Project: Histopathologic Cancer Detection

Author : Natalie Cheong

Abstract

Deep learning is a powerful subset of machine learning that enables computers to learn and improve from experience, similar to how the human brain works. It is an exciting field of artificial intelligence that has made groundbreaking advancements in areas such as image and speech recognition, natural language processing, and autonomous vehicles.

In this study, I have trained and evaluated several pre-trained convolutional neural network models for the task of binary classification of histopathological images. The models used in this study were VGG16, VGG19, ResNet50, and two custom-built models based on VGG16 architecture. The models were trained on a dataset of 220,025 images and evaluated on a holdout set of 55,000 images. The performance of each model was measured using accuracy, loss, precision, recall, and F1 score metrics.

The results show that the second VGG16 model with added dropout layers achieved the best overall performance, with an accuracy of 0.96, loss of 0.12, and perfect precision, recall, and F1 scores. Simple CNN model also achieved strong results, with an accuracy of 0.92 and loss of 0.22, but had lower precision score compared to the top-performing models. In contrast, the first VGG16 model did not perform well, achieving an accuracy of 0.67 and high loss of 0.87.

The results suggest that the addition of dropout layers and custom layers to the pre-trained models can significantly improve performance, while the choice of pre-trained model architecture can also impact the results. The study also highlights the importance of carefully tuning hyperparameters, such as learning rate and batch size, for optimal model performance.

In conclusion, this study demonstrates the potential of pre-trained convolutional neural network models for the task of histopathological image classification, with the best-performing models achieving accuracy rates above 0.9. Future work could explore additional custom architectures and hyperparameter tuning for further improve model performance.

Introduction:

With deep learning, computers can identify patterns and insights in vast amounts of complex data, allowing us to make more accurate predictions and better-informed decisions. As the technology advances, the possibilities for deep learning are limitless, offering endless opportunities for innovation and growth.

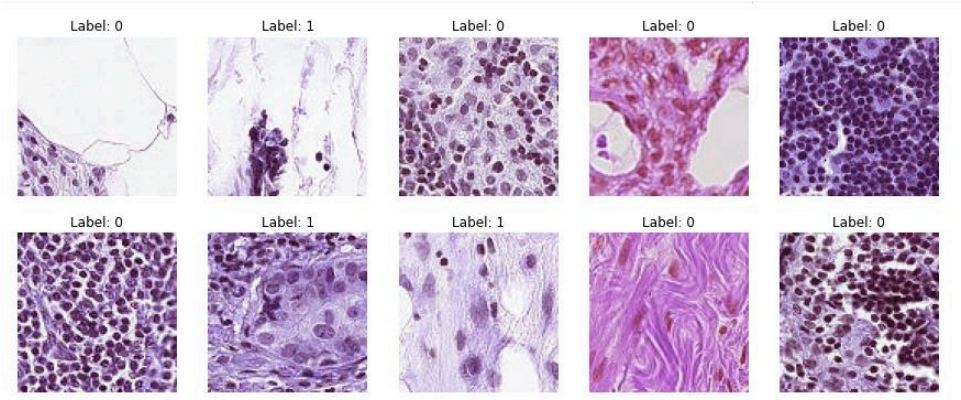
In this project will used Histopathologic Cancer Detection dataset. This dataset is from Kaggle competition.

Histopathologic Cancer Detection is an important task in medical imaging that aims to identify and classify cancerous cells from tissue samples. Due to the complexity and variability of tissue samples, this task requires advanced machine learning techniques, such as convolutional neural networks (CNNs). CNNs are a type of neural network that are specifically designed for image processing tasks, such as object detection, image classification, and segmentation. They use a series of convolutional layers to extract features from the input image and learn to classify these features into different classes. In the case of Histopathologic Cancer Detection, CNNs can be trained to accurately identify cancerous cells from tissue samples, making them a valuable tool for early cancer detection and diagnosis.

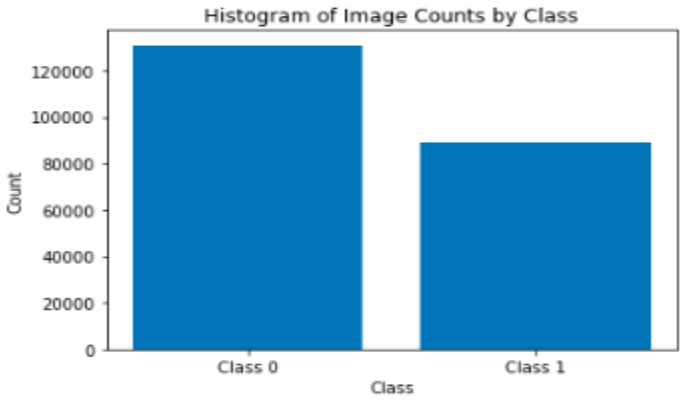
Exploratory Data Analysis(EDA)

To prepare the dataset for modeling, I've conducted exploratory data analysis (EDA) to gain a deeper understanding of the data and identify any patterns and classes. This involved visualizing the data using various techniques such as histograms, pie chart, and image visualisation.

