BIO Web of Conferences **61**, 01014 (2023) https://doi.org/10.1051/bioconf/20236101014
FBSE 2023

Prediction of Preoperative Scale Score of Dystonia Based on Few-Shot Learning

Yumeng Chen*

Department of Bioinformatics and Biostatics, Shanghai Jiao Tong University, Shanghai, China

Abstract. As a neurological disease, dystonia mainly has symptoms including muscle stiffness, dyskinesia, tremor, muscle spasm, etc. Dystonia score plays an important role in targeted auxiliary diagnosis, treatment plan design, and follow-up evaluation of patients. In this paper, the feature information of brain lateralization is extracted from electroencephalography (EEG) signals by clustering method, while information on time domain, frequency domain, and time sequence are extracted from EEG signals and electromyography (EMG) signals. Various deep-learning models are used to predict dystonia scores. Experiments show that this method can effectively predict dystonia based on the quantitative indicators extracted from few-shot neural signals. The methodology in this paper can help doctors judge the disease more accurately, make personalized treatment plans, and assist in monitoring the treatment effect.

1. Introduction

Muscle spasticity is a common symptom in many neurological disorders, such as stroke, multiple sclerosis, and cerebral palsy [1]. Accurate assessment of muscle spasticity is crucial for effective treatment planning and monitoring. However, the current clinical methods for assessing muscle spasticity are subjective and rely on the observation of physical signs and symptoms. Recent advances in the field of neurophysiology have provided new opportunities for objective assessment of muscle spasticity using electroencephalography (EEG) and electromyography (EMG) signals [2, 3].

In recent years, the development of new technologies such as deep learning has led to exciting breakthroughs in the field of neural signal processing, with numerous high-impact studies published in recent years. For example, deep learning has been applied in brain-computer interfaces (BCIs), disease diagnosis, gesture recognition, and other areas, with promising results [4-6]. While in EEG analysis, deep learning has been used to decode brain signals and translate them into commands for external devices [7]. Deep learning with convolutional neural networks is used for EEG decoding and visualization [8]. While in EMG analysis, deep learning has been used to estimate limb movement and predict human-machine interaction [9, 10]. Overall, these high-impact studies demonstrate the great potential of deep learning in improving the accuracy and efficiency of neural signal processing and suggest that it will play an increasingly important role in the development of future neural signal processing technologies. Due to the limited acquisition tools for neural signal processing and fewer data available for analysis, neural signal analysis mainly focuses on few-shot deep learning [11, 12].

Lateralized brain characteristics have important implications for the diagnosis and treatment of neurological disorders such as stroke, epilepsy, and spasticity. The identification of asymmetrical patterns in brain activity can help in the early detection of these disorders and aid in the development of targeted interventions [13]. Therefore, brain lateralization characteristics are an important indicator of neurological disorders.

Mapping brain asymmetry refers to the study of the differences in function and structure between the left and right hemispheres of the brain. There are various techniques used to map brain asymmetry, including functional magnetic resonance imaging (fMRI), electroencephalography (EEG), and transcranial magnetic stimulation (TMS).

Therefore, the random oversampling method is used in this paper to enhance the data for the few-shot data set, which extracts the brain lateralization features from the local field potentials (LFP) of the left and right brain STN and GPi nuclei of the subjects by K-Means clustering method. Meanwhile, features of the time domain, frequency domain, and time sequence are extracted from the electroencephalography (EEG) and electromyography (EMG) signals of the subjects. Combining the brain lateralization features and other signal features of the subjects, it makes deep learning prediction for dystonia scores through Linear Regression, RNN, and LSTM.

2. Problem Setting

The dystonia score is a valuable tool for healthcare providers to evaluate and manage muscle spasticity. It helps to optimize the patient's treatment plan by providing a baseline for the assessment of muscle spasticity and

* Corresponding author: cym020418@sjtu.edu.cn

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

monitoring the effectiveness of the treatment plan. The dystonia score is an important component of the comprehensive care plan for patients with muscle tone disorders.

