EC ENGR C247 Final Project

Hiroto Odaka
005950033
University of California, Los Angeles
hodaka@g.ucla.edu

Ryan Chien
905949417
University of California, Los Angeles
ryanchien@g.ucla.edu

Abstract

For this project we are tasked with classifying an electroencephalography (EEG) dataset across 9 subjects into four motor imagery tasks. EEG reflects the coordinated activity of millions of neurons near a non-invasive scalp electrode. We implemented the following models for this task: CNN, LSTM, LSTM+CNN hybrid, and Vision Transformer (ViT). The CNN was the best performing model, giving us the highest test accuracy of 71.6%.

1. Introduction

We introduce several neural networks that we use to classify the EEG data. We also explain the brief backgrounds of each model and our motivation to implement these networks for this EEG classification task.

1.1. LSTM

The long short-term memory (LSTM), a type of recurrent neural network (RNN) structure, is capable of learning order dependence in sequence prediction problems. RNN's consist of recurrent structures that locally feed information from previous inputs into, which allows it to make use of context and dependencies between time steps. Each LSTM layer has memory cells that stores and outputs information, and uses ”gates” to control the flow of information in and out of the cell. Because of this architecture, we believed that the LSTM would perform well for this task. We implented an LSTM with two layers, followed by two sequences of a fully connected layer, a batch normalization layer, a ReLU activation layer, and a dropout layer.

1.2. CNN

CNN [4] has been widely used for image recognition [9] because it can extract features of images in a less computationally expensive way, compared to fully connected networks. CNN has also been used in the EEG classification task [8]. Therefore, we chose to implement a CNN in this project. We constructed two kinds of CNN networks. The first one is a network where the data is convoluted and pooled in both spatial and temporal directions (labeled as the ”Vanilla CNN” model), and the second one is a network where spatial and temporal information is convoluted and pooled separately (we call it ”Spatio-temporal CNN”). Spatio-temporal CNN is inspired by EEGNet [8]. The convolutional and pooling layers are followed by batch normalization and dropout layers to prevent overfitting and gradient vanishing and exploding. In the last part of the network, we use fully-connected layers. More details of the architecture are on the extra pages.

1.3. LSTM + CNN

CNN-LSTM is an LSTM architecture designed specifically to solve sequence prediction problems with spatial inputs, such as images. It combines the tools of the vanilla CNN with extracting key features from images, and the tools of the vanilla LSTM with learning order dependencies within sequences. Other common problems they are used to solve include activity recognition, image description, and video description. Our CNN-LSTM consists of four convolutional blocks, each containing a convolutional layer, a batch normalization layer, a max pooling layer, a ReLU activation layer, and a dropout layer. Similarly to our Spatio-temporal CNN architecture, the first convolutional layer is filtered over the temporal dimension and the second convolutional layer is filtered over the spatial dimension. Following those layers, we incorporated the same LSTM architecture as the vanilla LSTM (since these architectures are known to perform well on sequential data).

1.4. ViT

ViT has been recognized as a high-performance image recognition model since it was first invented [1]. Researchers have applied ViT to EEG classification tasks [3]. We implemented ViT for this project to explore ViT's performance with the EEG data provided to us. Although several kinds of ViT architectures, including one with more than a billion parameters, have been introduced so far [7], we use a simple ViT model with a small number of parameters because of limited access to computer resources. Specifically, the patch size is 22 × 2 and the hidden dimension is only eight in our ViT model.

2. Results and discussion

Here we state the results of the neural networks we mentioned in the introduction, LSTM, CNN, LSTM+CNN, and ViT, for the EEG classification task. Each model's best performing validation set parameters during training were used on the test set. We elected to use accuracy as the performance measure for this project. The more detailed results can be found in the extra methods pages.

