A2 Final Report

Saneyar Khazin, Simon Bateman

Introduction

Histopathology (or histopathology) refers to the study and diagnosis diseases related to the tissues and involves observing tissues under a microscope (Royal college of Pathologists).

Histopathologists, doctors involved in the study and observation of tissues, provide a diagnostic service for cancer; they handle, and study tissues and cells removed from suspicious lumps, identify the nature of the abnormality, and categorize it as either malignant or benign. If the diagnosed as malignant, they provide information to about the type of cancer, its grade and, for some cancers, its responsiveness to certain treatments (Royal college of Pathologists).

From this, we can come to understand how critical this process can be. However, detection and classification of cell nuclei in histopathology images of cancerous tissue is a challenging task due to cellular heterogeneity. Deep learning approaches have shown some promising results on histopathology images in various studies.

In this project, we attempt to create two separate machine learning model based on Deep Neural Networks to aid in predicting i) whether based on the given images, a cell is cancerous or not and ii) the cell type of the provided image.

VGG Model

The original VGG model was an attempt at improving the original architecture of highly popular ImageNet. Some of the major changes include fixing other parameters of the architecture and steadily increasing the depth of the network by adding more convolutional layers, which is made possible by using small (3 x 3) convolution filters in all layers. A VGG model is structured to consist of many blocks, with a block consisting of: convolution + activation + pooling. These block will also downscale the images due to the use of the max pooling layers.

The models we have used in this assignment are a modified version of the original VGG Model. While in the original paper, the images sizes used were 224 x 244, our images were a lot smaller at 27 x 27. This meant we would have to make a couple of adjustments to get our model running well. Only a small amount of preprocessing is done, which includes substracting the mean RGB value, computed on the training set. The image is then passed through a stack of convolutional layers with small receptive field: 3 x 3. Convolution striding is fixed to 1 pixel for 3 x 3 convolutional layers, which translates to `padding = same`. Spatial pooling was carried out by five max-pooling layers in the original paper. But since the images we were supplied were much smaller, we used only 2-3 layers. Ofcourse, since the we were dealing with two different types of classifcation models here (cancerous or not vs cell type), compile parameters were different in both the cases. For the binary classification, we used a `sigmoid` activation function for the final layer since we were dealing

with binary values, whereas `relu` was used for classifying the cell type. The complie parameters for each of the models were:

```
model.compile(optimizer='adam',
              loss=tf.losses.BinaryCrossEntropy(from_logits=False),
              metrics=['binary_accuracy'])
```

for the `isCancerous` model and:

```
model.compile(optimizer='sgd',
              loss=tf.losses.CategoricalCrossentropy(from_logits=False),
              metrics=['categorical_accuracy'])
```

for `celltype` classification.

Spatial Constrained CNN

Sirinukunwattana *et al.* [1] uses a locality sensitive deep approach of a Neighbouring Ensemble Predictor (NEP) coupled with CNN to more accurately predict the class labels of detected cell nuclei. This process used 2 premises: the distance from the centre, and a weighted ensemble of local predictions. This process allocates higher probability to pixels closed to the centre of the nucleus that those further away. This enables the model to detect and assign class labels to multiple nuclei. Prior to the research done by Sirinukunwattana *et al.* [1], most of the cell detection methods rely on shape of nuclei as well as stability of the features for cell detection and texture of the nuclei for classification [1]. The methodology used by Sirinukunwattana *et al.* and Xie *et al.* are closely related, in which they use an SC-CNN (Spatially Constrained Convolution Neural Network). SC-CNN contains a layer that is designed to predict the centroid location of the nuclei as well as the confidence of the locations.

This used an overall image of the data allowing the neural network training to allocate priority to specific pixel in the photo. This approach did not consider the impact of varying locations of cell nuclei.

Other papers used a combination of Difference of Gaussian followed by other processing to detect nuclei, Bi and multilateral symmetric to detect nuclei. But these failed to detect thin fibroblasts and small nuclei as they replied on nuclei shaped and texture for detection. Using a basic VGG architecture did not provide greater accuracy, as was demonstrated by reducing the number of convolutions before passing the results onto a Neural Network layer.

Their approach potentially did not require the preparation of slides so the nucleus was position near the centre of the slide thus offering a more generic model.

We chose simple NLP network as the size of the images (27 x27 x 3) was small enough the total image could be processed as an input later with as this was simpler and produce good results on the data set. As a scanning tool this produced very good results with 100% recall.

Training

Comparing they trained data using rotated and flipped patches, and randomly changed colour hue and saturation.

They used SDG with training for 60 epochs with various learning rates and selected the best models based on RMS errors.

We did nuffinc and just loaded the images. And selected the errors based on binary accuracy and recall.

Initial image training was 500 x 500 pixles compared to 27x27 supplied for the assignment.

Precision, Recall, and F1 score were used to grade model – we obtained

Validation

We used random fixed they used 2-fold cross validation with (50 images/fold)

We reviewed and considrer Recall for binary classification, and F1 score.

We should maybe add table for each model – p1 and p2.

Model, Accuracy, Precision, Recall, and F1 score

Score and Evaluation

Model	Average F1 score	Recall	Accuracy
Base Model			

Model	Average F1 score	Recall	Accuracy
Base Model			

Final thoughts

One of the more surprising things we found out was that a base neural network with minimal layers and zero regularization seemed to perform quite well with the binary classification, altough it did require some work multiple classification model. While we were able to replicate the CNN model outlined by Sirinukunwattana *et al.* , we weren't able to replicate the exact algorithm used by them to classify the images. There, we based our model on the VGG model.

- Given the size of the imput VGG didn't improve results and took considerable more time to process results.
- We tried reduced VGG to reduce the overall training time but didn't improve results and took longer to train than the base model