Convolutional Recurrent Neural Network (CRNN) for EEG 4-class Motor Classification

Wenyang Zhu
UID: 904947071
EE 239AS Student
wendyzhu823@ucla.edu

Xiaohan Wang
UID: 405033965
EE 239AS Student
wangxiaohan329@gmail.com

Yang Tang
UID: 505036001
EE 239AS Student
ty111gm@gmail.com

March 18, 2019

Abstract

This project aims to classify the electroencephalograph (EEG) data [1] provided in Brain-Computer Interaction (BCI) Competition [2]. The data is classified into four different motor imaginary tasks (i.e. movement of the left hand, right hand, both feet, and tongue) using Convolutional Recurrent Neural Network. The report compares performance using both CRNN and CNN, and also the results between a single subject with all subjects. Besides, time duration variations from EEG data are also taken into consideration and compared in this report.

1 Introduction

This report constructs two networks, one is Convolutional Neural Network (CNN), and the other is Convolutional Recurrent Neural Network (CRNN).

1.1 Convolutional Neural Network (CNN)

The architecture of the Convolutional Neural Network is shown in Figure 3. It consists of four convolutional part, and each part includes a 2-dimensional convolutional layer, an Exponential Linear Unit (ELU) activation layer, a batch normalization layer, a 2-dimensdional max pooling layer, and a dropout layer. The first three layers aim to reduce the time-dimension, while the last one aims to reduce the feature-dimension. Since the features between adjacent time steps are relevant and the 22 EEG features are also relevant, CNN is a good choice to extract and learn such relevance. Finally, there is a full-connected layer to get the one-hot vector for classification.

1.2 Convolutional Recurrent Neural Network (CRNN)

The architecture of the Convolutional Recurrent Neural Network is shown in Figure 4. There are four similar convolutional parts in CRNN. We still do convolution on time-dimension because features from adjacent time steps are highly correlated and long length on time-dimension makes RNN layers difficult to learn the data. Between the convolutional parts and the full-connected layer, there are three bidrectional Long Short Term Memory (LSTM) layers. We also add a dropout layer after LSTM layers and remove some dropout layers in convolutional parts to avoid the input of LSTM layers missing too much important information. Since the dataset is time series, Recurrent Neural Network (RNN) like LSTM is a better choice than CNN. It obtains information along time-dimension to make a good model.

2 Results

This section lists the results of single subject and all subjects with CRNN, and the results of all subjects with

CNN. It also lists the accuracy for different time periods in each data sample.

2.1 Accuracy for single subject

The accuracy for 9 single subjects with CRNN is shown in Table 1.

2.2 Accuracy across all subjects

The accuracy for all subjects with both CRNN and CNN are shown in Table 2.

2.3 Accuracy with different time duration

The accuracy of all subjects over different time duration with both CNN and CRNN models are shown in Figure 1 and Figure 2, respectively.

3 Discussion

This section shows the results between two types of networks, and with different subjects. It also discusses a bit about the reason how we make choices on some hyperparameters and architectures. Finally, it makes hypothesis on the previous results with different time duration.

3.1 Compare between networks

Compared to CNN, CRNN gains about 12% higher accuracy. It shows that LSTM exactly works in time series data. Besides the accuracy, CRNN with LSTM needs 40 epochs to get its best performance while CNN only needs 20 epochs. We use the same Adam optimizer in both networks, so the longer training time is due to the complexity of CRNN. Three LSTM layers let CRNN need more time to find the optimal status.

3.2 Compare between one vs all subjects

We try the same CRNN model on each subject and all whole data set as well. We find that some subjects can reach the same accuracy as the whole data set, while the others can't reach such a high accuracy. It is because that single subject data set is so small that the CRNN model is not suitable. For single subject data set, the model needs to be much simpler and adds much dropout and regularization to avoid overfitting.

3.3 Choice of hyperparameters and architectures

We use grid search to find the optimal values of different hyperparameters. For instance, high dropout probability makes the model not learn well, while low dropout probability makes the model easy to be overfitting. To the architectures, we try different number of convolutional and LSTM layers and different order of these layers. Too much convolution and max pooling on time-dimension let the data lost time-series information, but without max pooling, the model will overfit on the training data set. Bidirectional LSTM gains higher accuracy than the normal one. We also try to transfer the 22 features to a 2-dimension data (6*7 matrix) based on the positions on the scalp they are extracted from; however, it seems not work better than directly do convolution on feature-dimension.

