Automatic Quantification of Lymphocytes as a Prognostic Marker in Cancer Tissue

Mahit Tanikella
Monta Vista High School
21840 McClellan Rd, Cupertino, CA 95014

Acknowledgements

I want to thank my computer science teacher Mr. Scott DeRuiter for being my mentor and guiding me throughout.

Introduction

Lymphocytes are immune cells that accumulate at sites of cancer tumor as immune response. Their count varies with tumor type and stage and is associated with how much disease has progressed. Quantifying tumor-infiltrating lymphocytes is very important for cancer research. Currently, tissue examination and cell density assessment are performed by pathologists, by manually inspecting the glass slides under the microscope. This process is very time consuming.

In this research paper, a novel way for automating quantification of lymphocytes in histopathology images of cancer tissue, using deep convolutional neural networks is developed. The network was built on top of popular, state of the art, 50-layer deep neural network called Resnet50 [1], which is trained on more than one million images. A technique known as transfer learning [2] is used where only the final few layers of the network are replaced and trained from scratch. The network was trained on google cloud with whole-slide images of breast, colon and prostate cancer, stained with CD3 and CD8 immunohistochemical markers, downloaded from Lymphocytes Assessment Hackathon website [3]. Immunohistochemistry [5] is a staining technique used to highlight cells of interest, lymphocytes in this case, in histopathology tissue samples.

The model has achieved 86.7% accuracy with 83% precision and 86% recall, with test data. Overall, F1 score is 0.8, which indicates that the model is predicting all classes. This model can be used in automation of cancer diagnostics and can aid in cancer research.

Materials and Methods

The training dataset was downloaded from Lymphocyte Hackathon Assessment website [3]. There are 20,000 patches of size 299x299 extracted at 40X magnification (~0.25 micron/px) from whole-slide images of breast, colon and prostate cancer stained with CD3 and CD8 immunohistochemical markers. Lymphocytes are visible as cells with a blue nucleus and a brown membrane in these images. In practice, a blue nucleus and a clearly visible brown rim are not always visible after immunohistochemistry. Furthermore, effects of background stain (i.e., brown color on the tissue without presence of lymphocytes), artifacts as

well as presence of ink on the tissue can make detection and quantification of lymphocytes quite challenging. Examples of patches in the pilot dataset are shown in Figure 1.

