- Renaming columns for consistency.
- Applied ‘bfill’ method to handle missing values in the dataset.

Data Analysis:

- Breast Cancer Mass Types Overview
- Visualization of Image Distribution

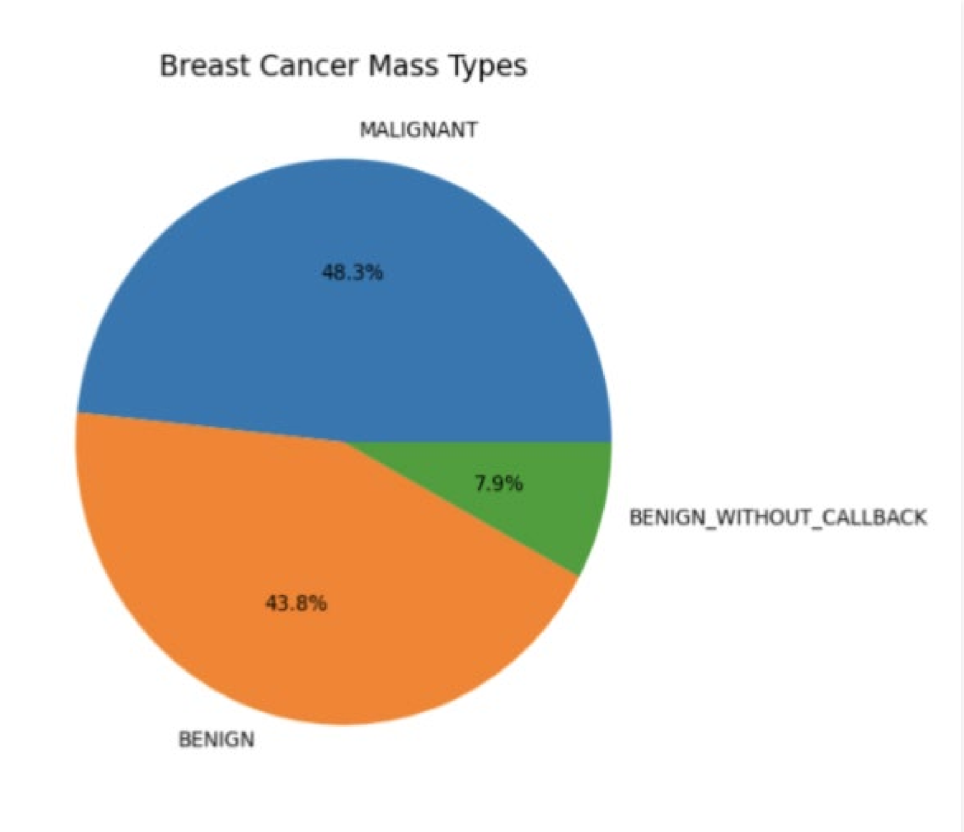
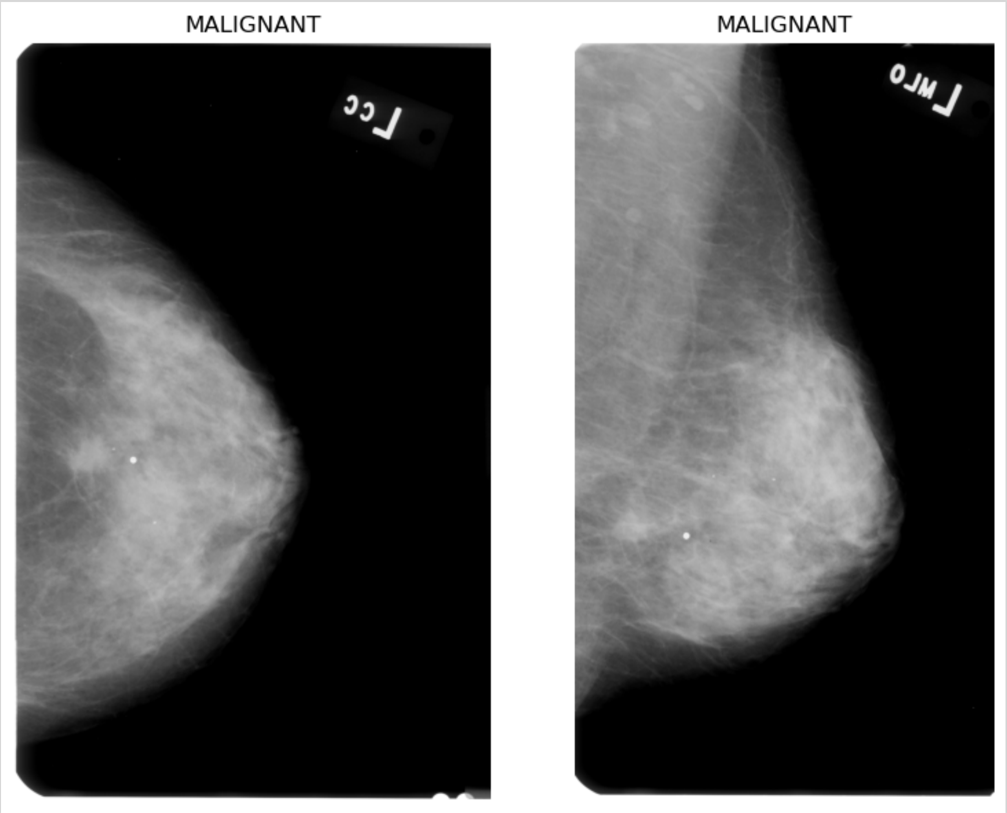
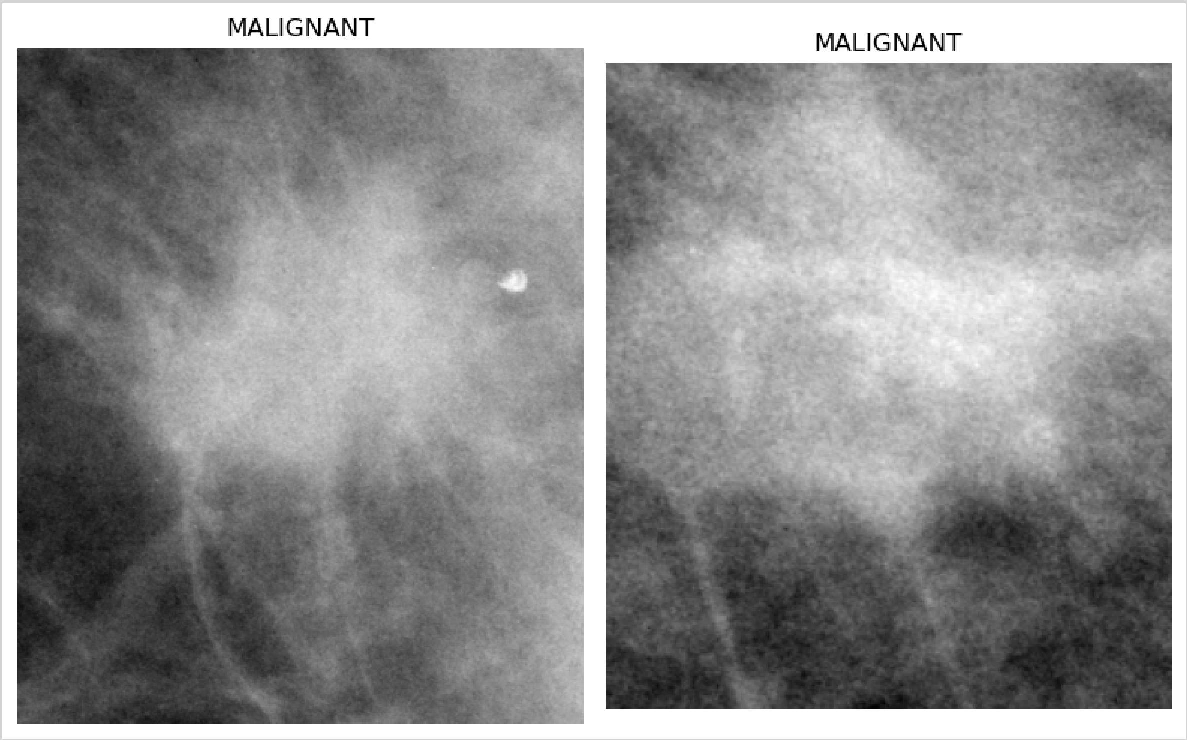