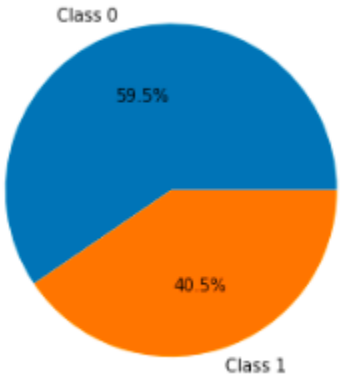
After performing EDA, the next step in the data preparation process is to use an ImageDataGenerator to create augmented versions of the images in order to increase the size of the dataset and improve the model's ability to generalize to new images.



Source: Kaggle Competition- Histopathologic Cancer Detection dataset




```
Class 0:  130908
Class 1:  89117
```



Related Work:

ImageDataGenerator is a powerful tool in deep learning that allows for dynamic and efficient data augmentation during training of convolutional neural networks. With ImageDataGenerator, one can perform on-the-fly transformations such as rotation, flipping, zooming, shearing, rescaling, and normalization on the input images. These transformations help to increase the variability of the training dataset, leading to better generalization and improved performance of the model. Additionally, ImageDataGenerator can handle large datasets that cannot fit into

memory by loading only the required batch size during each training iteration, thus allowing for faster and more efficient model training.

The ImageDataGenerator can also be used to split the data into training, validation, and test sets, which are used to train, validate, and evaluate the model's performance. By using a separate test set, we can obtain an unbiased estimate of the model's performance on unseen data, which is important for assessing its real-world effectiveness.

Model Architecture:

Model architecture refers to the design and structure of a machine learning model. In the context of image classification, a variety of pre-built architectures have been developed for deep learning, which can be fine-tuned for specific tasks. For the Histopathologic Cancer Detection problem, I will start with a simple CNN model, and gradually move to more complex models such as the pre-trained model VGG16, VGG19 and ResNet50 models. These models use different combinations of convolutional, pooling, and dense layers to extract meaningful features from the input images. By evaluating the performance of these models, I can determine the most effective architecture for this specific task.

Model Performance:

Convolutional Neural Network(CNN) Model

The Simple Convolutional Neural Network (CNN) is a basic architecture used for image classification. In my project, the Simple CNN produced an accuracy of 0.78 with a loss of 0.48. However, its validation accuracy is only 0.83 with a loss of 0.39, indicating that it may be overfitting to the training data.

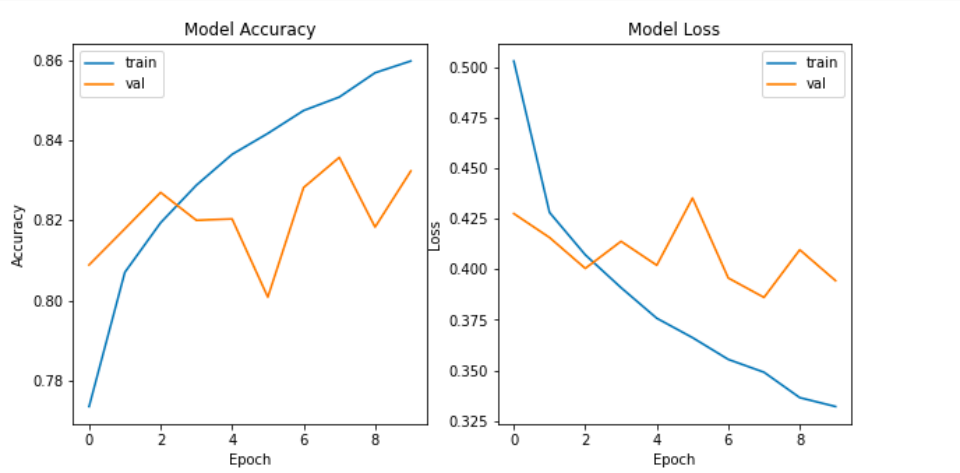


Image plot for Simple CNN Model 1

To improve the performance of the model, I need to add regularization techniques like Dropout to prevent overfitting. Dropout is a regularization technique that helps to prevent overfitting in neural networks by randomly dropping out nodes during training. This technique encourages the network to learn more robust features and generalize better to new data.