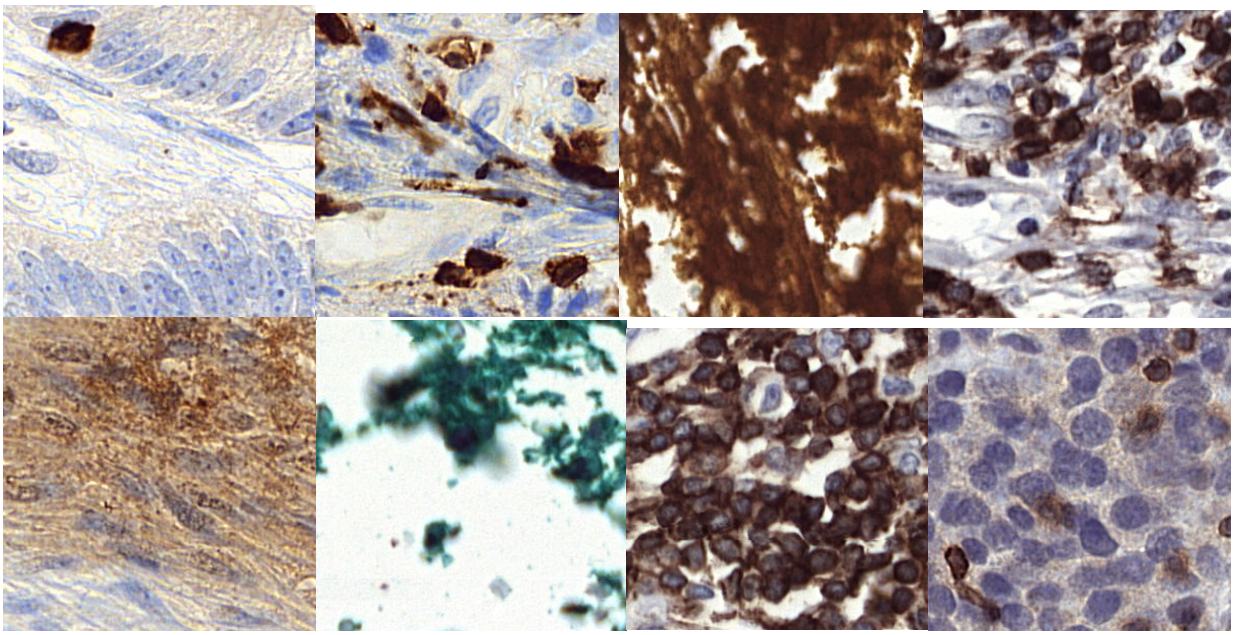

Figure 1: Sample whole-slide images of breast, colon and prostate cancer

The images are provided as HDF5 [4] file. They are read using the h5py library and saved as png image files. The count of lymphocytes and type of cancer for each image was provided in a separate CSV file. Locations of lymphocyte cells was not provided. Around 20% of patches did not contain any lymphocyte. Table 1 summarizes distribution of images across prostrate, breast and colon cancer along with their min, max and mean counts. Figure 2 shows distribution of counts.

	Min	Max	Mean	Total
Prostrate	0	55	3.7	4980
Breast	0	70	3.09	7404
Colon	0	59	2.7	7616

Table 1: Distribution of images across different cancers

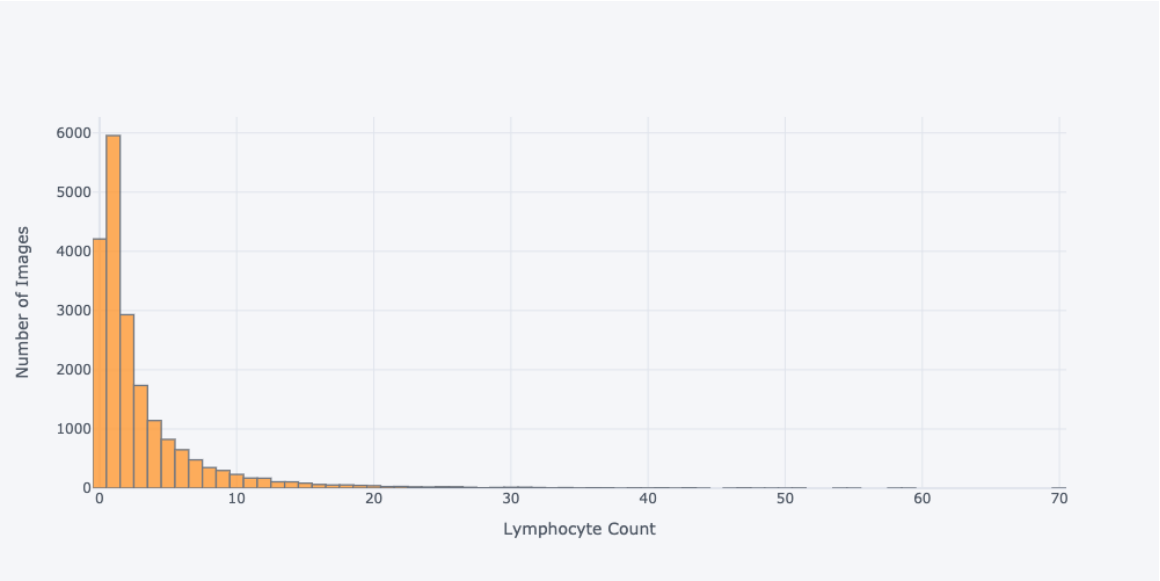

Figure 2: Distribution of lymphocyte counts

Neural network model was created and tested with sample data using Jupyter Notebook on a Mac. After basic testing is done, the network was trained on Google Cloud AI and ML [6] platform.

Convolutional Neural Networks (CNNs) with deep learning are commonly used for image classification tasks. Transfer learning is a technique where instead of training a large CNN from scratch, training is done on top of pretrained models. Training large neural networks from scratch requires very large datasets, time and compute resources. As all the extracted features i.e. edges, circles, lines etc. from images are similar, features extraction part of a pre-trained model can be reused. Keras [7], a deep learning software library, provides multiple pretrained models trained on ImageNet, a very large dataset of 1.2 million images with 1000 categories. Resnet50 model [1] is chosen for this work, as it is state of the art, most commonly used network for transfer learning. The architecture of the network is shown in Figure 3.

