Sophia Szady and Jack Aleksa

Deep Learning Final Project Report

5/15/23

***Introduction: Jack***

There are a multitude of diseases that plague the modern world. Over centuries, our ancestors were trained to recognize symptoms and associate them with certain diseases. However, diseases can have a range of symptoms, some of them extremely similar to others. Since modern medicine and advanced technology, doctors are now better trained in disease diagnosis, and second opinions, as well as a diagnosis by committee, have saved many lives. However, what if we used AI to help give more accurate diagnoses? We proposed constructing a neural network that takes a patient's given symptoms and outputs possible disease.

***Previous Solutions: Sophia***

Seen on Kaggle, 39 other uses of this specific dataset have been uploaded. Other methods of training the model have included using a CatBoostClassifier, Decision Tree Models, Random Forests, Boosted Trees, SVM, F1 score, and Naive Bayes. Other users just created visual representations of the data and looked at each disease specifically. More complex models such as Decision Trees and Random Forests resulted in models with very high accuracies reaching 100% in some cases.

***Dataset: Sophia***

Our dataset is titled *Disease Prediction Using Machine Learning* from Kaggle and was uploaded by user kaushik268, who is a student at Nirma University in Ahmedabad, Gujarat, India. On Kaggle the dataset has been downloaded 16577 times, showing the extent this dataset has been used. The dataset consists of 41 common diseases mapped to 132 different symptoms. Using binary identification of symptoms, each disease has a unique set of symptoms to identify a said illness. There are a total of 4692 cases in the data set with a representation of each disease.

***Proposed Method: Sophia***

Using the methods we have learned throughout the semester, we started with data preprocessing to reformat the data into x and y in order to pass through our model. Then we converted the data into one hot encoding using nested for loops and split the data into test, train, and valid sets. We chose to run an LSTM model on the data as they are useful for memory management and applied well to our data set. The LSTM model consisted of an initial layer that had the number of diseases by number of symptoms. Then we added a Dense layer that was the number of unique diseases, and used the softmax activation function as it pairs well with categorical cross entropy as a loss function. When compiling the model we used the Adam optimizer and monitored accuracy, and fit the model using validation data, with a batch size of 32 and 25 epochs.

***Evaluation Method: Jack***

We originally implemented EarlyStopping into our model but in some cases this caused our model to stop too early. However, if we increased the patients of our EarlyStopping function the model would continue to run epochs and the model would overfit. In the end, we decided to ditch EarlyStopping and limit the number of epochs to 25. We found that anything above would cause overfitting and anything below 25 epochs the model was still training.

Once the model was trained we used common methods that we learned in class for evaluating the different metrics in our model. First, we used the Keras evaluate model method to test our trained model with a set of data pulled from the original dataset specifically for testing. When then printed the loss and accuracy of the model. Our second method of evaluation was using a Seaborn heatmap, alongside a SKlearn confusion matrix. Using these in parallel we were able to see all the different cases in our testing set and see how cases were mapped from our trained model.

***Results: Jack***

Throughout this process, we were worried about overfitting and diligently watched the loss and accuracy over the epochs. The accuracy and loss ended up being extremely volatile and these numbers would dramatically fall and skyrocket over the course of the training. After testing our model it was reported that it has a final loss of 0.7399 and an accuracy of 0.8167.

***Discussion: Sophia***

There were 5 diseases out of the 41 in our model that were entirely misdiagnosed as different diseases. When looking into the symptoms to find similarities, 2 out of the 5, chicken pox being diagnosed as a drug reaction can be explained by the overlapping symptoms of itching and skin rash, as well as Jaundice being diagnosed as Tuberculosis seen through the overlapping yellowing symptoms of eyes and skin. The other 3 misdiagnosed diseases potentially could have resulted from the limiting of epochs while running our model. The discrepancies seen in our model show the potential dangers in using deep learning models to diagnose actual patients. Although models would be heavily vetted before being put into the real world, both type I and type II errors when it comes to diagnoses are very detrimental. Where our model had flaws in entirely misdiagnosing a few diseases is a big difference, for example between a drug reaction and the chicken pox seen in our model. This project has made exponentially clear the difficulties that doctors face in diagnosing a disease.