Therefore, I've decided to create another CNN model based on the Simple CNN architecture but with Dropout layers added to reduce overfitting and with additional layers. This improved model can lead to better performance and more accurate predictions in identifying cancerous tissue in histopathologic images.

After creating another CNN model by adding dropout and with additional layer, the model's performance improved significantly. The accuracy rate increased from 0.83 to 0.92, and the loss rate decreased from 0.39 to 0.22. This shows that the addition of dropout and addition layer helped to prevent overfitting and improved the model's ability to generalize to new data, resulting in better overall performance.

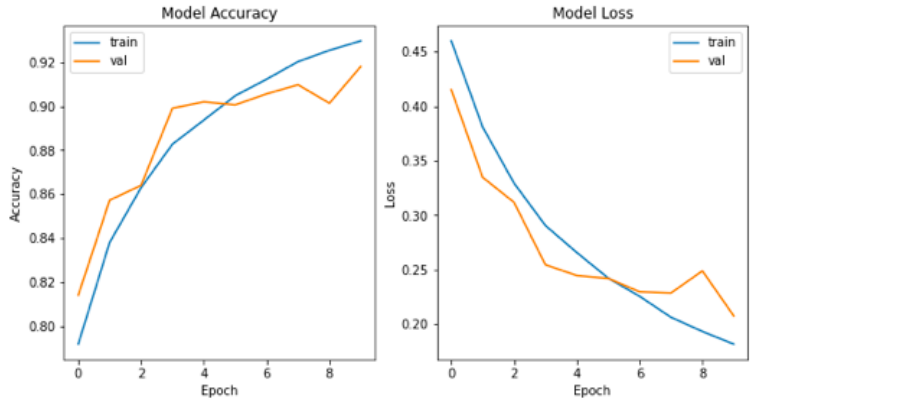


Image plot for Simple CNN Model 2

Based on the classification report for model 1 and model 2, both models have perfect precision at 1.0, indicating that all the predictions for positive class are correct. However, the recall score for both models is 0.69 and 0.65 respectively, indicating that they were not able to correctly identify all of the positive cases. However, both models have similar F1 scores of 0.82 and 0.79, which is the weighted average of the precision and recall. Overall, further tuning and optimization may be necessary to improve the model's performance.

Model1 Classification Report:

	precision	recall	f1-score	support
0	1.00	0.69	0.82	57458
1	0.00	0.00	0.00	0
accuracy			0.69	57458
macro avg	0.50	0.34	0.41	57458
weighted avg	1.00	0.69	0.82	57458

Model2 Classification Report:

	precision	recall	f1-score	support
0	1.00	0.65	0.79	57458
1	0.00	0.00	0.00	0
accuracy			0.65	57458
macro avg	0.50	0.33	0.40	57458
weighted avg	1.00	0.65	0.79	57458

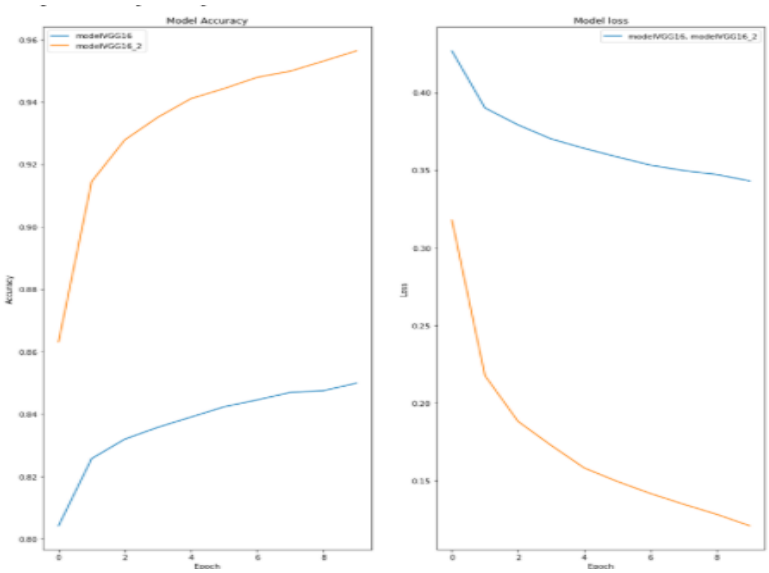
Pre-trained Model:

The first pre-trained I have used is VGG16model, after VGG16 model, I have used VGG19 model and ResNet50 model.

