University of London

Final Project

Deep Learning Mammography Masking Level Classifier

Brent Blake
CM3070
11 March 2024

Table of Contents

1. Introduction

I've chosen the "Deep Learning on a Public Dataset" project for CM3015, with a specific focus on classifying mammograms into low, medium, and high masking levels. The project's core objective aligns with my keen interest in leveraging advanced machine learning models to enhance diagnostic tools for breast health. Utilising an open dataset I found on Hugging Face[1], my initial prototype will establish a baseline model capable of providing base level predictions for mammogram masking levels. The ultimate goal is to refine this model, ensuring high accuracy in the classification process. Through the use of deep learning methodologies learned from the book F.Chollet's Deep Learning with Python[2], including tools like Jupyter notebooks, TensorFlow, and other Python libraries, I aim to contribute valuable insights to the field of medical image analysis, supporting medical professionals in the crucial task of cancer detection and patient care.

Breast cancer has always been a global health concern, emphasising the critical need for advanced diagnostic tools to aid in early detection and improve patient outcomes. In this context, the classification of mammograms based on masking levels (breast tissue density) proves to be a critical area for exploration. The motivation behind this project stems from the potential impact on aiding medical professionals in identifying obscured tumours,not necessarily in simplifying pointing tumours out, but spotting cases of high masking levels that may conceal cancerous growth. By providing a supplementary trained robotic opinion, this deep learning-based approach aims to assist doctors in making informed decisions, prompting additional tests when necessary, and ultimately contributing to early cancer diagnosis and most importantly saving lives.

The chosen dataset, synthesised with the latent diffusion model from the paper "Generative AI for Medical Imaging: Extending the MONAI Framework,"[3] offers a unique opportunity to train and evaluate deep learning models for mammogram classification. The dataset, released under the Open & Responsible AI licence, presents three masking level labels: "Low," "Medium," and "High." The dataset's creators, Pinaya et al., have outlined the dataset's potential for enhancing AI models' training for cancer masking, aligning seamlessly with the objectives of this project.

The project's primary objective is to develop a deep learning model capable of accurately classifying masking levels in mammograms with minimal loss, false positives, and false negatives. Using convolutional neural networks (CNNs) seems to be the best choice for this task, given their efficacy in image recognition. As we progress through this preliminary report, we will detail the design, implementation, and evaluation of our deep learning model, aiming to fulfil the criteria of a high-quality and impactful project in the realm of medical image analysis.

2. Literature Review

Breast cancer screening is a critical aspect of women's healthcare, and recent research has been dedicated to improving the accuracy and efficiency of mammography. Three notable projects contribute valuable insights into this domain.

1. Title: A deep learning method for classifying mammographic breast density categories[4]

Authors: Aly A. Mohamed, Wendie A. Berg, Hong Peng, Yahong Luo, Rachel C. Jankowitz, Shandong Wu

Published in: American Association of Physicists in Medicine (https://doi.org/10.1002/mp.12683)

This paper presents a thorough investigation into using deep learning techniques to classify mammographic breast density, focusing on distinguishing between the challenging categories of "scattered density" and "heterogeneously dense". The authors give a comprehensive overview of the current landscape of breast density assessment, explaining the importance of accurate classification for risk prediction and clinical decision-making in breast cancer screening.

The paper to me has some notable strengths. Firstly, it gives a great new age approach to addressing a significant clinical need by proposing a deep learning-based solution for enhancing breast density classification, which is currently subject to subjective interpretation and variability. Secondly, the authors use a good sizable dataset of 22,000 digital mammogram images obtained from the normal clinical practice, which will improve the robustness and generalisation of their end model. Lastly, the exploration of transfer learning from a pretrained model on ImageNet to the specific task of breast density classification offers valuable insights into the potential applicability of prelearned knowledge in medical imaging tasks.

The paper also seems to have certain weaknesses to consider. Firstly, the reliance on data from a single institution for model training may restrict the generalisation of the model. Secondly, the subjective nature of the labels for breast density assessment introduces a little degree of uncertainty into the evaluation process, potentially impacting the reliability of the results. Lastly, similarly the absence of definitive exact truth labels also hampers the ability to validate the accuracy of the deep learning model effectively.

Overall, the paper makes a significant contribution to the field of breast cancer screening by proposing a good deep learning-based approach to improve breast density classification. Despite some limitations, such as the subjective nature of ground truth and the reliance on data from a single centre, the thorough methodology and evaluation lead the project to some excellent results.

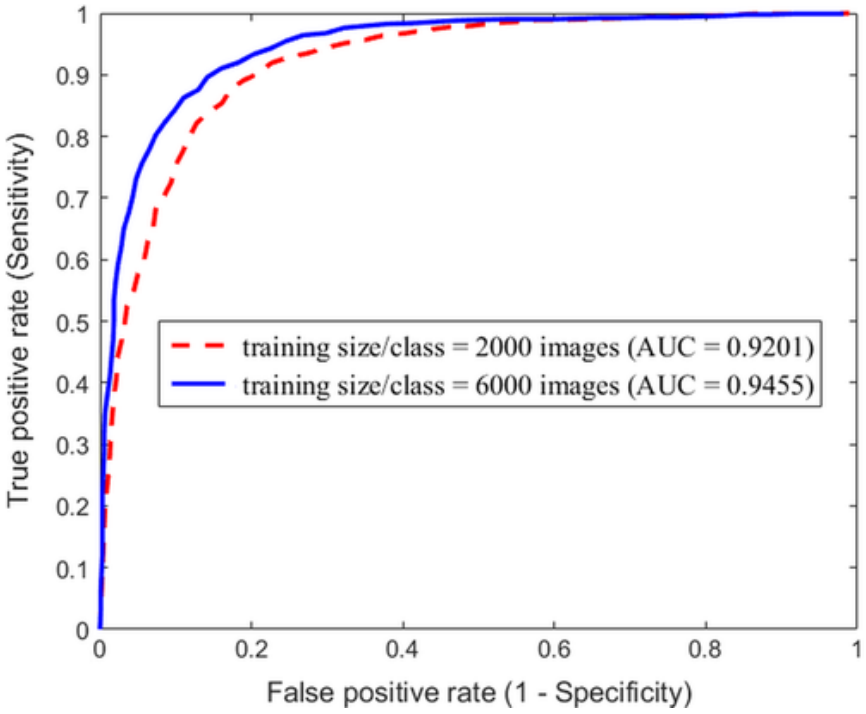


Figure 2 above from the paper shows the results of their impressive cnn model trained from scratch.

2. Title: Quantification of masking risk in screening mammography with volumetric breast density maps[5]

Authors: Katharina Holland, Carla H. van Gils, Ritse M. Mann, and Nico Karssemeijer

Published in: National Library of Medicine
(https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5332492/)

This paper addresses a crucial issue in breast cancer screening: the detection of interval cancers, which are cancers diagnosed between screening rounds. It explores the relationship between breast density and masking effect, aiming to identify women at high risk of interval cancers using quantitative measures.

The study also shows several strengths. Firstly, it suggests an approach of proposing quantitative masking risk measures to identify women at high risk of interval cancers, and that gives a fresh perspective on enhancing screening accuracy. Secondly, the research uses a rigorous methodology, utilising a substantial dataset of digital mammograms and implementing automated masking risk measurements. This careful approach enhances the robustness of the end result. Lastly, I heavily agree with the study's prediction that quantitative measures may surpass visual assessment in identifying women at risk of interval cancers, potentially leading to more tailored screening strategies and improved patient outcomes.

The study also presents a few weaknesses. Firstly, it also has a reliance on data from a single screening unit in the Netherlands and which as previously mentioned raises concerns about the generalisation of the findings to other populations or different screening programs. Lastly, the

utilisation of BI-RADS density assessments from a single radiologist can also introduce variability in density categorisation, reducing the reliability of comparisons with automated measures.

Overall this study offers valuable insights into the complex relationship between breast density and interval cancer risk. While the study demonstrates methodological rigour and clinical relevance, it also acknowledges limitations that may need further investigation. Overall, the findings contribute to advancing the understanding of breast cancer screening and lay a good possible foundation for the future research in personalised screening strategies.