Fig. 1. Breast Cancer Mass Types

Data Processing



Full Mammograms



Cropped Mammograms

Data Processing

Original Datasets Shape: mass_train: (1318, 14) mass_test: (378, 14)

Image Processing:

- Merge Datasets
- Image resizing.
- Color space conversion
- Normalization for pixel value standardization.
- Data splitting

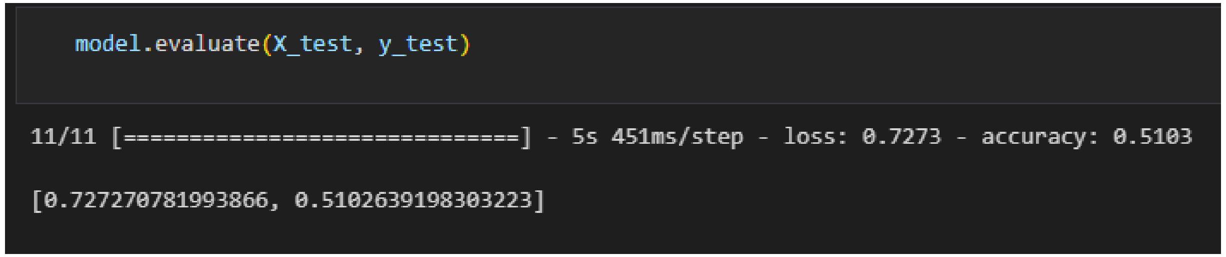
Data Augmentation Techniques:

- Rotation, shifts, shearing, zooming, flipping, brightness adjustment, channel shift, and fill mode.
- Gaussian Blur for noise reduction, Histogram Equalization

Base Model with Convolution Neural Network

Basic Model:

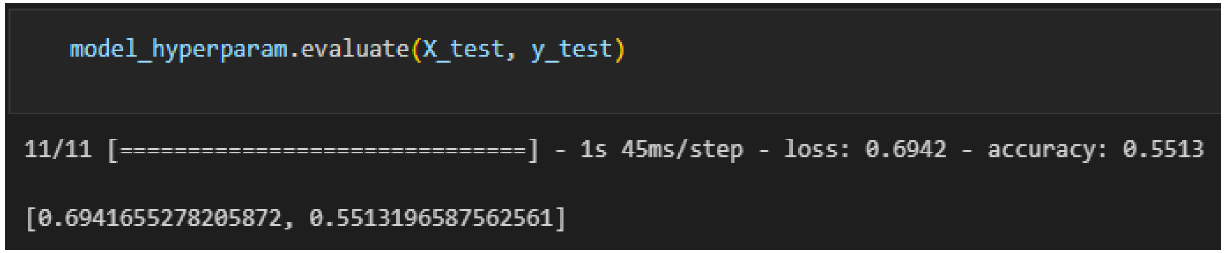
- Simple Convolutional Neural Network
- Relevance in Breast Cancer Images



```
model.evaluate(X_test, y_test)

11/11 [==============================] - 5s 451ms/step - loss: 0.7273 - accuracy: 0.5103

[0.727270781993866, 0.5102639198303223]
```