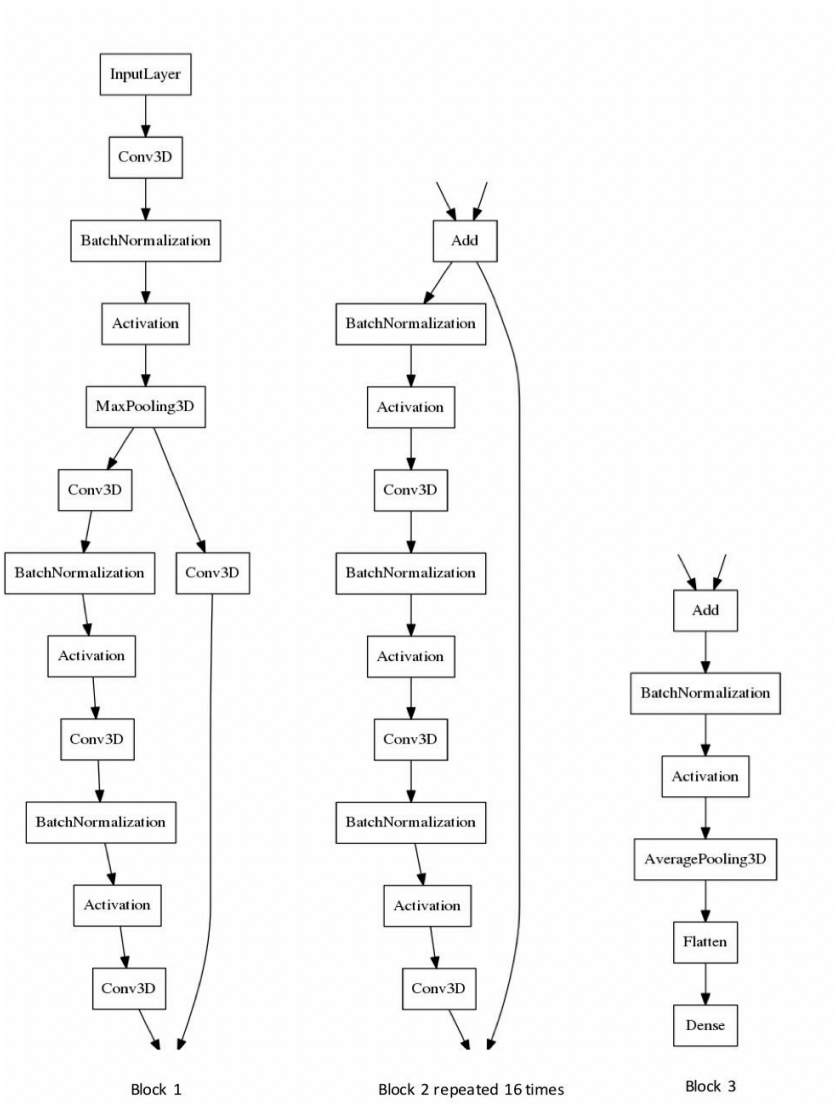

Figure 3: Architecture of Resnet50 Network

Data is classified into three categories, images which have no lymphocytes, images with one to ten lymphocytes and images with more than 10 lymphocytes. 70% of the data is used for training, 20% for validation and 10% for testing. Custom data generator which feeds the images to neural network in batches of size 16 is developed and used. Data is augmented by randomly rotating and flipping the images. For training, categorical cross entropy function was used with Adam optimizer [8]. The learning rate used was 0.00001. The model was trained for 20 epochs and it took one day for training. The training accuracy, and validation accuracy are saved for each epoch as training progressed.

When data is imbalanced, we might be getting good accuracy by always predicting the same class. For machine learning models, to determine how good they are, the following additional metrics are used for binary classification.

True Positives (TP) - correctly predicted positive values

True Negatives (TN) - correctly predicted negative values

False Positives (FP) - actual class is no and predicted class is yes

False Negatives (FN) - actual class is yes but predicted class in no

Accuracy = TP+TN/TP+FP+FN+TN, ratio of correctly predicted observation to the total observations

Precision = TP/TP+FP, ratio of correctly predicted positive observations to the total predicted positive observations

Recall = TP/TP+FN, ratio of correctly predicted positive observations to the all observations in actual class

F1 Score = 2*(Recall * Precision) / (Recall + Precision) - is the weighted average of Precision and Recall

For multi label classification, to calculate these values, the neural network is changed to produce binary output using binary cross entropy loss and trained for each label separately. The final models were used to make a prediction for the test dataset, and using scikit-learn metrics [9] API, precision, recall and F1 score are calculated for each label separately and average is taken.