3. Title: Prediction of Cancer Masking in Screening Mammography Using Density and Textural Features[6]

Authors: James G. Mainprize PhD, Olivier Alonzo-Proulx PhD, Taghreed I. Alshafeiy MD, James T. Patrie MS, Jennifer A. Harvey MD and Martin J. Yaffe PhD

Published in: Science Direct (https://doi.org/10.1016/j.acra.2018.06.011)

This paper presents a contribution to the field of breast cancer screening by addressing the challenge of mammographic density and how it can affect diagnostic accuracy, which is the same thought through the previous papers. The study focuses on the need to identify women with high masking risk for alternative screening methods. By investigating various models to predict masking status based on both biometric and image-based parameters, the study demonstrates a comprehensive approach to addressing this complex issue.

A notable strength of the study lies in its methodology, which includes the measurement of quantitative volumetric breast density, BI-RADS density, and the analysis of texture metrics. The use of stepwise multivariate logistic regressions to select predictive parameters enhances the effectiveness of the analysis. The evaluation of model accuracy using the area under the receiver operator characteristic curve (AUC) also gives a good robust assessment of the predictive power of the models.

The results indicate that the optimal model, incorporating texture metrics along with traditional density measures, outperforms density alone in predicting masking risk. This finding suggests the potential utility of texture metrics in guiding a stratified screening strategy, which could lead to more effective early detection of breast cancer.

However, my main issue with this project in particular is even though it does take into account of BI-RADS density and texture metrics to conclude a masking risk, it requires a thorough investigation on a patient for each test. It also does not seem to make use of deep learning with a CNN, which I think is very valuable in being able to fit seamlessly in the normal mammogram investigation process while also capable of being extremely powerful, given the right training process. I believe this is where I will largely differ in my project in trying to make it seamless yet powerful, only relying on the mammogram at hand.

In conclusion, the paper makes a valuable contribution to the understanding of mammographic density and its implications for breast cancer screening. By highlighting the potential of texture metrics to improve some predictive models for masking risk, the study opens avenues for future research aimed at refining more thorough screening strategies and enhancing early detection efforts.

4. Title: Lungs Disease prediction using Medical Imaging with Implementation of VGG, Resnet and Convolutional Neural Network[7]

Author: Nitish Raj Pathak

Published in: Medium - Analytics Vidhya Community (https://medium.com/analytics-vidhya/lungs-disease-prediction-using-medical-imaging-with-implementation-of-vgg-resnet-and-183e73b85df9)

This literature provides an overview of CNNs and their application in medical imaging, specifically in the context of diagnosing lung diseases such as pneumonia. It begins by explaining the fundamental concepts of CNNs, highlighting their significance in image recognition and classification tasks. The inclusion of popular CNN architectures like VGG-16 and ResNet50 adds depth to the discussion, offering insights into their structures and capabilities.

The main appeal of the paper to me lies in its use of VGG-16 and ResNet50 in the classification of lung diseases demonstrates the versatility of these models in medical image analysis, these models are pre built/trained off of a huge amount of data from imagenet and are extremely good at classifying images, this means coming with pre trained weights. Being able to use this great model, without building the wheel again, is a great idea to possibly train a model to classify medical scans, as I will attempt in my project.

However, the paper could benefit from a more detailed discussion on the implementation process, including the selection of hyperparameters and optimization techniques. Providing some useful insights into the decision-making process behind model development would enhance the reproducibility and transparency of the study.

Overall, the paper offers a valuable introduction to the application of pretrained classification CNNs in medical imaging, particularly in the context of lung disease prediction. However, the literature is rather surface level and only covers the basics of using said pretrained models and does not go much into detail into getting the most out of said models.

```
r = model.fit_generator(
  train_set,
  validation_data=val_set,
  epochs=5,
  steps_per_epoch=len(train_set),
  validation_steps=len(test_set)
)
```

```
WARNING:tensorflow:From /tensorflow-1.15.0/python3.6/tensorflow_core/python/ops/math_grad.py:1424: where (from tensorflow.python.ops.array_ops) i
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1033: The name tf.assign_add is deprecated. Pl

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1020: The name tf.assign is deprecated. Please

Epoch 1/5
163/163 [==============================] - 72s 439ms/step - loss: 0.1637 - acc: 0.9454 - val_loss: 0.1153 - val_acc: 0.9375
Epoch 2/5
163/163 [==============================] - 61s 376ms/step - loss: 0.0537 - acc: 0.9795 - val_loss: 0.2456 - val_acc: 0.8750
Epoch 3/5
163/163 [==============================] - 62s 379ms/step - loss: 0.0379 - acc: 0.9881 - val_loss: 0.1988 - val_acc: 0.8750
Epoch 4/5
163/163 [==============================] - 63s 385ms/step - loss: 0.0202 - acc: 0.9933 - val_loss: 0.2458 - val_acc: 0.8750
Epoch 5/5
163/163 [==============================] - 62s 381ms/step - loss: 0.0156 - acc: 0.9964 - val_loss: 0.2854 - val_acc: 0.8750
```

Author's decent result of 0.875 accuracy and 0.2854 loss from simply using a pretrained VGG model.

3. Project Design

I am utilising the "Deep Learning on a Public Dataset" template, and my approach aligns with the deep learning methodologies presented in F. Chollet's "Deep Learning with Python."

Domain and Users:
This project is heavily involved in the domain of medical imaging, with a specific focus on mammography for breast cancer screening. The primary users of this project include medical professionals, such as radiologists, oncologists, and healthcare practitioners directly involved in breast cancer diagnosis and treatment. The aim is to provide these professionals with a reliable tool that enhances the accuracy of mammographic assessments, ultimately leading to earlier and more accurate detection of breast cancer.

Understanding the needs of medical professionals is crucial in the development and deployment of this model. Radiologists, for instance, require tools that can assist them in interpreting mammograms more accurately and efficiently, reducing the risk of missed or delayed diagnoses. Oncologists rely on accurate diagnostic information to plan and provide appropriate treatment strategies for patients. Therefore, the model should be designed to seamlessly integrate into their existing workflow, providing valuable insights and support without adding extra unneeded complexity.

To ensure the accessibility and usability of the model, considerations must be made regarding its deployment and integration into a medical professionals process. This may involve integrating the model into existing medical imaging software or developing a standalone application that can be easily accessed and utilised within clinical settings, as the model should be able to be simply inserted in the software taking in the medical image as an input and simply output masking levels for said scan. Additionally, training and education programs may be necessary to familiarise medical professionals with the model's capabilities and limitations, empowering them to make informed decisions based on its outputs.

The introduction of the model would involve rigorous testing and validation in real-world clinical settings. This includes conducting prospective studies to evaluate its performance in diverse patient populations and comparing its effectiveness against current standard practices. Feedback from medical professionals should be actively solicited and incorporated into iterative improvements to ensure the model meets their evolving needs and expectations.

Ultimately, the successful deployment of this model has the potential to really streamline breast cancer screening practices, improving patient outcomes and reducing the burden on healthcare systems. By aligning closely with the needs and workflows of medical professionals, this project aims to make a meaningful impact in the fight against breast cancer.

Design Choices Justification:
The decision to construct a convolutional neural network (CNN) is rooted in the unique demands of medical imaging tasks. CNNs excel in discerning intricate patterns within images, proving particularly advantageous in identifying masking risk levels. The choice of grayscale input images at a resolution of 320x256 pixels aligns precisely with the typical dimensions of mammograms, ensuring relevance to the specific domain. While transfer learning is currently not planned to be implemented for the final product, it will be used as a benchmark to test against, and possibly a potential route I could take for

exploration in future iterations for improvement, offering the possibility of leveraging pre-trained models for enhanced training efficiency.