Hyperparameters with Optuna:

- Number of Convolutional layers
- Number of Filters per Layer
- Number of Units in Dense Layer
- Dropout
- Learning Rate

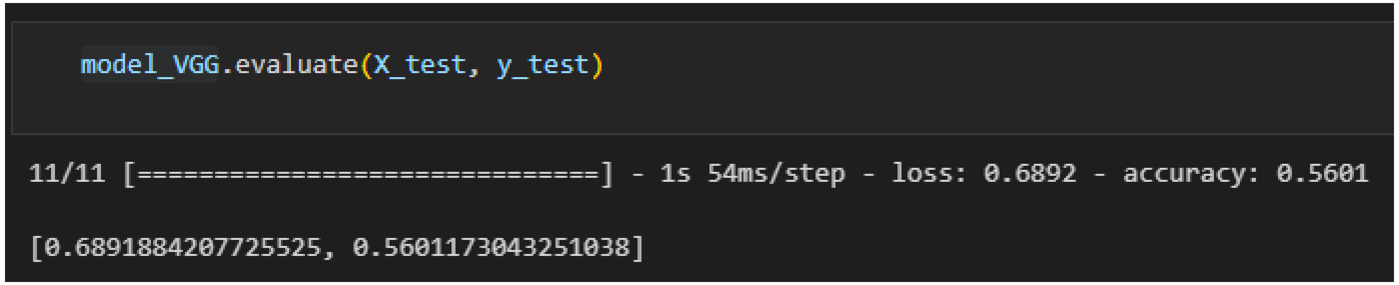


```
model_hyperparam.evaluate(X_test, y_test)

11/11 [==============================] - 1s 45ms/step - loss: 0.6942 - accuracy: 0.5513

[0.6941655278205872, 0.5513196587562561]
```

CNN Model with VGG16 Architecture

VGG16’s architecture:

- Structure: Consists of 16 layers, including convolutional, pooling, and fully connected layers.
- Simplicity: Known for its straightforward and uniform structure.
- Convolutional Layers: Followed by pooling layers, creating a hierarchical feature extraction process.

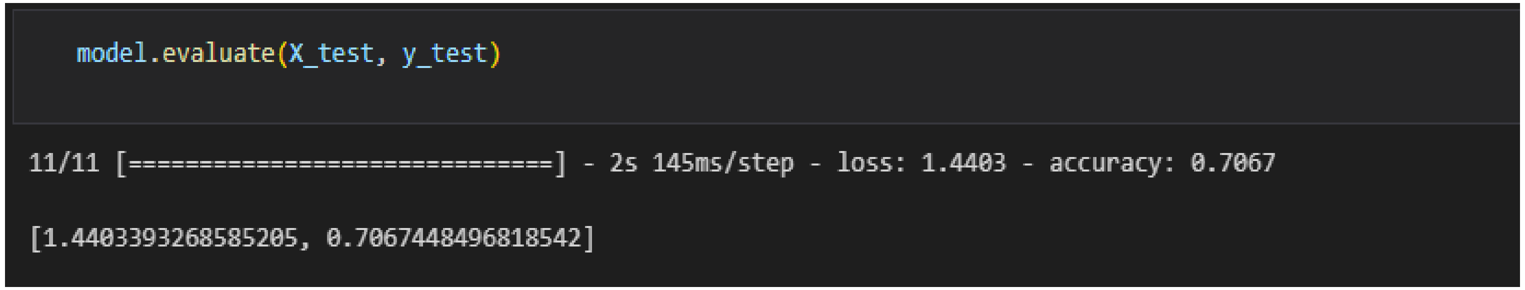


```
    model_VGG.evaluate(X_test, y_test)

11/11 [==============================] - 1s 54ms/step - loss: 0.6892 - accuracy: 0.5601

[0.6891884207725525, 0.5601173043251038]
```

CNN Model with InceptionResNetV2

InceptionResNetV2’s architecture:

- Combination: Merges the *Inception* and *ResNet* architectures.
- Deep Network: Consists of multiple layers with intricate connections.
- Feature Extraction: Utilizes various inception modules to extract features at different scales.
- State-of-the-Art Performance: Achieves high accuracy on various image classification benchmarks.



```
model.evaluate(X_test, y_test)
```

```
11/11 [==============================] - 2s 145ms/step - loss: 1.4403 - accuracy: 0.7067

[1.4403393268585205, 0.7067448496818542]
```

CGAN Model working with CBIS-DDSM dataset

Generator Architecture:

Input:

- latent_dim: Dimensionality of the noise vector
- num_classes: Number of classes or labels (used for conditional generation)
- img_shape: Shape of the output image (For the CBIS-DDSM Dataset the shape is (224, 224, 3).

