**MSBD 6000B Group Project**

Introduction

The multiple-instance learning (MIL) was introduced by Dietterich, Lathrop & Lozano-Pérez in 1997 [1]. MIL is a form of weakly supervised learning for problems with incomplete knowledge about labels of training examples. Training instances are arranged in sets, called bags, and a label is provided for the entire bag. MIL has been utilized to a wide range of applications from image concept learning to stock market prediction.

Task

The x-ray images are needed to classify into normal and abnormal to help detect the beast cancer and the algorithm of MIL is needed to implement.

Data

There are three data set including train, validation and test with different dimensions and total 398 x-ray images.

Data Preprocessing

Due to different images size, all the images including training images, validation images and testing images are resized to 227 x 227 and normalized by dividing 255. Image data generator is used for generating batches of tensor image data with real-time data augmentation.

With the image data generator, we randomly rotated the original images with the degree range 20 for random rotations, shift the images randomly by 0.2 fraction of total width and height and also randomly flip inputs horizontally to enrich the training dataset and cover more variance of the data distribution hence to generalize a better model for prediction.

Since the ImageDataGenerator is designed to be iterated by the deep learning model fitting process, creating augmented image data just-in-time. This reduces the memory overhead, but the tradeoff is that it adds some additional time cost during model training.

Implementation

The implementation of the method and model is followed the suggested paper [2]. However, the architecture of CNN and the EM-based method in this project is fine tuned and modified respectively. The images are cropped into many image patches. If the amount of tissue in a patch is less than a certain ratio, for example, 20%, the patch is dropped.

Here is our CNN model for each EM step.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

==========================================================

input (InputLayer) (None, 227, 227, 1) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv1\_1 (Conv2D) (None, 111, 111, 80) 4000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batchnorm\_1 (BatchNormalizat (None, 111, 111, 80) 320

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

max\_pooling2d\_1 (MaxPooling2 (None, 55, 55, 80) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv1\_2 (Conv2D) (None, 26, 26, 120) 240120

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batchnorm\_2 (BatchNormalizat (None, 26, 26, 120) 480

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

max\_pooling2d\_2 (MaxPooling2 (None, 13, 13, 120) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv1\_3 (Conv2D) (None, 11, 11, 160) 172960

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv1\_4 (Conv2D) (None, 9, 9, 200) 288200

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

max\_pooling2d\_3 (MaxPooling2 (None, 4, 4, 200) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

flatten\_1 (Flatten) (None, 3200) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_1 (Dense) (None, 320) 1024320

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_1 (Dropout) (None, 320) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_2 (Dense) (None, 320) 102720

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_2 (Dropout) (None, 320) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

instance\_output (Dense) (None, 2) 642

==========================================================

Total params: 1,833,762

Trainable params: 1,833,362

Non-trainable params: 400

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Due to the limitation of our local machine resources, it is nearly impossible to complete our model training process in our local environment. Thus, we deployed our model training process in the Amazon Web Services cloud environment to train our model with a much more powerful machine instance. After that, we pull the model and weights back to our local environment for prediction.

Validation

The validation set is used to evaluate the performance of our model. The accuracy of validation set is about 0.7.

Future Works

The image random cropping function which selects a randomly chosen offset would be implemented. The image random cropping function could reduce the size of images and let the process of whole function can be faster.

Conclusion

This work capitalises on the advances and success of multi-instance learning which has become more and more popular after it was formally introduced in 1997. There are many supervised learning methods to be adapted or extended for the MIL setting and MIL has been proved successful practically to learning problems with ambiguous label. Future work will focus on exploring different choices of deep learning models, as well as further development of applications of this idea.

References:

[1] Dietterich, Thomas G.; Lathrop, Richard H.; Lozano-Pérez, Tomás (1997), "Solving the multiple instance problem with axis-parallel rectangles", Artificial Intelligence, 89 (1–2): 31–71.

[2] Le Hou; Dimitris Samaras; Tahsin M. Kurc; Yi Gao; James E. Davis; Joel H. Saltz; D (2015), “Patch-based Convolutional Neural Network for Whole Slide Tissue Image Classification”