Results

Figure 4 shows how training and validation accuracy increased as training progressed with number of epochs. The accuracy remained almost the same at 0.867 after 17 epochs.

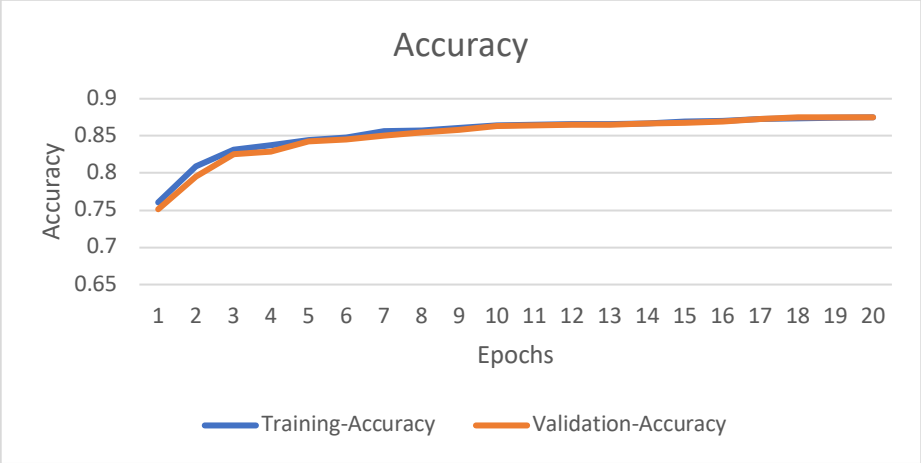

Figure 4: Training and Validation Accuracy vs. Epochs

Figure 5 shows how training and validation loss decreased as training progressed with increase in number of epochs. This shows that chosen learning rate of 0.00001 is good. Also, model was not overfitting because both training and validation loss are going down.