Upsampling blocks: 5 *Conv2DTranspose* layers

- Increasing the dimensions from (7, 7, 128) to target size 224x224

CGAN Model working with CBIS-DDSM dataset

Discriminator Architecture:

Input Layers

- image: Input for the data
- label: Input for the label data (num_classes = 2)

Validity

- takes generated images and the target label as input
- validity = discriminator([generated_image, label])

Model Compilation:

- Model([noise, label], validity)
- Combines the input layers and output layer into a Keras Model

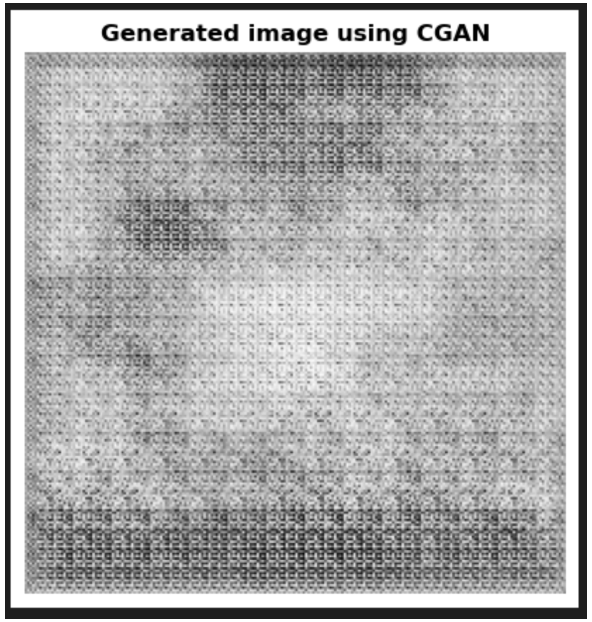
Apply CGAN Model to synthesize new images

Image Comparison:

- The generated images did not really closely resemble the original data within the CBIS-DDSM dataset

Challenges Encountered:

- Discrepancy from Real Images: translating scanned films, mostly observed as grayscale images, into RGB images of size 224x224x3
- Data Size Limitation - Original data: (1696, 224, 224, 3)
- Training Time
    - 1000 – 3000 epochs
    - 1-2 hours per trial

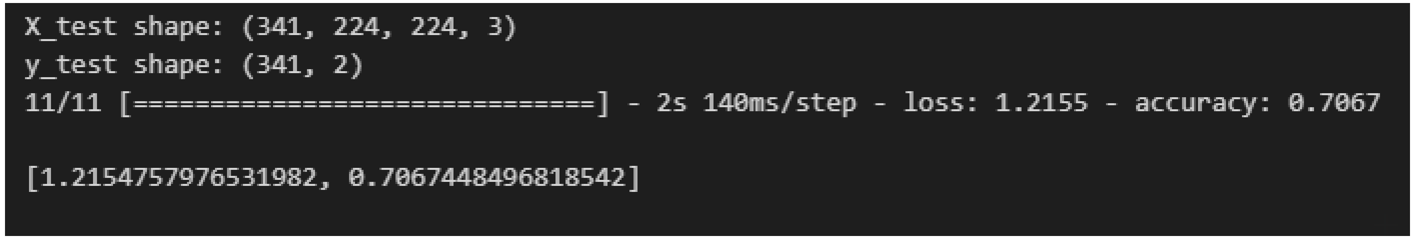

Generated image using CGAN

Combine CGAN to InceptionResNetV2 model

Proportion between Original and Synthetic Images:

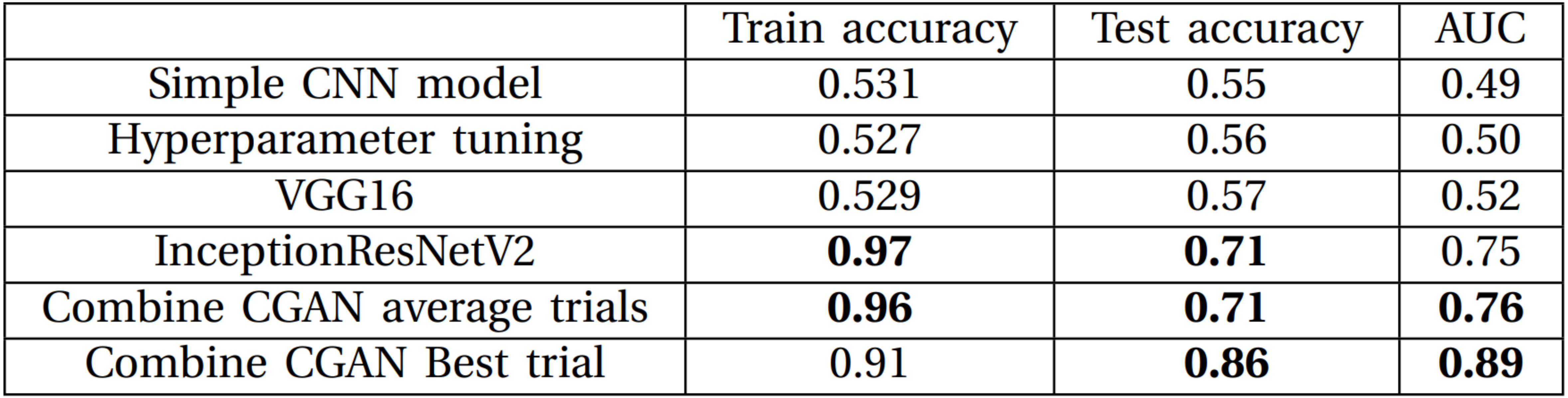
- Original data: 1187 images
- Synthetic data: 550 images (300 Benign images, 250 Malignant images)

→ New Training Data: 1737 images



```
X_test shape: (341, 224, 224, 3)
y_test shape: (341, 2)
11/11 [==============================] - 2s 140ms/step - loss: 1.2155 - accuracy: 0.7067

[1.2154757976531982, 0.7067448496818542]
```