3.4 Compare between different time duration

We write a for loop with time duration from 100 to 1000 to find the optimal time duration for all subjects dataset. For both CNN and CRNN model, the general trends are same that as we use data over longer period of time, the test accuracy are also increasing. After some time duration around 500, the accuracy are getting stable with some little oscillation later. Thus, we can conclude that time = 500 is required to get a reasonable classification accuracy. Besides, the best test accuracy for CNN model is 60.7% at time duration 600, and the best test accuracy for CRNN model is 67.3% at time duration 1000.

The conclusions fit the intuition and theory. At the beginning, since the time duration is too short, we need to alter the architecture of the models to fit the output (we delete one convolutional layer for both CNN and CRNN when time duration is shorter than 450 and 300, respectively). The information is not enough, so we get lower accuracy at first. As we increase the time duration, the models are learning more local features and have enough information to make the right decision. Besides, bidirectional CRNN model can analyze the time-series information more comprehensively, so the accuracy of CRNN model are generally better than that of CNN model.

References

[1] BCI Competition IV. BCI Competition IV, www.bbci.de/competition/iv/.

[2] Brunner, C, et al. BCI Competition 2008 Graz Data Set A.

[3] S. Roy, I. Kiral-Kornek, and S. Harrer. Chrononet: A deep recurrent neural network for abnormal eeg identification. *CoRR*, abs/1802.00308, 2018.

A Algorithm Performance and Model Architecture

Person Id	Training/Test Data	Accuracy
0	237/50	60.3%
1	236/50	28.1%
2	236/50	40.7%
3	234/50	52.4%
4	235/50	60.6%
5	236/50	43.6%
6	238/50	72.2%
7	232/50	32.9%
8	231/50	74.7%

Table 1: Accuracy of Single Subject with CRNN.

Model	Epoch	Accuracy
CNN	20	52.2%
CRNN	40	64.5%

Table 2: Accuracy across All Subjects with CNN and CRNN.

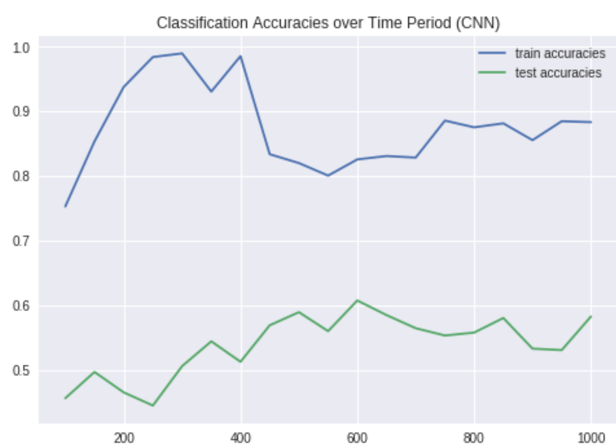

Figure 1: Classification Accuracy of CNN over Different Time Duration. Range is from 100 to 1000 with time step 50.

??

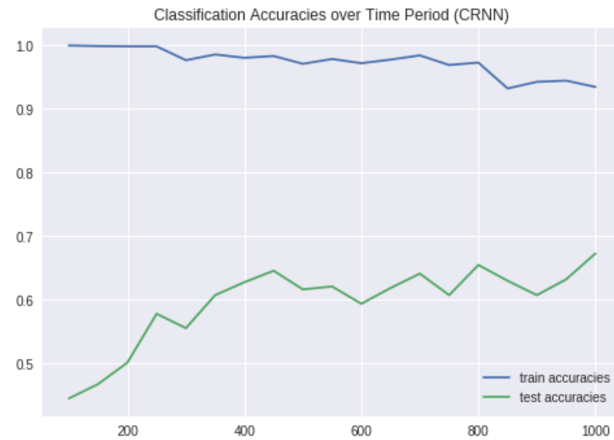

Figure 2: Classification Accuracy of CRNN over Different Time Duration. Range is from 100 to 1000 with time step 50.