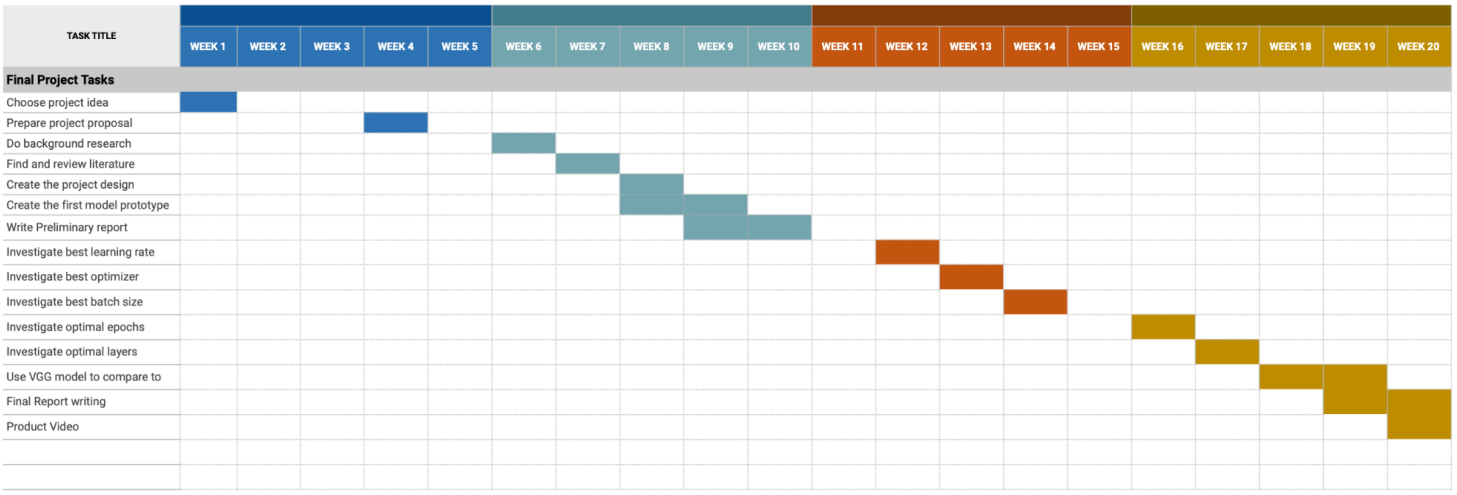
Overall Project Structure:
The project unfolds systematically, commencing with crucial data handling processes like image resizing and conversion to grayscale. This leads to the implementation of a CNN architecture, featuring convolutional and pooling layers for hierarchical feature extraction. The subsequent phases encompass model training, evaluation, and many iterations of fine-tuning to optimise overall performance.

Technologies and Methods:
In my project, I utilised a range of popular technologies to develop and refine my deep learning model for mammographic masking level classification. Python served as the primary language in a jupyter notebook format, with TensorFlow enabling the seamless and easy building/saving/loading of deep learning neural networks. Keras provided a flexible interface for model development and training, while Pandas was used for the simple data manipulation in preprocessing and matplotlib setup. Matplotlib was used many times for visualisation of the results. For the image processing, I used PIL. Git ensured version control, with GitHub for remote storing of the project versions, and Google Colab for attempted remote development. This comprehensive toolkit streamlined my workflow and helped with efficient model training and evaluation across various computing environments.

Work Plan:
Throughout the project, I tracked and planned my progress using a Gantt chart, organising tasks on a week-by-week basis. This method provided a clear visual representation of the timeline, allowing for efficient allocation of time. By breaking down the project into manageable weekly tasks, I maintained a steady pace of progress and ensured that deadlines were met.

Final Project - Gannt Chart

DATE 11/03/2024



Testing and Evaluation:
In preparation for testing and evaluating the deep learning model for mammographic masking level classification, a methodical approach was employed. The evaluation strategy followed multiple stages, starting with the meticulous selection and preprocessing of the dataset. Subsequently, the base prototype model was developed, and extensive hyperparameter tuning was conducted to optimise its performance. To validate the model's efficacy, a comprehensive suite of metrics, including accuracy, F1 score, precision, and recall, was used. Also, confusion matrices were used to gain insights into the model's performance across different masking levels. To ensure generalizability, the dataset was partitioned into training, validation, and testing sets, using cross-validation techniques. Also, various hyperparameters such as learning rate, optimizer, batch size, epochs, and model architecture were systematically investigated through experimentation. This deep testing and evaluation methodology provided comprehensive insights into the model's performance, facilitating informed decision-making regarding its potential deployment in real-world clinical settings. Lastly as a coup de grace, the best model iteration was compared to a popular pretrained VGG model in an attempt to evaluate how well the custom built model compares.

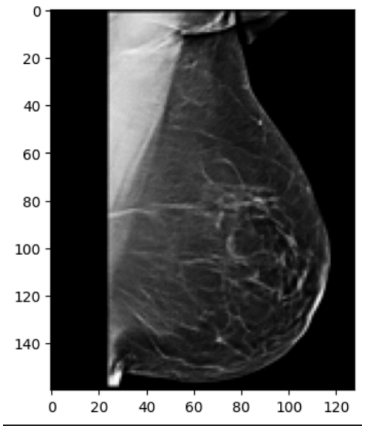
4. Implementation and Evaluation

Project Overview:
The project choice of course is the ‘deep learning on a public dataset’ choice and it revolves around the implementation of a convolutional neural network (CNN) for predicting masking risk levels in mammography images. The goal is to enhance breast cancer screening accuracy, particularly in identifying masking risk, which refers to situations where certain breast tissues may hide potential cancers.

Implementation Details:
The project employs Python as the primary programming language, utilising TensorFlow and Keras for deep learning model development. The dataset comprises mammographic images, loaded and preprocessed(reduce size and grayscale) from PNG files. A CNN architecture is employed, the base/prototype model starts out consisting of one convolutional layer followed by a 2d max-pooling layer, then flattened and then culminating in a dense layer (all subject to change for fine tuning) with softmax activation for predicting three masking risk level classes. Challenges during implementation included ensuring proper data preprocessing and lacking the computational resources to work at a vast scale. To overcome these, rigorous resizing, grayscale conversion, and having to tune the models settings to viable options due to memory and processing limitations. The model is trained using the Adam optimizer and sparse categorical cross-entropy loss over one epoch to begin with as the base model.

Training a Base Model for a starting point:
Firstly of course the 99999 images and label data is loaded in, each image is reduced in size to 128x160 pixels and turned into grayscale, this reduces the amount of computation needed when training the model drastically, the colour is completely unneeded for training and reducing the size really improves training speed and requires much less memory, and it does not really reduce the models efficacy. You could argue that reducing the quality of the image will reduce overfitting and improve generalisation too. Now importantly once the data is processed, we then firstly split all the

data where 80% is for training purposes and 20% is for purely testing, now that 80% training gets split again to the total data split being 20% for training validation, 60% for training and 20% for testing.



Eg. mammogram with low masking

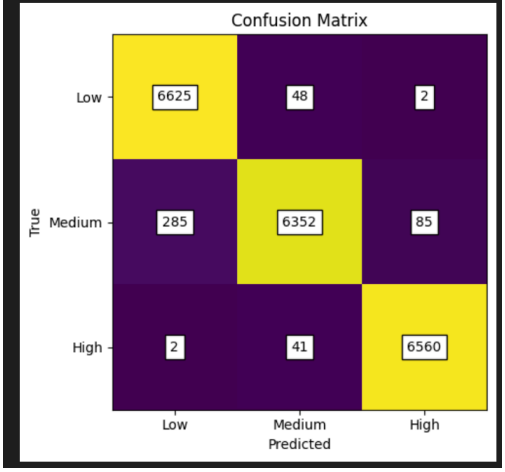
```
#initial split of the data 80% train, 20% test
X_train_temp, X_test, y_train_temp, y_test = train_test_split(images, labels, test_size=0.2, random_state=42)

#further split the training data into 75% train, 25% validation
X_train, X_val, y_train, y_val = train_test_split(X_train_temp, y_train_temp, test_size=0.25, random_state=42)

#clear unused variables for memory
del X_train_temp
del y_train_temp
del images
```

Now that the data is processed and ready, I begin by creating a simple base model where I can start from and improve on. As previously mentioned this model consists of firstly the input layer (which takes in the pixel brightness for each pixel), a convolutional layer, a 2d max-pooling layer, a flatten layer, then finally the 3 node dense layer with softmax activation for classification.

```
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 160, 128, 1)]     0

 conv2d (Conv2D)             (None, 158, 126, 16)      160

 max_pooling2d (MaxPooling2  (None, 79, 63, 16)        0
 D)

 flatten (Flatten)           (None, 79632)             0

 dense (Dense)               (None, 3)                 238899

=================================================================
Total params: 239059 (933.82 KB)
Trainable params: 239059 (933.82 KB)
Non-trainable params: 0 (0.00 Byte)
```