CNN Models Performance Comparison

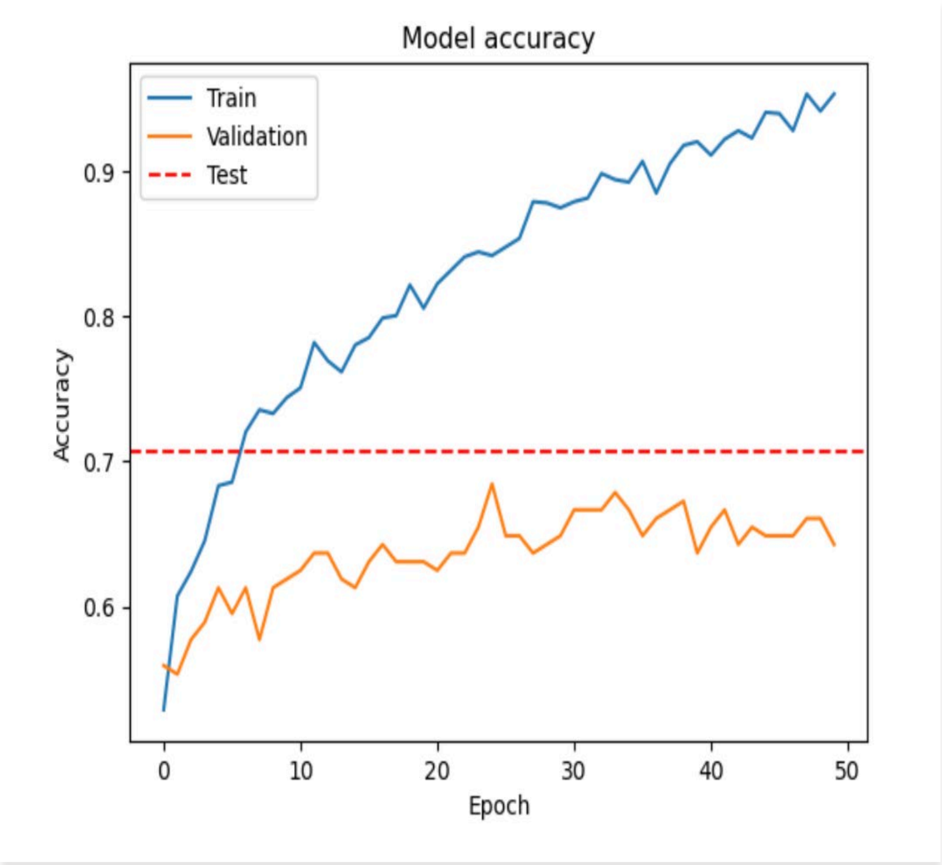


	Train accuracy	Test accuracy	AUC
Simple CNN model	0.531	0.55	0.49
Hyperparameter tuning	0.527	0.56	0.50
VGG16	0.529	0.57	0.52
InceptionResNetV2	**0.97**	**0.71**	0.75
Combine CGAN average trials	**0.96**	**0.71**	**0.76**
Combine CGAN Best trial	0.91	**0.86**	**0.89**

Evaluation

Classification stage:

- Initial approach:
  - Simple CNNs Classification Model with *accuracy 0.55*
  - **VGG16** for image classification with *accuracy 0.57*
- Transitioned to **InceptionResNetV2**, improved *accuracy 0.71*



InceptionResNetV2 Model's Result

Evaluation

Generative stage

After many implementations with trial/error, we constructed a conditional GAN to generate synthetic data.

