Predicting Body Rocking Behavior using LSTM-RNN

Shalin Rathi Devang Upadhyay Ashish Kumar Jayantilal Jain

sjrathi drupadhy ajain32

1. ***METHODOLOGY***

**[1] Data**

We have 10 different sessions and each session has an arm and wrist data in the form of 6 features. These six features are in defined as acc1, acc2, acc3, gyro1, gyro2, gyro3. Each session has a time-stamp for every observation and a detection value. This detection value is either 0, which means that no body-rocking behaviour is observed or 1, which means that a body-rocking behaviour is observed.

For every session the 6 features for arm and for the wrist are each combined into a single dataset. The combined dataset now has 12 independent variables. We extract these different features from the combined dataset using time series methods.

**[2] Architecture**

We used the CNN-LSTM model. The first layer is a 1-dimensional convolutional layer with 64 filter, followed by a dropout layer with keep probability of 0.5, followed by a max-pooling and a flatten layer. Then the LSTM layer with 100 neurons in as the first LSTM layer. Next layer is a dropout layer with probability of 0.5, then a dense layer of 50 neurons.

1. ***MODEL TRAINING AND HYPERPARAMETER SELECTION***

Firstly, we trained the model on single layer LSTM module with one dense layer. Then we tried with a two-layer LSTM with one dense layer. Then we implemented two conv1d layers with 1 LSTM and 1 dense layer. Thenone conv1d and 2 LSTM with 1 dense. The models mentioned gave a ow accuracy as compared to the CNN-LSTM model which we defined in the architecture section of the report. Therefore that model was selected inn the end.

The following are the hyperparameters we got for the LSTM layer:

Cells in the LSTM layer = 100

input\_shape = (Number of rows, 1 , 84)

batch input\_shape = (Number of rows, 1 , 84)

The following are the hyperparameters for the dense layer:

Dense layer = 1

Activation = ‘ReLU’

The different combinations of activation functions we tried were ‘ReLU’ and ‘Softmax’

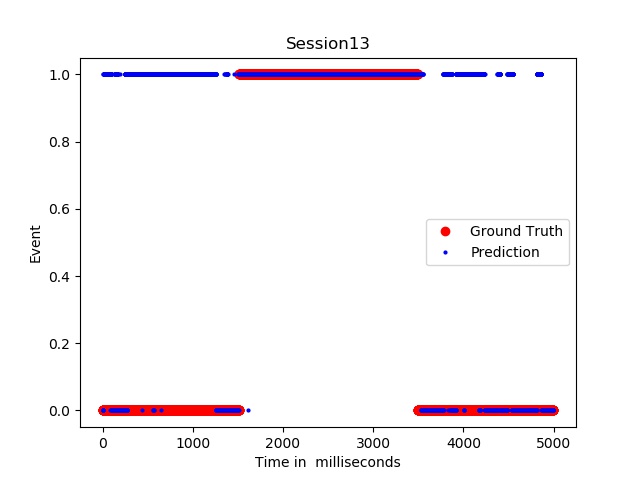
Cells in the LSTM layer we tried were 100 and 150

1. ***EVALUATION***

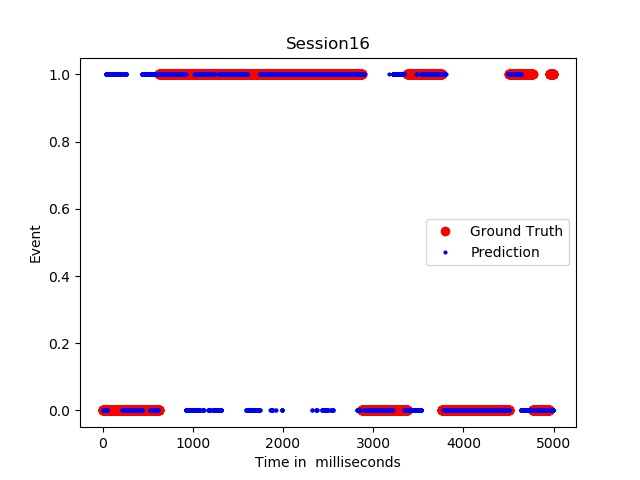
We trained for 10 epochs with a batch size of 128. We tested this model on sessions 15 and 16 and got an overall accuracy of **70.3%**

For every session, we plotted graphs that showed ground truth vs our prediction. This gave us an idea about the accuracy that we received for each session. Some sessions showed very good results, while some gave a bad accuracy.

The graphs have been plotted for a specific time frame Given below are a few of the interesting plots incurred:

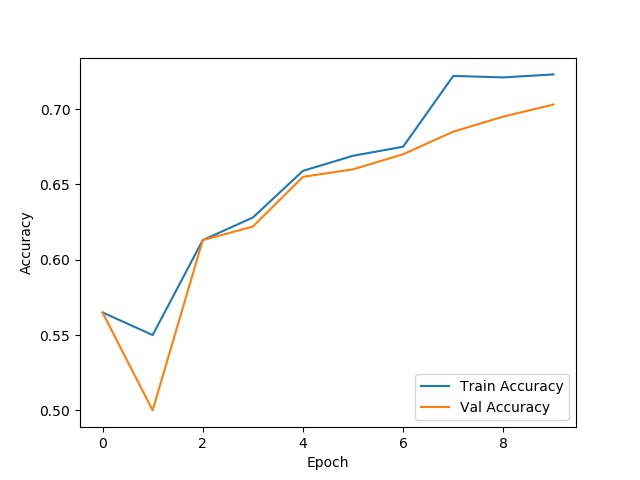


*Figure 1. Ground Truth vs Prediction(session 13)*

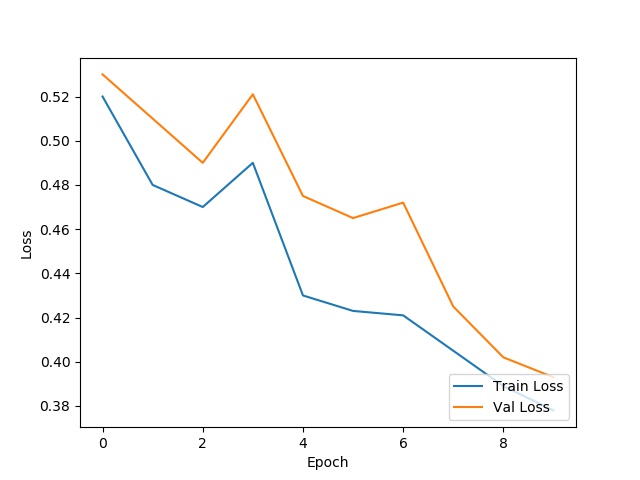


*Figure 1. Ground Truth vs Prediction(session 16)*

We also plotted some figures to determine the validation accuracy and loss along with training accuracy and loss versus the epoch. This gave us an idea on the changes in the accuracies and losses. The plots are given as follows:



*Figure 3. Accuracy vs Epoch*



*Figure 4. Loss vs Epoch*

1. ***REFERENCES***
2. <https://arxiv.org/pdf/1709.05956.pdf>
3. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6077579/>