Upon compiling this model I then can train it with parameters being 1 epoch, a batch size of 16 and the adam optimiser, all trained with the training and validation data splits. Once the fitting is complete we can then test this base model on the testing data split and inspect the base models results using the metrics of the accuracy, F1 score, precision, recall and also a confusion matrix to visualise the results more clearly.

```
625/625 [==============================] - 5s 8ms/step
Results: [0.13882388174533844, 0.9768499732017517]
Accuracy: 0.97685
F1 Score: 0.9767627858890039
Precision: 0.9771766914265317
Recall: 0.97685
Confusion Matrix:
 [[6625   48    2]
 [ 285 6352   85]
 [   2   41 6560]]
```



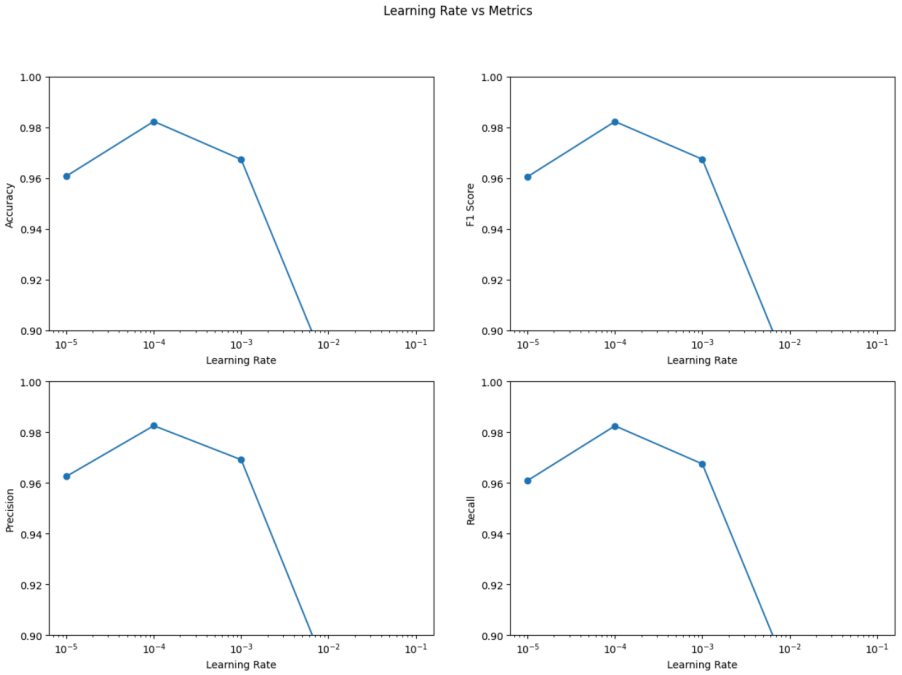
As we can see the base model's test results already seem fairly good, with all metrics being about 97.6%. Having a close look at the confusion matrix we can spot that the model does seem to be misclassifying 285 low masking levels as medium, which is less than ideal. Fortunately this is the base model which I now plan to iterate on to fine tune and find the best optimal parameters to get the greatest classification results.

The fine tuning process:
The plan to create the best possible model is by iteratively going through parameter by parameter thoroughly training and testing each and every viable parameter value and comparing each model's results and moving forward with the best model and the process then repeats for the next parameter. Below are the parameters that will be tested in that respective order and the best model will move forward and be used for testing in the next parameter and so on:

1. Learning rate (0.1, 0.01, 0.001, 0.0001, 0.00001)
2. Optimiser (sgd, rmsprop, adagrad, adam)
3. Batch size (8, 16, 32, 64)
4. Epochs (1, 5, 10, 20)
5. Layers (1 conv and 1 dense, 1 conv and 2 dense, 2 conv and 1 dense, 2 conv and 2 dense)

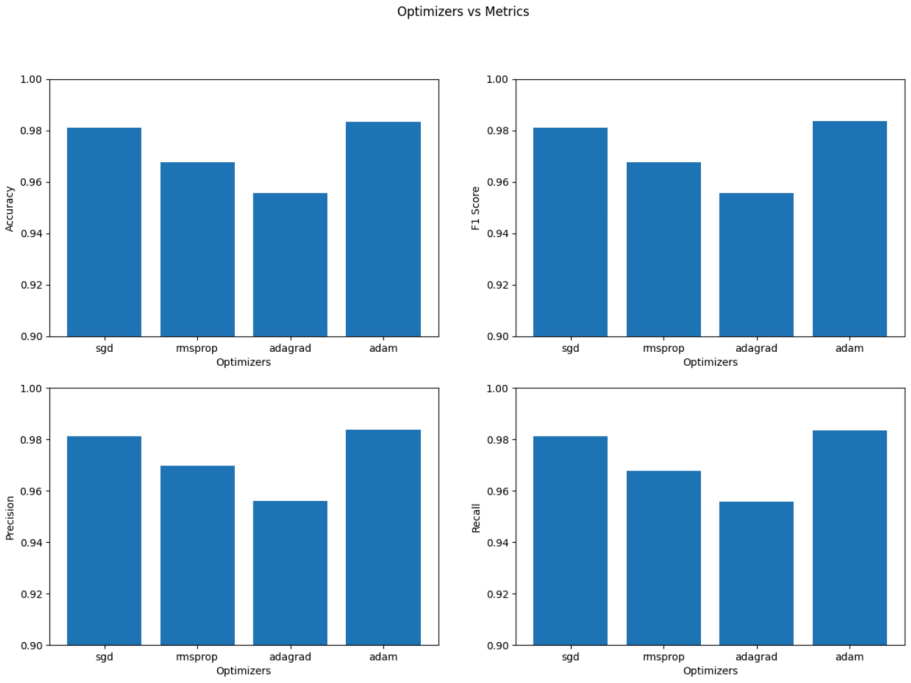
For each of these 5 parameters we will proceed with the same process as was done to train the base model using the same training and validation data; and then finally testing the trained model with the test data split. In essence a model for each parameter value will be compiled, trained, tested and the test results will be analysed to confirm the best parameter value.

Learning Rate Results:



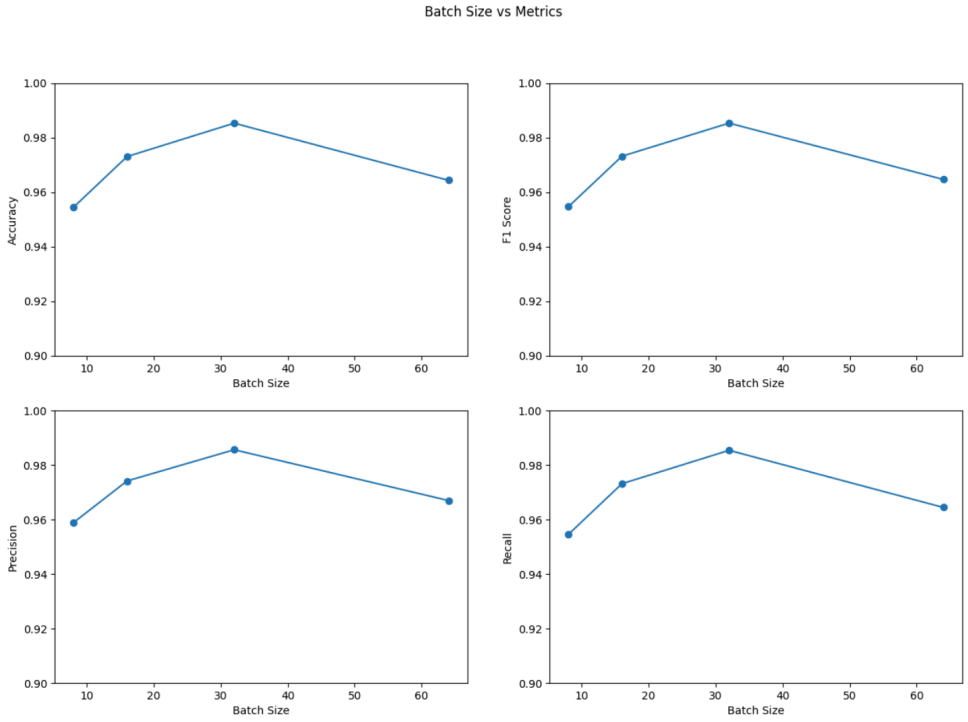
After training and testing the learning rates these are the results visualised for the test results of each learning rate. It can be clearly seen that a learning rate of 10^{-4} or 0.0001 performed best across the board. The other rates of 0.00001 and 0.001 did follow somewhat closely behind, but 0.1 was far off, unable to get past random guessing performance (0.33). We will carry the best learning rate through to the next parameter testing.