2.1. LSTM

The training and the test accuracy of the Vanilla LSTM across all subjects is 100% and 27.3% respectively. The training and test accuracy of the Vanilla LSTM for only subject 1 is 100% and 20% respectively. The results clearly show that the LSTM overfit the data. Current research suggests that this is a common problem among LSTM models. One potential solution to reduce overfitting is through Adaptive Boosting (AdaBoost), an ensemble method [2]. Some other simpler methods of improvement could be incorporate different methods of regularization such as increasing dropout or adding noise to the data.

2.2. CNN

The training and test accuracy of Vanilla CNN is 67.6% and 59.8% respectively. On the other hand, Spatio-temporal CNN's training accuracy is 95.1% and its test accuracy is 71.6% which is more than 10% better than Vanilla CNN's. This result is better than one in research done in 2017, where they used the same dataset (BCI Competition Dataset). This shows that convolutional and pooling layers along the spatial and temporal directions separately are effective in the EEG classification task, as suggested in the paper [8].

Regarding the loss trajectories of Spatio-temporal CNN1 the validation loss stops decreasing after around the 30th epoch while the training loss keeps decreasing. Also, the validation accuracy starts plateauing around that epoch although the training accuracy continues to improve2. This means that the model performance of the classification would no longer improve even if training the model for more epochs.

When training Vanilla CNN and Spatio-temporal CNN only with subject 1's data and optimizing the model, both models become over-fitted to the training data. The test accuracy of Spatio-temporal CNN is as low as 58.0% while the training accuracy is almost 100%.

2.3. LSTM+CNN

Data preprocessing was performed on the dataset before passing it into the LSTM+CNN model. The training and test accuracy of the LSTM+CNN is 100% and 62.5% respectively. The training and test accuracy of the LSTM+CNN for only subject 1 is 100% and 45% respectively. Similarly to the Vanilla LSTM, the LSTM+CNN severely overfit the dataset, even though the data preprocessing and convolutional layers improved the baseline accuracy from the Vanilla LSTM. As stated in the LSTM section, AdaBoost and other methods of regularization could be used to reduce the overfitting.

2.4. ViT

The ViT model we constructed classifies the test dataset with an accuracy of 42.0% while it classifies the training dataset with 54.4% accuracy. According to a paper investigating ViT [5], the patch size needs to be small enough according to tasks for ViT to perform well. In our project, we set the patch size to 22 × 2, which could be not small enough, due to limited access to computational resources. If we had access to a much bigger size of random-access memory (RAM) and high-performance GPU, we would try a smaller patch size which could make the performance better. Moreover, in our model, the hidden size is 8 and the number of heads is 2 because of the same reason although, in the original ViT [1], the hidden size is 768 and the number of heads is 12. Increasing the hidden size and head size with more computer resources could improve ViT's representation, which leads to higher performance. A previous paper [6] also points out that the size of the dataset is also critical in the training of transformers because of its smooth loss function. Increasing the data size could improve ViT's performance further.

2.5. Data Preprocessing

We implemented data preprocessing, following the teaching assistant's code. Specifically, the preprocessing is done using a combination of trimming the dataset, max pooling, adding values of average plus gaussian noise, and subsampling. The performance of Spatio-temporal CNN, which performs the best among the models we made, changed little with the data preprocessing in the cross-subject classification. On the other hand, in the within-subject classification, the test accuracy of Spatio-temporal CNN with the data preprocessing shows more than a 10% increase than the one without the preprocessing. The ViT-based model shows around 3% improvements in both training and test accuracy.

From these results, we could say that data preprocessing does not always improve a model's performance.

2.6. Classifying across all subjects versus one subject

For each model, we classified across all 9 subjects in the dataset as well as exclusively for subject 1. We found that all the models tended to overfit when classifying for subject 1, thus resulting in much lower test and validation accuracies and near perfect training accuracies.