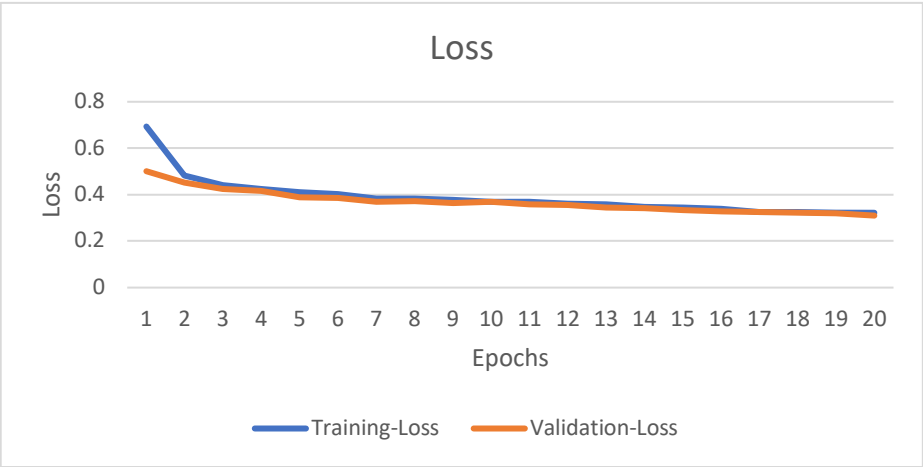

Figure 5: Training and Validation Loss vs. Epochs

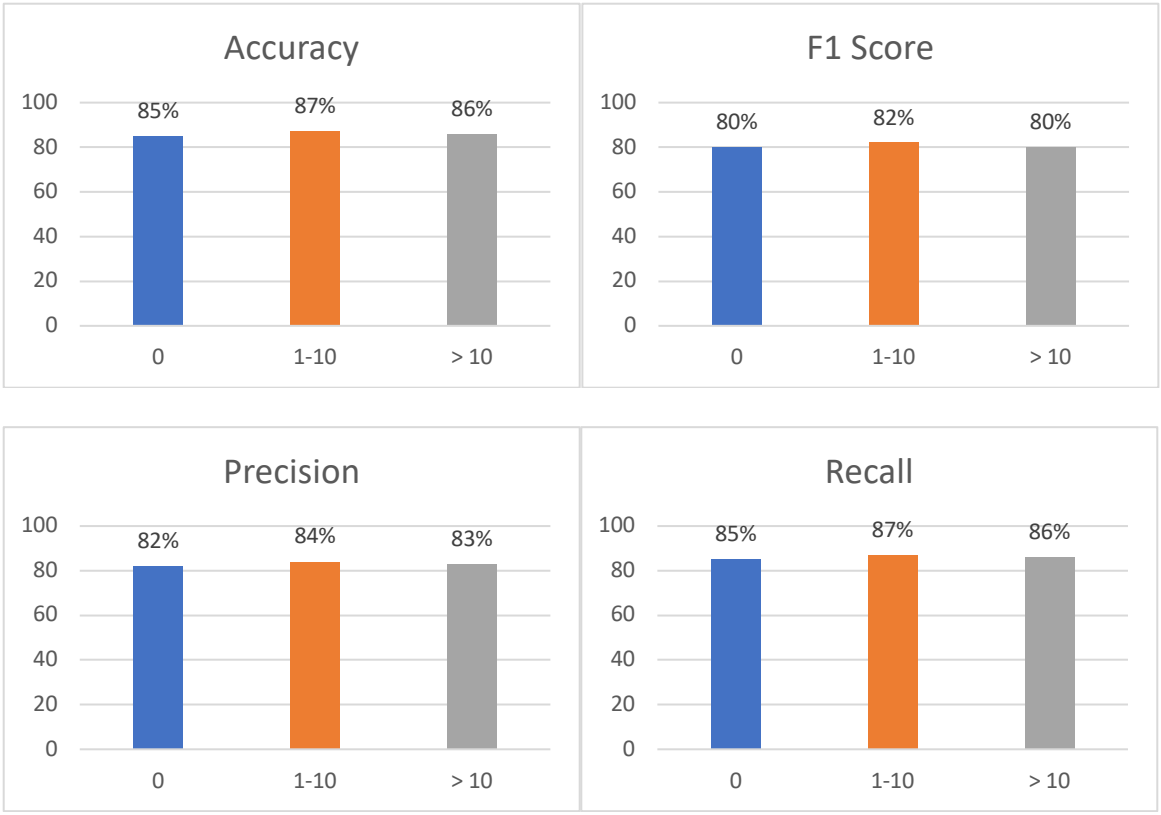

Figure 6: Accuracy, Precision, Recall and F1 Score for different classes

Classification accuracy alone is not enough to figure out if the model can be used for diagnostics. A model can be poor at predicting positives and still achieve a high accuracy by always predicting negative. To establish the validity of the model, three other metrics namely precision, recall and F1 score are calculated based on predictions obtained with test data. These are computed separately for each class and shown in Figure 6. Overall, average precision of 83%, recall of 86% and F1 score of 80% is achieved. These values establish that the model is able to distinguish between the classes. Higher value of recall over precision means that the model shows more false positives and false negatives. This is actually good because it is better to predict more positives than negatives.

Conclusions

In this paper, a novel way to automatically quantify lymphocytes in whole slide images of cancer tissue is presented. The neural network model developed on top of Resnet50 has achieved accuracy of 86.7% with test data. The model has achieved 83% precision and 86% recall, with F1 score of 0.8, which indicates that the model is predicting all classes. Higher value of recall over precision means that the model predicts more false positives than negatives. Further improvement in accuracy can be possibly achieved by tuning hyper parameters, adding more classes, and adding more layers to the network.

This model can be used as diagnostics tool for cancer. This will also help with cancer research.

References

1. Kaiming He and Xiangyu Zhang and Shaoqing Ren and Jian Sun (2015), Deep Residual Learning for Image Recognition, arXiv
2. Hussain, Mahbub & Bird, Jordan & Faria, Diego. (2018). A Study on CNN Transfer Learning for Image Classification.
3. Lymphocyte Assessment Hackathon, https://lysto.grand-challenge.org/
4. Hierarchical Data Format, https://en.wikipedia.org/wiki/Hierarchical_Data_Format
5. Immunohistochemistry, https://en.wikipedia.org/wiki/Immunohistochemistry
6. Google Cloud Machine Learning Engine, https://cloud.google.com/ml-engine/
7. Keras: The Python Deep Learning library. https://keras.io/
8. Kingma, D. P. & Ba, J. (2014). Adam: A Method for Stochastic Optimization (cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015)
9. Scikit-learn, Machine Learning in Python**,** https://scikit-learn.org/