Therefore, this study uses the deep learning method to extract features of the time domain, frequency domain, and brain lateralization to predict dystonia scores with the help of local field potential (LFP) and electromyography (EMG) signals of STN and GPi nuclei in patients' left and right brains. (Fig.1)

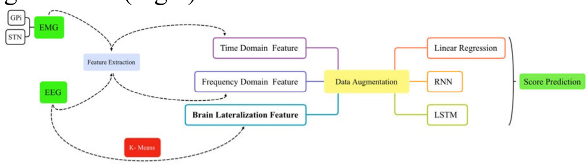

Fig. 1 Research Pipeline

3. Model

Three deep learning models are applied in this study.

3.1 Linear Regression

The linear regression is a basic model widely used in data analysis and machine learning, which describes the linear relationship between independent variables and dependent variables by fitting a straight line (or a hyperplane in high-dimensional space) [14].

3.2 Recurrent Neural Network (RNN)

A recurrent neural network is a neural network model that can deal with sequence data, which has a recursive structure in time and can take the output of the previous time as the input of the current time, so as to capture the time sequence information in sequence data [15].

3.3 Long Short-Term Memory (LSTM)

A long short-term memory network is a recurrent cyclic neural network, which can effectively avoid the gradient disappearance or explosion when processing long-term sequence data, so as to better capture the long-term dependence in sequence data [16].

4. Experiment

4.1 Data Pre-Processing

4.1.1 Feature extraction of time domain and frequency domain:

The data set includes the basic personal information, dystonia score, local field potential (LFP) and electromyography (EMG) signals of STN and GPi nuclei of the left and right brain of 12 patients, among which the dystonia score was selected by BFMDRS (Burke-Fahn-Marsden dystonia rating scale) scale. For each subject, four time zones were selected with 50000 signal data in each time zone, altogether 200000 signal data. For each time zone, four features of time domain were extracted from electromyography (EMG) signals of left and right brain STN, GPi nuclei, and local field potential (LFP) respectively. Besides, four frequency domain features were extracted by Fourier transform.

4.1.2 Feature extraction of brain lateralization:

Considering the importance of brain lateralization features, the following proposals are put forward by this study innovatively. Brain lateralization features are extracted to concat them with feature vectors of the time domain and frequency domain as feature input vectors of each subject. According to Formula 1, four cosines of local field potentials (LFP) of brain STN and GPi nuclei at about 200,000 signal points are calculated for each subject, which represents the lateralization feature values at each moment. Then, all cosines are clustered into four categories by K-Means clustering, and the signal probabilities in four categories of each subject are calculated as the extracted lateralization features of the brain. The arrangement of the four probabilities is based on the clockwise order of the four clusters in the two-dimensional plane after PCA dimensionality reduction.

To sum up, each subject extracted 52-dimensional feature vectors as input.

4.1.3 Data Enhancement:

Because the training of this study is few-shot, random oversampling is used to expand the training set, so that the neural network can learn more information. According to the number of oversampling, it is divided into three levels (level 1: 8-24, level 2: 24-48, level 2: 24-48) to reduce the influence of uncorrelated features on model training by enriching uncorrelated features.

4.2 Model Training

The size of the original sample set is 12, the number of training samples is 8, and the number of test samples is 4. Linear Regression, RNN, and LSTM are selected for training, with 1e-5, 1e-4, 1e-3, 1e-2, and 1e-1 set as the learning rates respectively. Meanwhile, the number of network layers is 2-4 and the hidden size is 64-96 for training. According to the loss convergence with the experiment, 2600 rounds are trained when the learning rate is 1e-5, and 2000 rounds are trained at other learning rates. In addition, in order to compare the training results of RNN and LSTM models, learning rate = 1e-2, layer = 4, and hidden size=128 are set for training according to the pre-experimental results. At the same time, timing information is added and the data of each subject in four time zones are input into the model as timing data in turn for training.

4.3 Model Validation

After each epoch training, log (mse_loss), log (mae_loss), mse_loss std, and mae_loss_std of the test data set are calculated. Finally, the average value is calculated as the test set loss result of this training.