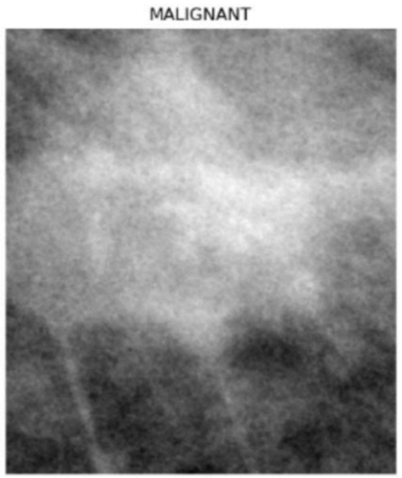


Fig. 4. Original Image from CBIS-DDSM Dataset

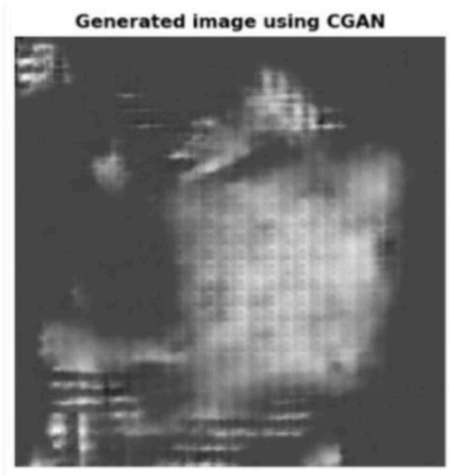


Fig. 5. The generated image from the CGAN model

Result analysis

Synthetic Image Integration with 550 synthetic images via Conditional GANs

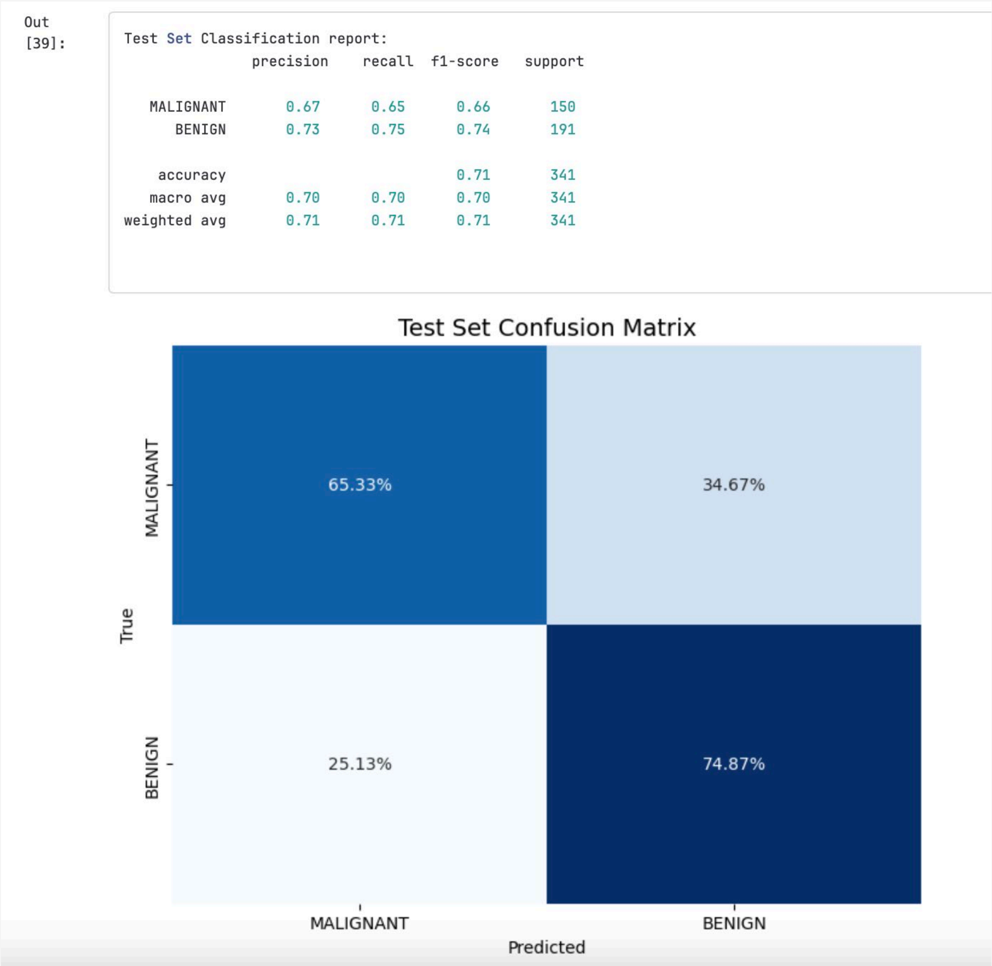
- 300 Benign images
- 250 Malignant images

Re-evaluated InceptionResNetV2 with Original Dataset + Synthetic Data

- Model maintained accuracy of 0.71, but demonstrated enhanced performance in specific areas.

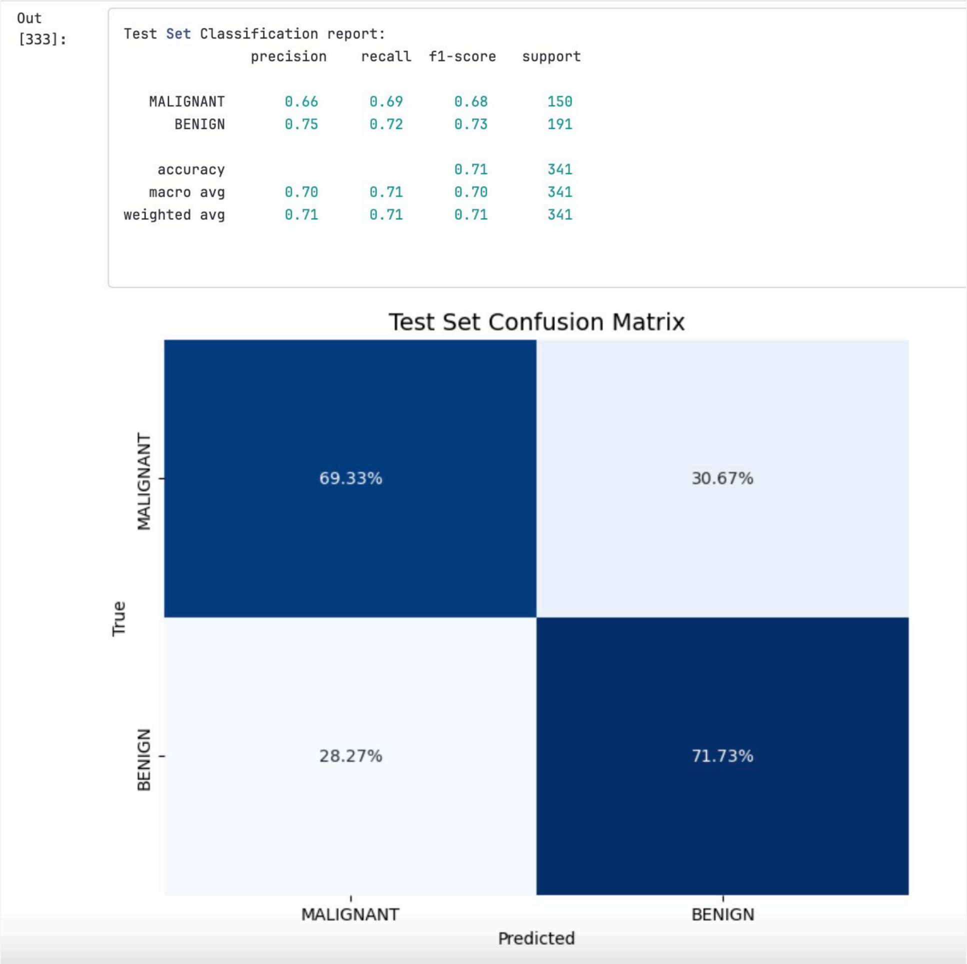
Insights from confusion matrix highlight improved classification for Malignant, but not for Benign cases.