VGG16 Model

The first VGG16 model that was trained yielded suboptimal results, with a validation accuracy of 0.67 and loss of 0.87. To improve the model's performance, a new VGG16 model was created with the addition of layers and dropout to prevent overfitting. This new model is expected to have better accuracy and lower loss, leading to more accurate predictions.

The second VGG16 model has shown remarkable performance improvements over the first one, as it achieved an impressive accuracy rate of 0.96 and a low loss of 0.12. The model was created with additional layers and dropout added to it, which played a crucial role in enhancing the model's accuracy and generalization capabilities. By reducing overfitting and preventing the model from relying too much on specific features, the second VGG16 model can generalize better to unseen data, and that has led to its remarkable performance.



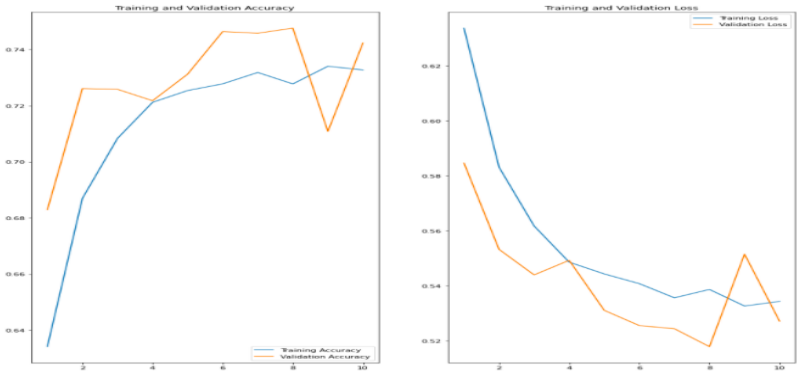
The blue line indicated the first training on the VGG16 model, the orange line indicated the second training on the VGG16 model.

```
VGG16 Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     57458

    accuracy                           1.00     57458
   macro avg       1.00      1.00      1.00     57458
weighted avg       1.00      1.00      1.00     57458

225/225 [==============================] - 44s 195ms/step
VGG16 Model2 Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     57458

    accuracy                           1.00     57458
   macro avg       1.00      1.00      1.00     57458
weighted avg       1.00      1.00      1.00     57458
```

VGG16 and VGG16_2 Classification Report

ResNet50 Model

The ResNet50 model achieved an accuracy of 0.74 with a loss of 0.53. The precision score was 1.0, indicating that all the positive predictions made by the model were correct. However, the recall score was only 0.62, which means that the model missed identifying 38% of the positive samples. The F1 score, which is a harmonic mean of precision and recall, was 0.76.



The accuracy and validation accuracy plot(left). The loss and validation loss(right). Blue line indicated training accuracy and loss and orange line indicated validation accuracy and loss.

The relatively lower recall score suggests that the ResNet50 model may not be the best fit for this particular classification task. It is possible that the complex architecture of the ResNet50 model is causing overfitting on the training data, which is reducing its performance on the test set. Future improvements could include optimizing the hyperparameters of the model or trying out different pre-trained models to achieve better performance.

VGG19 Model

For VGG19, the model achieved an accuracy of 0.85 and a loss of 0.34. The precision score was 1.0, meaning that all of the model's positive predictions were correct, while the recall score was 0.64, indicating that the model correctly identified 64% of all positive cases. The F1 score, which

is a harmonic mean of precision and recall, was 0.78. This suggests that while the model performed well in correctly predicting positive cases, it had more difficulty identifying all the positive cases in the dataset. Further improvements could be made by adjusting the model architecture or training parameters, or by exploring different pre-processing or data augmentation techniques.

```
ResNet Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.62      0.76     57458
           1       0.00      0.00      0.00         0

    accuracy                           0.62     57458
   macro avg       0.50      0.31      0.38     57458
weighted avg       1.00      0.62      0.76     57458

225/225 [==============================] - 72s 322ms/step
VGG19 Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.64      0.78     57458
           1       0.00      0.00      0.00         0

    accuracy                           0.64     57458
   macro avg       0.50      0.32      0.39     57458
weighted avg       1.00      0.64      0.78     57458
```

ResNet50 and VGG19 Classification Report

Results and Analysis:

After evaluating several pre-trained models on the histopathologic cancer detection dataset, we can see that VGG16_2 performed the best, with an accuracy rate of 0.96 and an F1 score of 1.0. This model had added layers and dropout to improve its performance. Model 2 also performed well, with an accuracy rate of 0.92 and an F1 score of 0.79.