2.7. Comparison between the models

Among the all models we implemented, spatio-temporal CNN performed the best in terms of test accuracy. This could be because spatial direction convolution and temporal direction convolution layers can effectively capture key features respectively, as argued in the paper about EEGNet [8]. The second best-performing model is LSTM+CNN, although the model is overfitted to the training data. It is also notable that the ViT model's test accuracy, 45.5% with data-preprocessing, is higher than the LSTM model's despite the fact that the number of parameters of the ViT model, which is 2,044, is nearly 70 times smaller than that of the LSTM model, which is 139,268. Overall, we could say that models with a CNN architecture perform better than those without the architecture.

2.8. Possible ways for improvement

There can be several ways to increase the performance of all the models in this task. First, simply collecting more data would make the models' performance better since the model could learn more universal features. Second, access to more computational resources would make it possible to explore better parameters, allowing for usage of KFold Cross Validation to be more computationally feasible. Third, we could use unsupervised machine learning models to learn the more important features in the dataset to model.

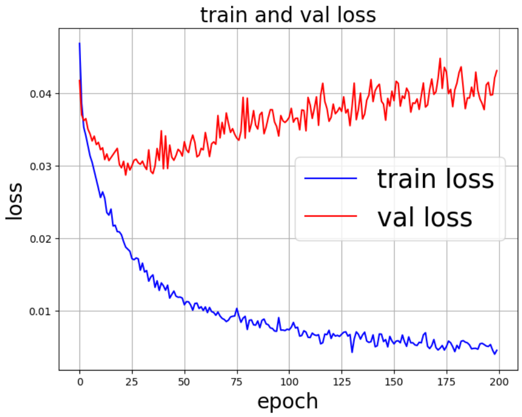

Figure 1. Training and validation loss trajectories of Spatio-temporal CNN

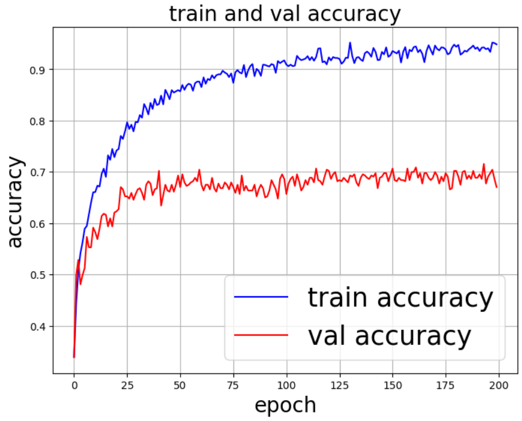

Figure 2. Training and validation accuracy trajectories of Spatio-temporal CNN

References

[1] et al. Alexey Dosovitskiy, Lucas Beyer. An image is worth 16x16 words: Transformers for image recognition at scale. *Computer Vision and Pattern Recognition*, 2021.

[2] Liangming Wang Yunsheng Lu Fagui Liu, Muqing Cai. An ensemble model based on adaptive noise reducer and overfitting prevention lstm for multivariate time series forecasting. *IEEE Access*, 7, 2019.

[3] Huihui Zhou Jiayao Sun, Jin Xie. Eeg classification with transformer-based models. *2021 IEEE 3rd Global Conference on Life Sciences and Technologies*, 2021.

[4] Hinton G. LeCun Y, Bengio Y. Deep learning. *Nature*, 521:436–44, 2015.

[5] et al. Lucas Beyer, Pavel Izmailov. Flexivit: One model for all patch sizes. *arXiv preprint*, 2022.

[6] Songkuk Kim Namuk Park. How do vision transformers work? *ICLR 2022 conference paper*, 2022.

[7] et al. Salman Khan, Muzammal Naseer. Transformers in vision: A survey. *ACM Computing Surveys*, 54(200):1–41, 2022.

[8] et al. Vernon J. Lawhern, Amelia J. Solon. Eegnet: A compact convolutional network for eeg-based brain-computer interfaces. *Journal of Neural Engineering*, 15(5), 2016.

[9] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. Cnn-rnn: A unified framework for multi-label image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2285–2294, 2016.