EVALUATION AND CONCLUSIONS



```
Out
[39]:   Test Set Classification report:
                     precision    recall  f1-score   support

           MALIGNANT       0.67      0.65      0.66       150
              BENIGN       0.73      0.75      0.74       191

            accuracy                           0.71       341
           macro avg       0.70      0.70      0.70       341
        weighted avg       0.71      0.71      0.71       341
```

Classification for Original Dataset

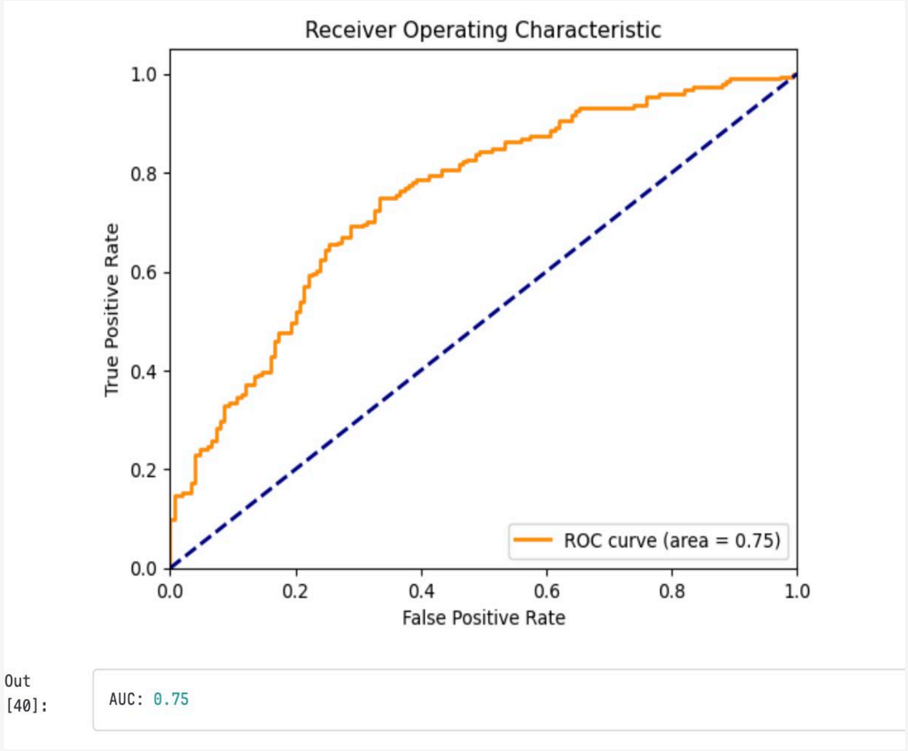


```
Out
[333]:  Test Set Classification report:
                     precision    recall  f1-score   support

           MALIGNANT       0.66      0.69      0.68       150
              BENIGN       0.75      0.72      0.73       191

            accuracy                           0.71       341
           macro avg       0.70      0.71      0.70       341
        weighted avg       0.71      0.71      0.71       341
```

Classification for Original Dataset + Synthetic Data

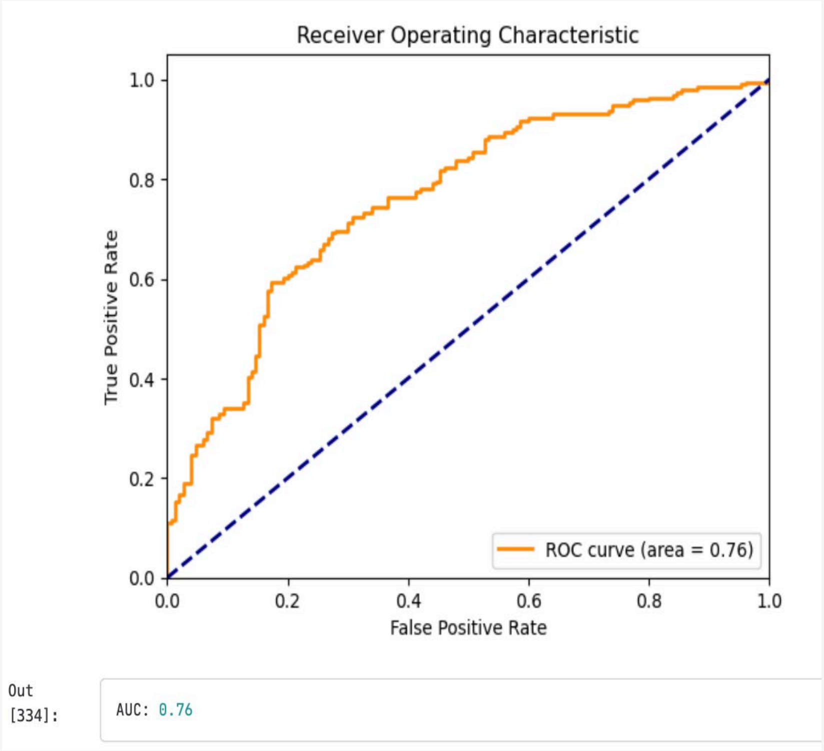
4 EVALUATION AND CONCLUSIONS

Classification for Original Dataset



```
Out
[40]:   AUC: 0.75
```

Classification for Original Dataset + Synthetic Data



```
Out
[334]:   AUC: 0.76
```

- While the increase is modest, it indicates that the introduction of synthetic data has had a positive impact on the model's ability to discriminate between classes
- The improvement in AUC suggests that the model is making better distinctions between true positive and false positive rates

Discussion

- The Conditional Generative Adversarial Network (GAN) displayed instability, yielding inconsistent results and posing challenges in achieving reliability and reproducibility.