5. Experiment Results and Analysis

5.1 Comparison of Internal Results of Three Models

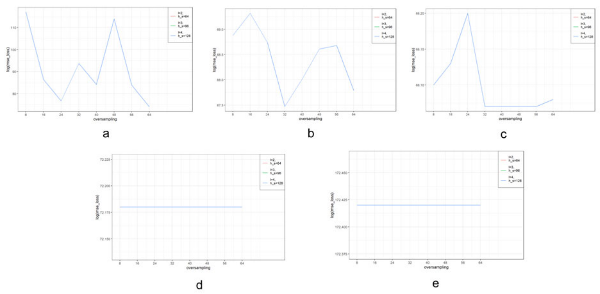

Fig. 2 Results of Linear Regression

As for the linear model which is simple with a small number of parameters, the number of layers and hidden units have little effect on loss. When the learning rate is 1e-1 (Fig. 2a), loss shows the same trend in level 1 and level 3, but the trend is opposite in level 2. When the learning rate is 1e-4 and 1e-5 (Fig. 2d and Fig. 2e), the loss is larger and not affected by the number of oversampling, which may fall into the local optimal solution due to the low learning rate. As for the linear model, the learning rate is between 1e-3 and 1e-2, and the suitable oversampling number is about level 2.

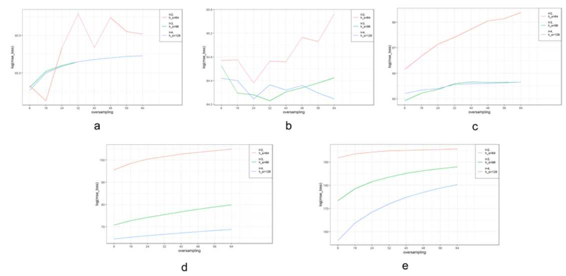

Fig. 3 Results of LSTM

LSTM model has more parameters and a complex gating structure, so the number of layers and hidden units has a great influence on loss. With the deepening network structure, loss tends to decline. The LSTM model is sensitive to the number of oversampling, and its structure can effectively solve the short-term dependency on RNN. However, as for the model structure compared with RNN, LSTM contains more parameters to learn, which dramatically reduces the learning speed of LSTM and is greatly affected by the number of training sets.

When the learning rate is 1e-3-1e-5 (Fig.3 c-e), loss increases at levels 1, level 2, and level 3, which may result from the over-fitting caused by the increasing random over-sampling. When the learning rate is 1e-2-1e-2 (Fig.3a-b), loss generally increases at levels 1-3 with an unstable trend, which may be the unstable convergence of loss led by the small learning rate. As for the lstm model, it is more appropriate to choose the learning rate between 1e-3 and 1e-2 with the over-sampling number around level 1.

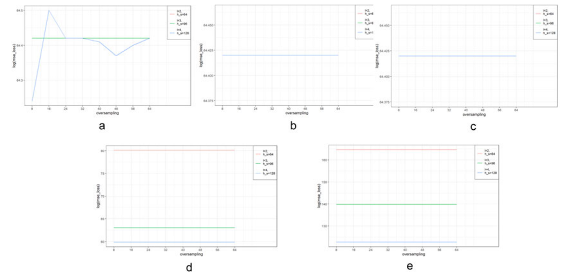

Fig. 4 Results of RNN

The RNN model, as a more complex deep learning model than a linear model, has more parameters and a complex network structure. The number of layers and hidden units has a great influence on loss (Fig.4). With the deepening network structure, loss shows a downward trend. RNN model is insensitive to the number of oversampling, and loss is almost unaffected in the level 1-3 stage. As for each RNN neuron, its parameters are always shared, so the random of oversampled repeated samples has no great impact on parameter learning and loss.

5.2 Ablation Experiments

In order to further compare the effects of lateralization on model training, ablation experiments are conducted (Table 1 S_N, Serial Number; H_S, Hidden Size; L, Layer; Lr, leaning rate; O_s, Over sampling; E, epoch; L_R, Linear Regression; L_B_F, Lateralized brain features). As for the linear model, the addition of lateralized features has a great influence on model training (A1-5, A1-5), which is greatly affected by the number of oversampling (A1, A2). Thus, it has a certain effect on the data enhancement of the few-shot linear model. However, the model is relatively simple, so the network depth has almost no influence on loss (A1, A3).

As for the LSTM model, the addition of lateralization features has a certain effect on model training (B1-5, B1-5), which is also affected by the number of oversampling (B1, B2). Besides, the network depth has a certain effect on model training (B1, B3).