Optimiser Results:



For the optimisers there results seems to show 2 standout choices of sgd and adam, however in all metrics adam does slightly edge out sgd, meaning I will stick with adam as the optimiser for the rest of model training.

Batch Size Results:

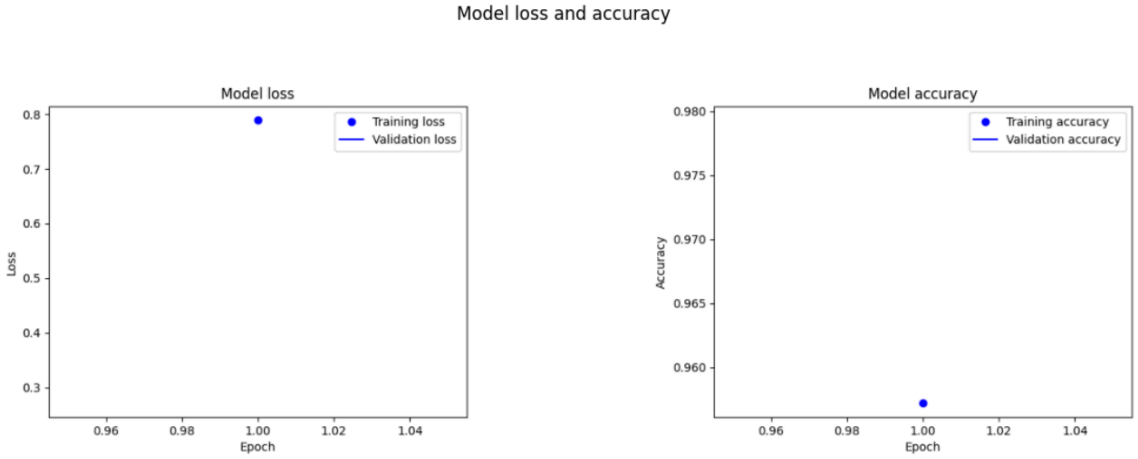


The batch size results gave a great looking curve giving me a really good indication that the optimal value is indeed 32 as anything lower or higher trails off in all of the metrics. These results are terrific and instil confidence in me that using a 32 batch size will be the optimal size for the rest of the project.

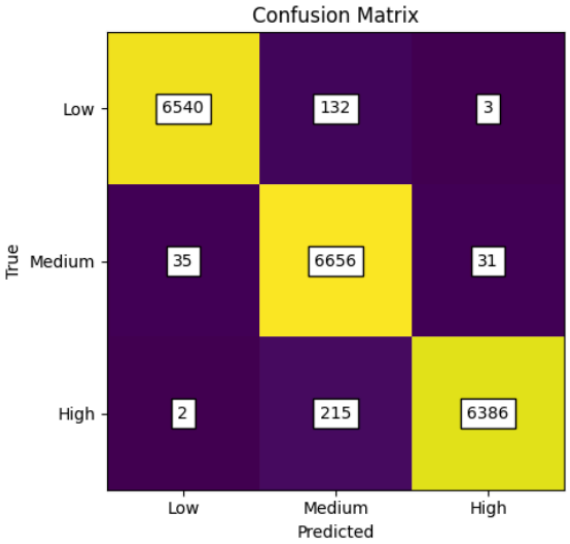
Epoch results:

The results for the epoch fine tuning are not as obvious as the previous parameters, so we will have a closer look at each of the training histories and the confusions matrices for each model to make an informed and calculated decision on the best amount of epochs for the final model.