- There should be a significant room for improvement in generating more realistic images with the complex dataset, emphasizing the need for further refinement in the generative model.

Conclusion

- CGANs demonstrate potential method for image generation
- Challenges encountered with the CBIS-DDSM dataset influenced the model's ability to generate accurate and representative images
- Further exploration are necessary to overcome these obstacles and achieve better results with complex medical imaging datasets.

REFERENCES

[1] Oza, Pratik and Sharma, Prashant and Patel, Samir and Adedoyin, Folashade and Bruno, Agostinho, Image Augmentation Techniques for Mammogram Analysis, Journal of Imaging, vol. 8, no. 5, pp. 141, May 20, 2022, https://www.mdpi.com/2313-433X/8/5/141/htm

[2] Ding, Kaiyue and Zhou, Ming and Wang, Hao and others, A Largescale Synthetic Pathological Dataset for Deep Learning-enabled Segmentation of Breast Cancer, Scientific Data, vol. 10, p. 231, 2023, https://www.nature.com/articles/s41597-023-02125-y

[3] Cha, Kuo Han and Petrick, Nicholas and Pezeshk, Aria and Graff, Christian G. and Sharma, Deep and Badal, Andreu and Sahiner, Berkman, Evaluation of Data Augmentation via Synthetic Images for Improved Breast Mass Detection on Mammograms Using Deep Learning, Journal of Medical Imaging (Bellingham), vol. 7, no. 1, p. 012703, 2020, https://doi.org/10.1117/1.JMI.7.1.012703, Epub 2019 Nov 22, PMID: 31763356, PMCID: PMC6872953

[4] Rebuffi, Sylvestre-Alvise and Gowal, Sven and Calian, Dan Andrei and Stimberg, Florian and Wiles, Olivia and Mann, Timothy A, Data Augmentation Can Improve Robustness, In Advances in Neural Information Processing Systems, pp. 29935–29948, Curran Associates, Inc.,2021. https://proceedings.neurips.cc/paper_files/paper/2021/file/fb4c48608ce8825b558ccf07169a3421-Paper.pdf

[5] Author, A. (2013). I.J. Image, Graphics and Signal Processing, 5(6), 47-54. Published Online April 2013 in MECS http://www.mecs-press.org/DOI:10.5815/ijigsp.2013.05.06

[6] Breast Cancer CNN Model. By JOSHUA AMPOFO YENTUMI. https://www.kaggle.com/code/joshuaampofoyentumi/breastcancer-cnn

[7] CBIS-DDSM: Breast Cancer Image Dataset. Kaggle. https://www.kaggle.com/datasets/awsaf49/cbis-ddsm-breast-cancer-image-dataset/code

[8] DenseNet169 Model. By Hithesh M R from Kaggle https://www.kaggle.com/code/hitheshmr/densenet169-cbis-ddsm

[9] Dive into Deep Learning - Chap 7: Convolutional Neural Networks by Aston Zhang (Author), Zachary C. Lipton (Author), Mu Li (Author), Alexander J. Smola (Author) https://d2l.ai/index.html

[10] Deep Learning by Ian Goodfellow and Yoshua Bengio and Aaron Courville MIT Press http://www.deeplearningbook.org 2016

REFERENCES

[11] Optuna: A Next-generation Hyperparameter Optimization Framework By Akiba, Takuya and Sano, Shotaro and Yanase, Toshihiko and Ohta, Takeru and Koyama, Masanori. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining https://optuna.readthedocs.io/en/stable/

[12] Fine-Tuned DenseNet-169 for Breast Cancer Metastasis Prediction Using FastAI and 1-Cycle Policy. By Adarsh Vulli, Parvathaneni Naga Srinivasu, Madipally Sai Krishna Sashank, Jana Shafi, Jaeyoung Choi, and Muhammad Fazal Ijaz. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9025766/

[13] Very Deep Convolutional Networks for Large-Scale Image Recognition. By Karen Simonyan and Andrew Zisserman. 2015. https://arxiv.org/abs/1409.1556

[14] Understanding VGG16: Concepts, Architecture, and Performance. https://datagen.tech/guides/computer-vision/vgg16/

[15] Step-by-step VGG16 implementation in Keras for beginners. By Rohit Thakur. https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c

[16] What is a Conditional Generative Adversarial Network (cGAN)? https://datascientest.com/en/what-is-a-conditional-generativeadversarial-network-cgan

[17] Conditional GAN. Keras. https://keras.io/examples/generative/conditional_gan/

[18] How to Develop a Conditional GAN (cGAN) From Scratch. By PhD.Jason Brownlee. 2020. https://machinelearningmastery.com/how-to-develop-aconditional-generative-adversarial-network-from-scratch/

[19] A guide to convolution arithmetic for deep learning. By Vincent Dumoulin and Francesco Visin. March 24, 2016. https://arxiv.org/pdf/1603.07285v1.pdf

[20] Experiment with the MNIST dataset. https://colab.research.google.com/drive/1vn3gqr5KOgOtpEpRpbP2Jx21a_S5eWMj

[21] OpenAI’s GPT (Generative Pretrained Transformer) Models. 2021. https://chat.openai.com/

Q&A

THANK YOU FOR ATTENTION