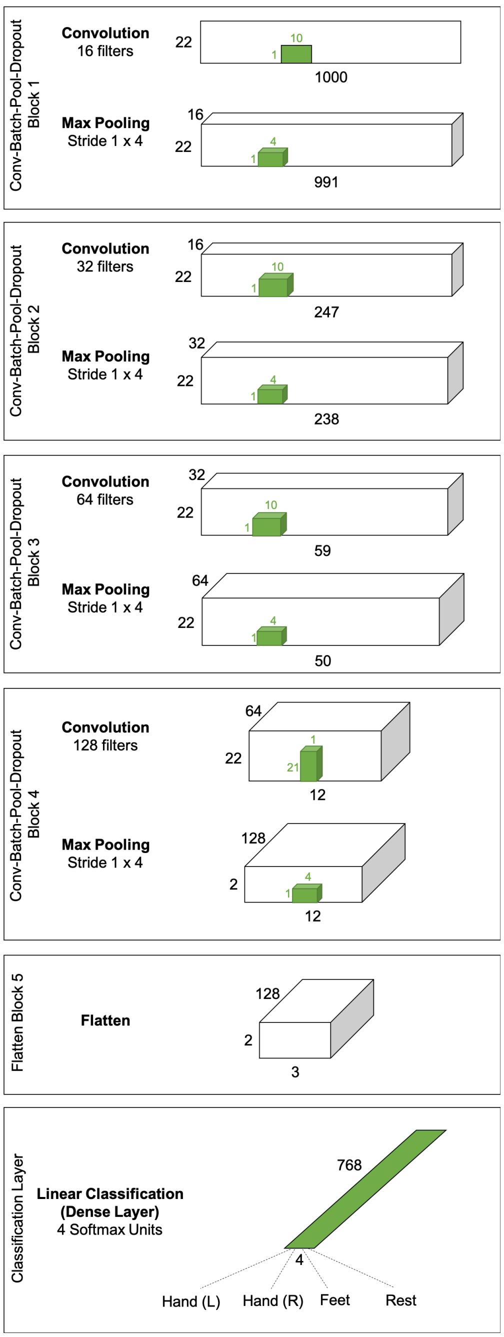

Figure 3: Architecture of CNN. (Loss Function: Cross-entropy. Optimizer: Adam)

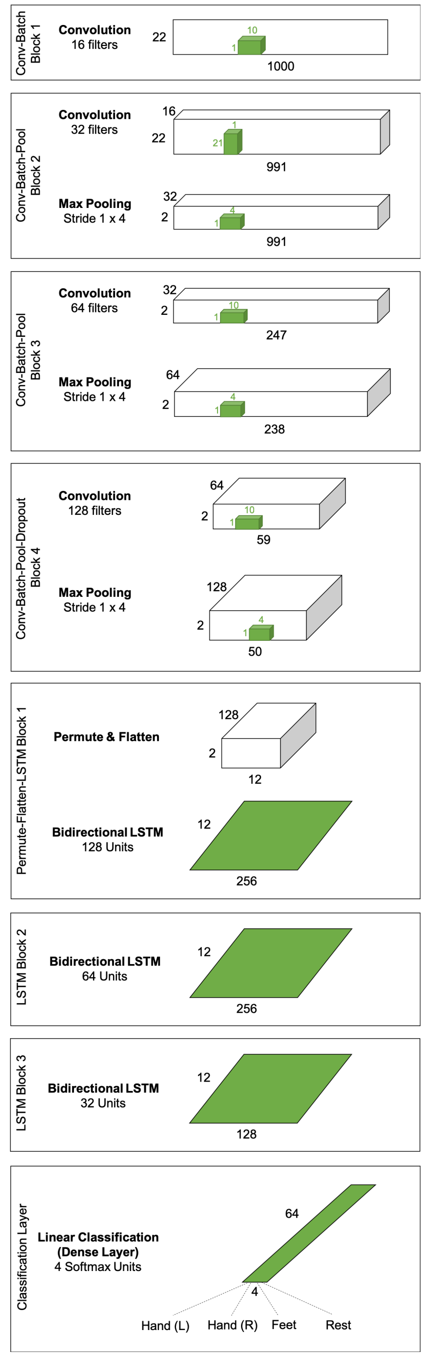

Figure 4: Architecture of CRNN. (Loss Function: Cross-entropy. Optimizer: Adam)