1 Epoch

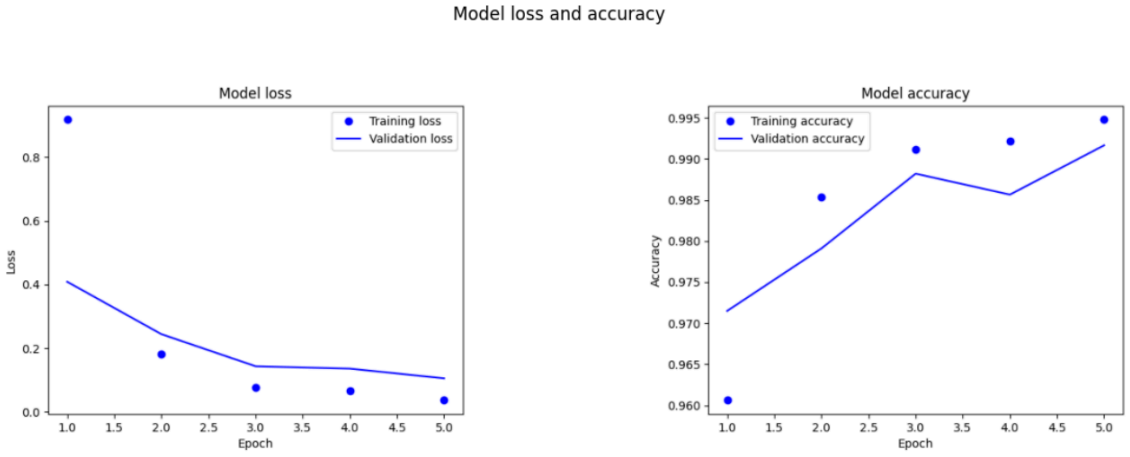


1 epoch training history

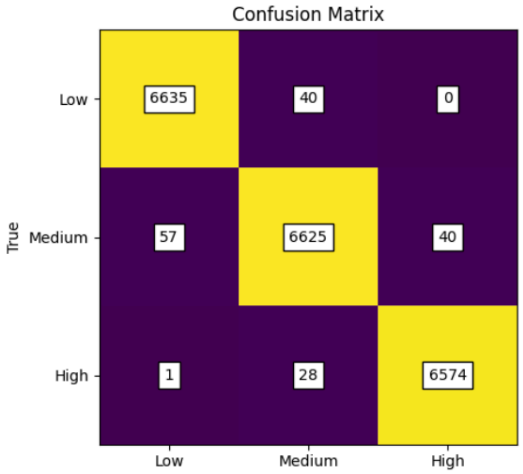


1 epoch confusion matrix

5 Epochs



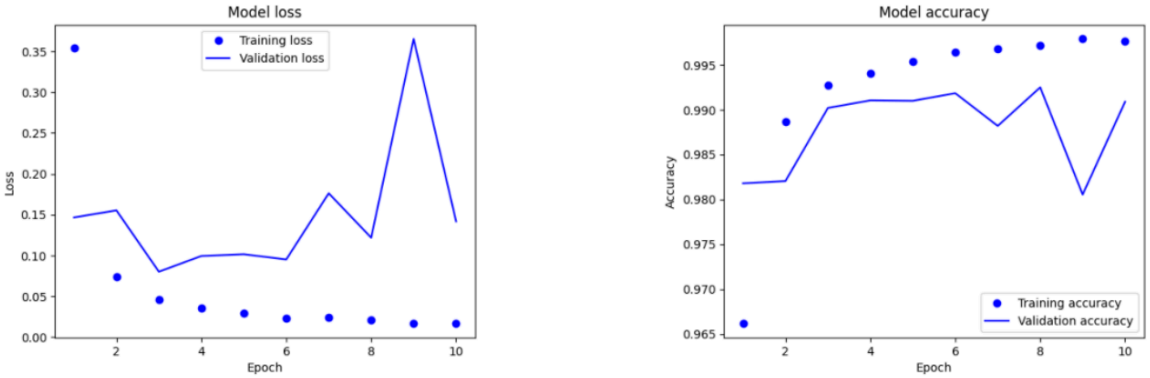
5 epoch training history



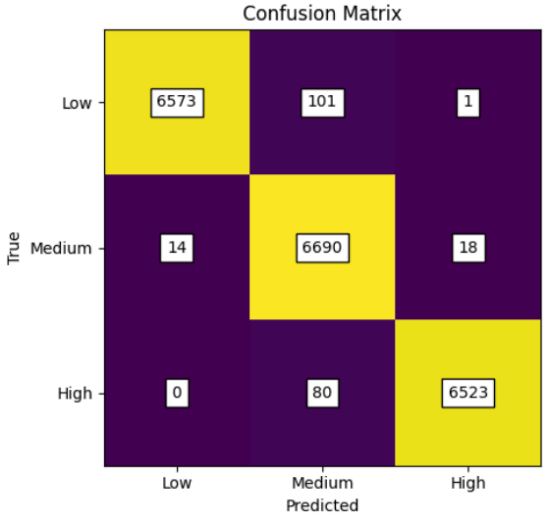
5 epoch confusion matrix

10 Epochs

Model loss and accuracy



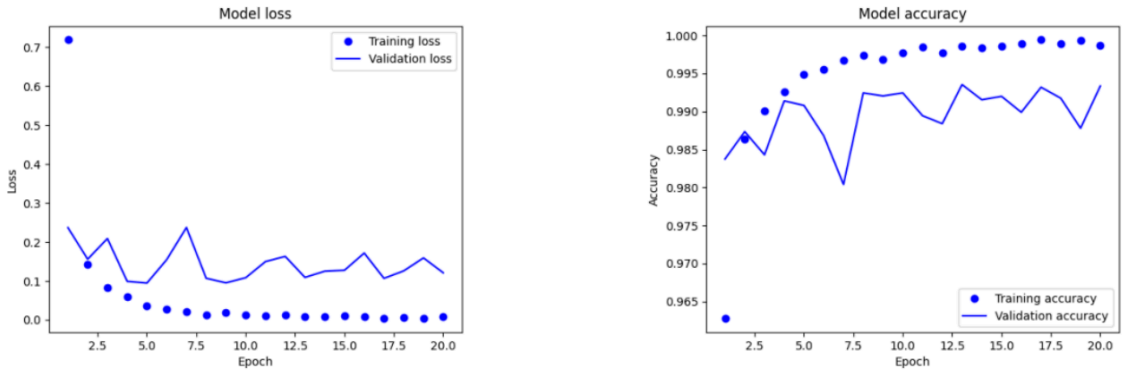
10 epoch training history



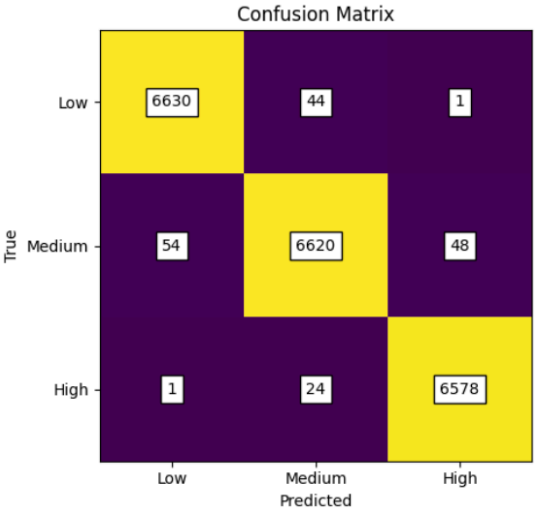
10 epoch confusion matrix

20 Epochs

Model loss and accuracy

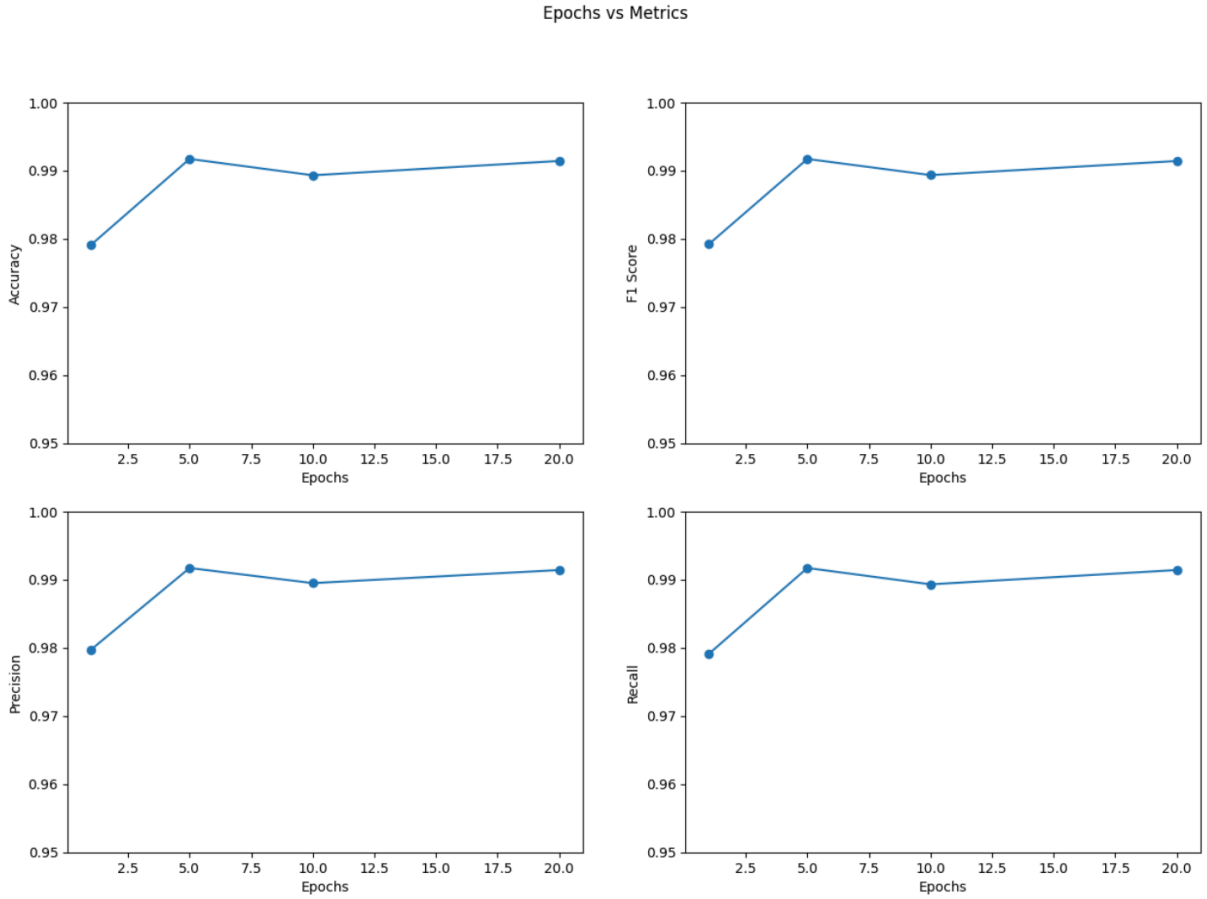


20 epoch training history



20 epoch confusion matrix

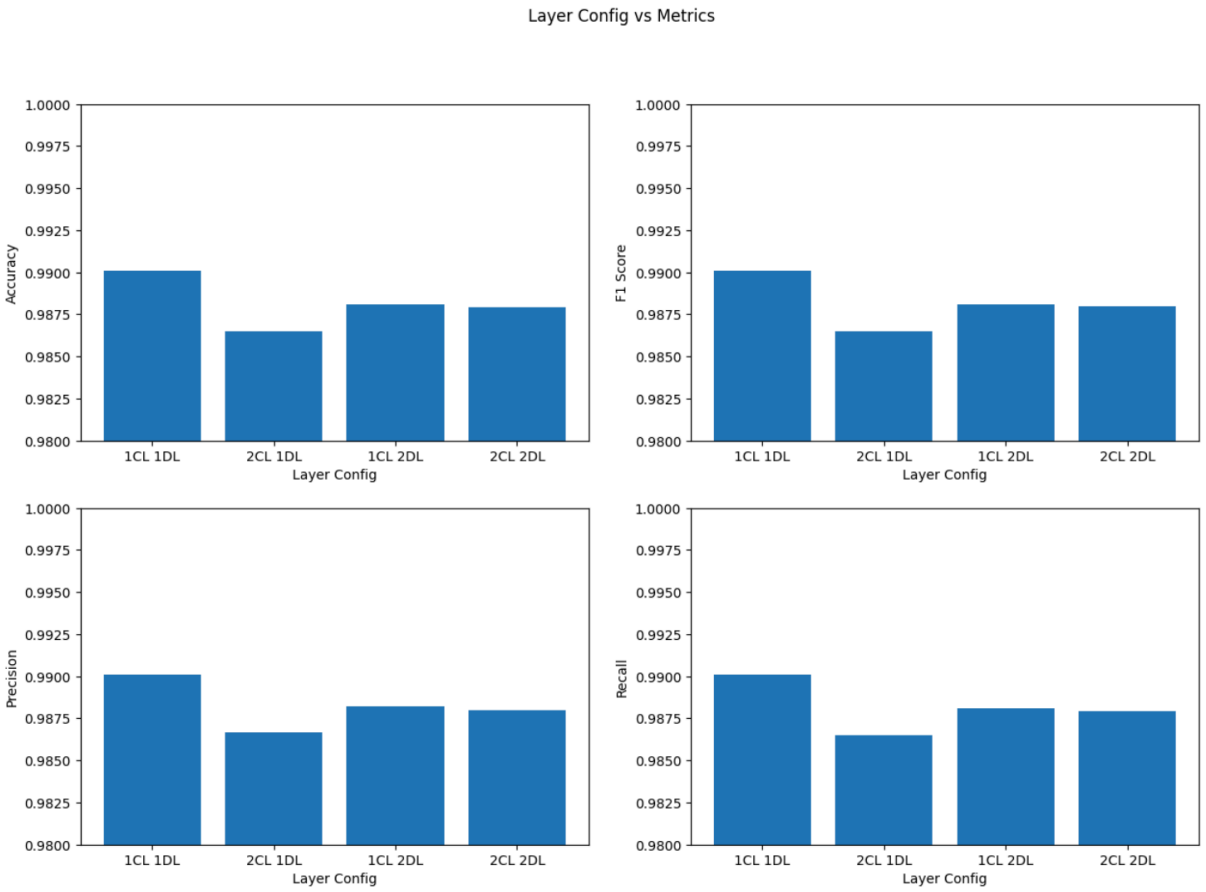
Overall Epoch Results:



Now with all the metrics and results in view and analysed, I firstly took notice that clearly 1 epoch was not enough. Also it became very clear to me that after about 5 epochs the result metrics had a much greater increase from this parameter change compared to any of the other previous parameters, looking at about at least 1% shift in all metrics. The reason the choice of best epoch is not so obvious is due to the metrics showing the results seem to plateau and stagnate after the 5 epoch mark, but they are not necessarily getting worse. As we can see the 5 and 20 epoch metric results seem to show similar percentages making the call a little more difficult. However upon analysing both the training

