Task 3   
Saim Ali  
  
**3.1**

To understand how a LSTM cell works we must first understand how a Long Short Term Memory model works. LSTM is a model in a Recurrent Neural Network which is used to predict the next value or character of a sequence using the prior data given. This sort of RNN can be used to predict next words in a sentence, picture interpretation, as well as many other real world uses. In order to achieve these functions a LSTM cell must be used in the RNN. The LSTM is composed of three different gates which help it evaluate data fed into the cell and output computed data according to the cell’s parameters.

A LSTM differs from the traditional RNN cell as it adds an internal state. This internal state is used alongside the output and input which is fed into the cell. The LSTM cell’s first gate is the Forget Gate. This gate as the name suggests lets the cell know which information which is fed into the cell can be forgotten as it is no longer contextually relevant. To decide whether we should keep the data from the previous timestamp or forget it a function is used in the forget gate. After using the function, a sigmoid function is then applied over it which results in a number between 0 and 1. If the number which is multiplied with the cell state of the previous timestamp 0 we will forget everything and if the value is 1 we will forget nothing. This helps the LSTM filter out useless data which can become redundant in larger scenarios.

Secondly, the Input Gate is present in the cell which is used to quantify the importance of the new information which is being carried by the input. A function is used in the input gate which results in a number which again has a sigmoid function applied over it resulting in a number between 0 and 1. The new information is evaluated as 0 being not important and 1 being important. The same information of the hidden state and the current state will be passed through the tanh function now. The tanh operator will create a vector with all possible values between -1 and 1 to regulate the network.

Lastly, the Output Gate is used in a LSTM to determine the value of the next hidden state. This gate will also contain the information from the previous inputs. A third sigmoid function is used with the values of the current and previous hidden state. The new cell state generated from the cell state is passed through the tanh function where both of these outputs are then multiplied. Based off of the answer given our network will decide which information the hidden state should carry, and this state is used for prediction. Then the new cell state and new hidden state are carried over to the next time step.

These three gates in short are: forget gate(determine which relevant information from prior step is needed to be kept), input gate(decides what relevant information can be added from the current step), and output gate(finalized the next hidden state).

**3.2**

One problem faced by traditional RNN models is that it is difficult to train a RNN which requires long-term memorization. LSTM addresses this flaw in the RNN model by performing better in these large long-term datasets as it had more additional special units that can hold information longer. The inclusion of a memory cell in LSTM which can maintain information in memory for long periods of times makes LSTM the far better option for models that address long-term memorization.

**3.3**When creating these plots, I faced many challenges and did complete the task with colored plots but unfortunately my code was erased with a clean installation of jupyter on my pc. I redid this task and was unable to create a for loop which iterates through the y\_train.txt file and reading 1-6 and deciding the colors for the datapoints. Before my code was erased, I managed to evaluate the graphs and create a general idea of what is occurring in the dataset.  
  
My findings:   
From the plots I accumulated, I found that most of the features are affected by three main tasks: Walking, Walking upstairs, Walking downstairs. From my interpretation of the data I believe that more energy consuming activities involving walking require heavier body movements so they will greatly affect any of the nine features that we have in our dataset. I also found that the green color in the plots (standing) did not affect any of the features at all and I believe since standing is a non-energy expending activity, this activity will not affect the dataset. I also found laying to have a similar affect on the data as standing as it is a low energy activity.

Chart, histogram

Description automatically generatedChart, histogram

Description automatically generatedChart, histogram

Description automatically generatedChart, histogram

Description automatically generatedChart, histogram

Description automatically generatedChart, histogram

Description automatically generatedChart

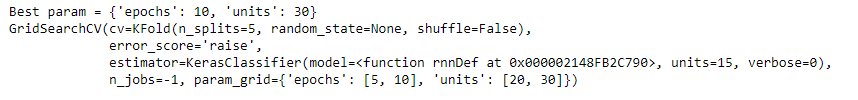
Description automatically generatedChart, histogram

Description automatically generatedChart

Description automatically generated

**3.4** The K-fold cross validation is a procedure that we use in statistical analysis to estimate the skill of the model on new data. It is a resampling process which has a single parameter called k. K refers to the number of folds or groups that the data we are given will be split into. For example, if k=5 then a 5-fold cross validation will take place. K-1 folds are used in our data as the training and the remaining fold will be used for the testing. We will exhaust all combinations of k’s which can be testing or training sets. By doing so we can estimate the skill of a model on unseen data in an easy way. When evaluating the different models, we will select the best using the k-fold cross validation. This method also helps us avoid overfitting .

**3.5** Using the nine given features I tuned the RNN model parameters and chose epochs = 10, units = 30 using the 5-fold cross validation. Grid search was used to find these parameters to find the best accuracy I could possibly find with my machine. I was able to compile an accuracy of 78% which is pretty good for this model. I found this task to be hard to run on my computer as it is very old so was limited by my resources to get a better accuracy rating.

Table

Description automatically generated  
  
**3.7** When I replaced my RNN with a LSTM structure and repeated 3.5-3.6 I found that the LSTM structure provided me with a higher accuracy rating. I believe this is because we are dealing with a relatively large dataset so having a 3 gated cell in the LSTM structure which can use memory to keep/forget inputs help us gain accuracy through this approach. My best accuracy for the LSTM with the parameters after some tuning and grid search was 88% which is higher than the RNN model accuracy I got.   
Table

Description automatically generated  
  
**3.9** A bi-directional LSTM is a neural network which has the sequence information in both directions backwards or forward referring to the past and future. In this sort of LSTM the input will travel in two directions rather than one in the traditional LSTM. Since the input flows in both directions this model is able to utilize information from both sides which makes it a slightly more powerful tool for modeling neural networks. There is an addition of an LSTM layer which reverses the direction of the information flow which helps achieve this affect. This structure is especially helpful in natural language processing where we need to analyze both sides of an input to give the user the best available guess of the next word.  
  
**3.11** This task was a true test to my ability to interpret and work with data in python. I am still relatively new to machine learning in python, but this task helped me gain a few skills in the subject. From this task my conclusion is what follows: This task was based off of a data collection from 30 volunteers using a smartphone to track their activities over a period of time. When evaluating this dataset, you much combine parts together and think critically to be able to fully utilize and understand the dataset. The six different activities (WALKING, WALKING UPSTAIRS, WALKING DOWNSTAIRS, SITTING, STANDING, and LAYING) were recorded in an axial plane which then was given in the dataset. From evaluating the data, I found that more extraneous activities which involved more movement such as walking were far more dominant in the dataset. Any activity that involved walking was the easiest to identity throughout the time series above. As for the neural network models created through my studies above, I believe I was able to correctly create a RNN as well as a LSTM while also tuning the parameters through grid search while also taking advantage of the 5-fold method used. I found the LSTM to be able to get a higher accuracy rating which I hypothesized as the LSTM is a more intricate model which uses three gates to store and forget information while calculating the next output. I was unable to fully get a working bi-directional LSTM to work through my code, but my hypothesis is that the bidirectional LSTM will have an even higher accuracy rating as we will be adding an additional method to the structure of the LSTM which allows us to traverse to a previous input to evaluate it in out output. In conclusion, this dataset was very hard to comprehend but after many long hours and obstacles I was able to somewhat evaluate the data and report on my findings through the models I constructed above.