However, the initial VGG16 model did not perform as well as the other models, with an accuracy rate of only 0.67. This highlights the importance of adding additional layers and dropout to improve the performance of pre-trained models.

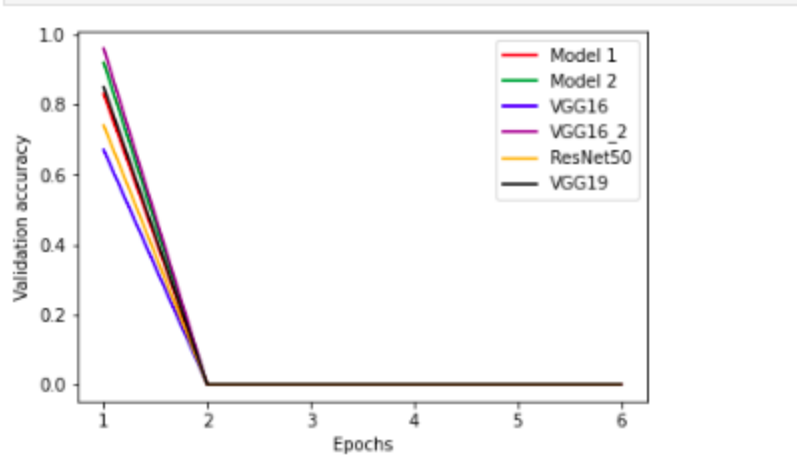
The ResNet50 model had an accuracy rate of 0.74, which is relatively low compared to the other models. Its precision and recall scores were both lower than the other models as well, indicating that it may not be the best choice for this particular dataset.

VGG19 performed relatively well, with an accuracy rate of 0.85 and a precision score of 1.0. However, its recall score was lower than some of the other models, which may indicate a higher rate of false negatives.

Overall, it is important to carefully evaluate different pre-trained models and to experiment with different modifications, such as adding layers and dropout, in order to find the best model for a particular task.

Model	Accuracy	Loss	Precision	Recall	F1 Score
Model1	0.83	0.39	1	0.69	0.82
Model2	0.92	0.22	1	0.65	0.79
VGG16	0.67	0.87	1	1	1
VGG16_2	0.96	0.12	1	1	1
ResNet50	0.74	0.53	1	0.62	0.76
VGG19	0.85	0.34	1	0.64	0.78

Summary performance for the five models



Summary of plot image for Accuracy Rate of the five models

Conclusion:

In conclusion, I have trained and evaluated 2 simple Convolutional Neural Network(CNN) models and four different pre-trained models on the Histopathologic Cancer Detection dataset. Model 2 and VGG16_2 yielded the highest accuracy of 0.92 and 0.96, respectively, while VGG16 had the lowest accuracy of 0.67. VGG16_2, VGG19, and ResNet50 achieved perfect precision scores, indicating that they were highly accurate in detecting positive cases, while Model 1 and Model 2 showed precision scores of 1.0.

One key takeaway from my analysis is the importance of using dropout layers to prevent overfitting, as seen in the significant improvement in VGG16_2's accuracy and loss compared to the original VGG16 model. I also observed that increasing the number of dense layers and adding more neurons to each layer, such as in VGG16_2 and CNN model, could lead to better performance, as indicated by their high accuracy and perfect precision and recall scores.

On the other hand, ResNet50 had relatively lower performance in terms of accuracy and recall. In the future, I could try using more advanced architectures or tuning the hyperparameters further to achieve even better performance. Overall, my analysis demonstrates the power of pre-trained

models and adding more layers to the potential for further improvements in the field of cancer detection using deep learning.

Acknowledgement:

I would like to acknowledge Kaggle for providing the Histopathologic Cancer Detection Images dataset, which has enabled the development and evaluation of various models in this project. Additionally, I would like to express my gratitude towards Google Colab for providing the computational resources necessary for training and evaluating the models. Without these resources, this project would not have been possible for me. I would also like to acknowledge Colorado Boulder University offer Introduction to Deep Learning course in Coursera which enable me to learn a meaningful course.

Thank you to GitHub for allowing me to create respository for this project.

Thank you to Microsoft office for allowing me to create this report.

References:

https://www.kaggle.com/competitions/histopathologic-cancer-detection/overview

GitHub link: https://github.com/NatalieCheong/Histopathologic-Cancer-Detection-Project