histories and confusion matrices, I believe 5 epochs would be the best amount. The history for the 5 epoch show continual and constant improvement the entire way through, without any decline or plateau in validation accuracy at all, but the larger epoch values all seem to show now real improvement in validation after 5 epochs, making me believe there is slight overfitting that is occurring when going over 5 epochs. So to make sure the model end result has good generalisation and as little overfitting as possible, I will choose 5 epochs as the training parameter for future models.

Layer Configuration results:



With the models final parameter tuning it can be seen that increasing the amount of layers for more than 1 convolutional layer and 1 dense layer has no improvement on the model and slightly reduces its accuracies. Meaning the final model will have 1 conv layer and 1 dense layer in it.

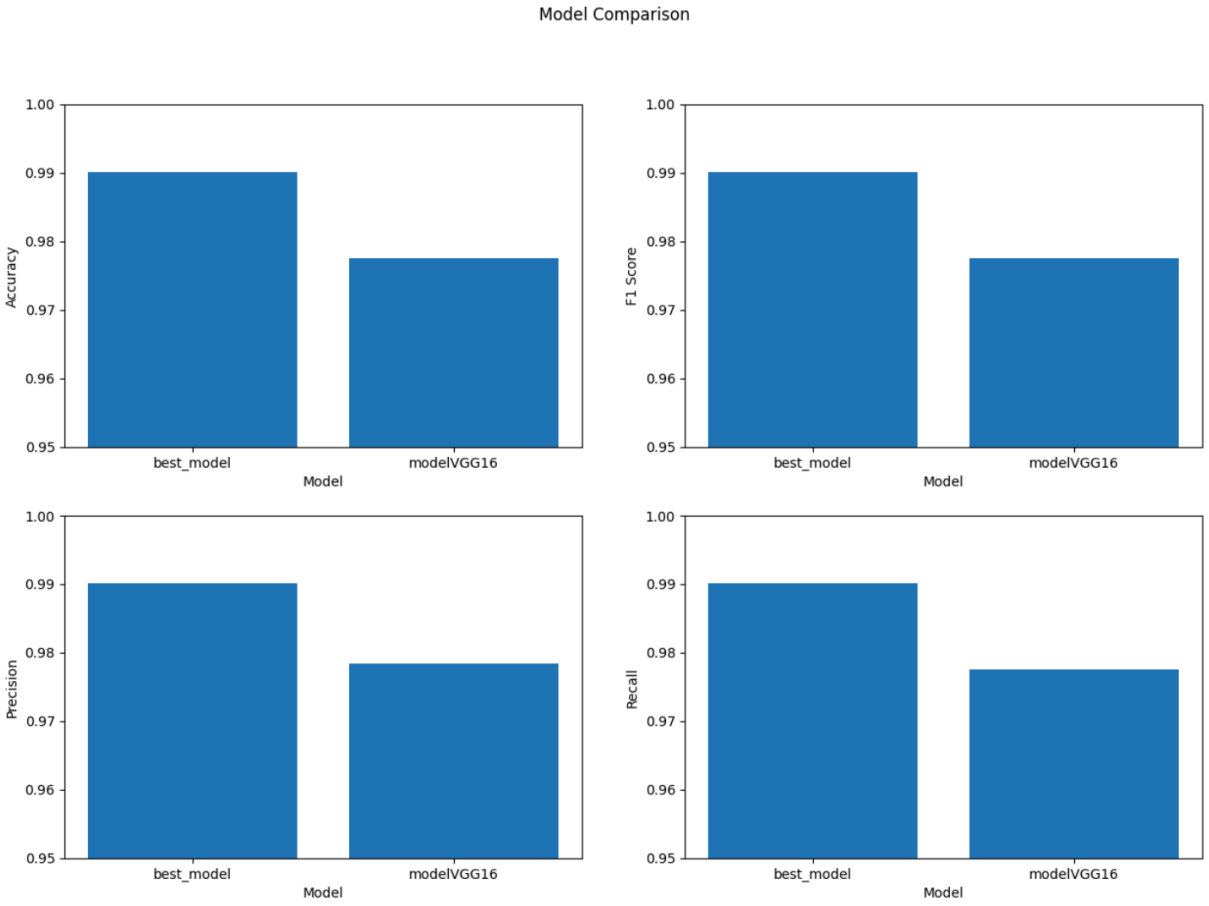
The optimal fine tuned model:

After all the iterative training and testing for each possible and viable parameter setting in my deep learning model, I have ended with a model configuration that has:

- A learning rate of 0.0001
- The adam optimiser for training
- A training batch size of 32
- 5 epochs of training
- 1 convolutional layer and 1 dense layer

The final evaluation against a pretrained VGG model:

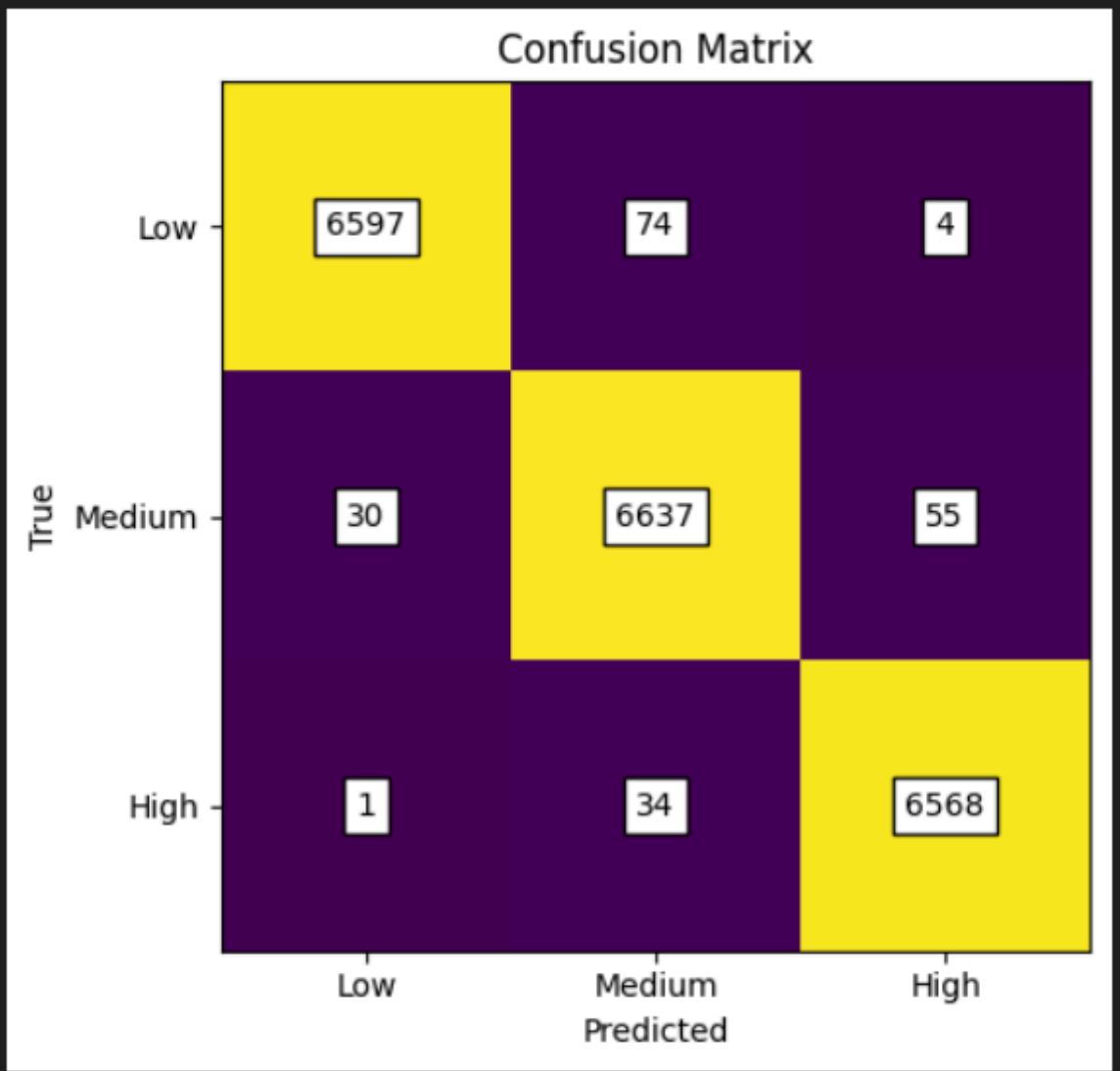
For a final good evaluation of my model, I wanted to compare my model to the keras vgg16 pretrained model with its model weights trained to the imagenet dataset. I planned to fit this model with the same data used to train my best model. However in researching and learning how to use and configure the VGG model, I had made the discovery that it in order for it to work optimally it needed to take in RGB pixel values and images of the dimension 224x224, this means I needed to preprocess my image data again, essentially tripling the amount of memory needed by adding in colour values and also increasing the amount of pixels put in quite significantly. So I did in fact preprocess the images to fit the models needs and did try and fit it with the same amount of images my best model was trained with, however this proved impossible (as the python kernel kept crashing due to lack of memory) with the limited computational power my apple mac air was able to produce, since it does not have a dedicated GPU. I did also attempt to use google colab to train this VGG model remotely, but this also showed not very feasible for me as I live in South Africa with very low internet speeds, meaning uploading 100000 images onto the google cloud and then processing them would take a hugely unreasonable amount of time that is not in my scope. So it really should be noted that I had to reduce the training data for this VGG model to 10000 images instead of 100000 for the train, validation and test split. Once this was done the model consisting of the frozen VGG layers and then a flatten and a dense classification layer for the model to classify into the masking levels was created and trained also on 5 epochs. Finally here are the results of my best iteration of my custom built model vs the pretrained VGG model:



I was quite glad to see my custom model did outperform the pretrained VGG16 model in all metrics across the board.

The final model's test results:

```
Results: [0.13368579745292664, 0.9901000261306763]
Accuracy: 0.9901
F1 Score: 0.990102876024772
Precision: 0.9901181059577124
Recall: 0.9901
Confusion Matrix:
 [[6597   74    4]
 [  30 6637   55]
 [   1   34 6568]]
```



5. Conclusion

In conclusion, the project has successfully developed and evaluated a deep learning model for classifying masking levels in mammograms, contributing to the field of medical image analysis. The implementation of a convolutional neural network (CNN) allowed for accurate and efficient classification of breast tissue density, aiding in the early detection of breast cancer.

Throughout the project, various methodologies and techniques were employed, guided by insights from existing literature and best practices in deep learning. The project followed a systematic approach, starting with data preprocessing, model development, and hyperparameter tuning, culminating in a comprehensive evaluation of the final model's performance.

The results demonstrate the effectiveness of the custom built CNN model, achieving high accuracy, F1 score, precision, and recall rates. The iterative fine tuning process led to the identification of optimal hyperparameters, enhancing the model's accuracy and generalisation capabilities.

Also to mention, the comparison with a pretrained VGG16 model provided valuable insights into the relative performance of the custom-built model and established benchmarks. Despite computational constraints and limitations, the custom model outperformed the VGG16 model, showing the efficacy of my iterative approach in step by step improvement to get the best I could out of the model.

Overall, the project's success highlights the potential of deep learning in medical imaging applications, particularly in breast cancer screening. By using advanced machine learning techniques, such as CNNs, and making the use of open datasets and resources, the project has made small steps towards improving diagnostic accuracy and patient outcomes in breast cancer care.

Moving forward, future iterations of the project could explore additional optimizations, such as a combination learning techniques, data augmentation strategies like rotating, flipping and blurring images to prevent overfitting and improve generalisation. It would also definitely be worth attempting to combine my model with a pretrained model similar to VGG and iterate over that to find any room for improvement. Moreover, efforts to integrate the developed model into clinical workflows and conduct real-world validation studies would be essential steps towards moving the research findings into impactful clinical practice. The model is also only in its research stage and would need to be incorporated in clinical software or its own user friendly software.

In conclusion, the project represents a significant contribution to the field of medical image analysis, with the potential to revolutionise breast cancer screening practices and ultimately provide a tool in helping save lives.

6. References

1. Pinaya, W. H., Graham, M. S., Kerfoot, E., Tudosiu, P. D., Dafflon, J., Fernandez, V., ... & Cardoso, M. J. (2023). Generative AI for Medical Imaging: Extending the MONAI Framework. arXiv preprint arXiv:2307.15208. Retrieved from https://arxiv.org/abs/2307.15208
   https://huggingface.co/datasets/SinKove/synthetic_mammography_csaw

2. Chollet, F. (2017). Deep Learning with Python. Manning Publications. Retrieved from https://www.manning.com/books/deep-learning-with-python

3. CSAW-M: An Ordinal Classification Dataset for Benchmarking Mammographic Masking of Cancer. arXiv preprint arXiv:2112.01330. Retrieved from https://arxiv.org/abs/2112.01330

4. Mohamed, A. A., Berg, W. A., Peng, H., Luo, Y., Jankowitz, R. C., & Wu, S. (2017). A deep learning method for classifying mammographic breast density categories. Medical Physics, Volume, Pages. DOI

5. Holland, K., van Gils, C. H., Mann, R. M., & Karssemeijer, N. (2017). Quantification of masking risk in screening mammography with volumetric breast density maps. Scientific Reports, 7(1), 3812. DOI

6. Mainprize, J. G., Alonzo-Proulx, O., Alshafeiy, T. I., Patrie, J. T., Harvey, J. A., & Yaffe, M. J. (2018). Prediction of Cancer Masking in Screening Mammography Using Density and Textural Features. Academic Radiology, Volume, Pages. DOI

7. Nitish Raj Pathak (2020). Lungs Disease prediction using Medical Imaging with Implementation of VGG, Resnet and Convolutional Neural Network. Medium, Analytics Vidhya Community

My Github Repo for my code and model version storage:

https://github.com/brentvblake/mammography-DL-masking-level-classifier