As for the RNN model, the addition of lateralization features has little effect on model training (C1-5, C1-5) and model training (C1, C2). In addition, the depth of the network has little effect on model training (C1, C3).

Longitudinal comparison of the three models' results show that the RNN model loss is smaller and the most stable, with its lateralization features reducing linear model and LSTM model loss to some degree.

Table 1. Results of Ablation Experiments

S_N	H_S	L	lr	O_s	e	log(MSE)	log(MAE)	MSE std	MAE std	L_R	RNN	LSTM	L_B_F
A1	64	2	1.0E-01	8	2000	116.99	3.58	37.26	0.61	√			√
A2	64	2	1.0E-01	24	2000	76.74	3	14.95	0.26	√			√
A3	96	3	1.0E-01	8	2000	116.99	3.58	37.26	0.61	√			√
A4	96	3	1.0E-02	16	2000	69.32	2.8	15.33	0.23	√			√
A5	128	4	1.0E-03	32	2000	68.07	2.8	14.53	0.22	√			√
a1	64	2	1.0E-01	8	2000	129.88	3.98	39.28	0.62	√			
a2	64	2	1.0E-01	24	2000	99.35	3.46	24.63	0.44	√			
a3	96	3	1.0E-01	8	2000	129.88	3.98	39.28	0.62	√			
a4	96	3	1.0E-02	16	2000	68.92	2.81	14.72	0.22	√			
a5	128	4	1.0E-03	32	2000	68.53	2.81	14.56	0.22	√			
B1	64	2	1.0E-01	8	2000	64.83	2.83	14.07	0.25			√	√
B2	64	2	1.0E-01	24	2000	65.34	2.83	14.15	0.25			√	√
B3	96	3	1.0E-01	8	2000	64.81	2.83	14.06	0.25			√	√
B4	96	3	1.0E-02	16	2000	64.19	2.88	13.27	0.26			√	√
B5	128	4	1.0E-03	32	2000	65.56	2.83	14.33	0.25			√	√
b1	64	2	1.0E-01	8	2000	65.16	2.84	14.08	0.26			√	
b2	64	2	1.0E-01	24	2000	64.65	2.82	13.92	0.25			√	
b3	96	3	1.0E-01	8	2000	64.71	2.83	14.03	0.25			√	
b4	96	3	1.0E-02	16	2000	64.08	2.87	13.25	0.26			√	
b5	128	4	1.0E-03	32	2000	65.48	2.82	14.29	0.25			√	
C1	64	2	1.0E-01	8	2000	64.42	2.89	13.2	0.26		√		√
C2	64	2	1.0E-01	24	2000	64.42	2.89	13.2	0.26		√		√
C3	96	3	1.0E-01	8	2000	64.42	2.89	13.2	0.26		√		√
C4	96	3	1.0E-02	16	2000	64.42	2.89	13.2	0.26		√		√
C5	128	4	1.0E-03	32	2000	64.42	2.89	13.2	0.26		√		√
c1	64	2	1.0E-01	8	2000	64.42	2.89	13.2	0.26		√		
c2	64	2	1.0E-01	24	2000	64.42	2.89	13.2	0.26		√		
c3	96	3	1.0E-01	8	2000	64.42	2.89	13.2	0.26		√		
c4	96	3	1.0E-02	16	2000	64.42	2.89	13.2	0.26		√		
c5	128	4	1.0E-03	32	2000	64.42	2.89	13.2	0.26		√		

5.3 Results Comparison of Time Sequence Information

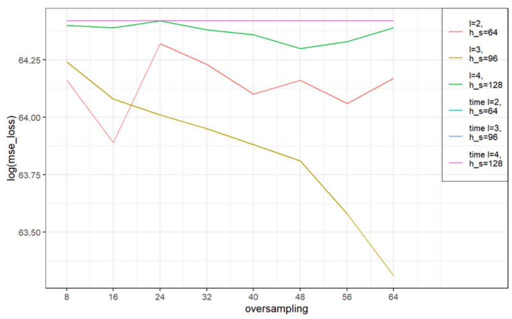

Fig. 5 Comparison of Settings in LSTM

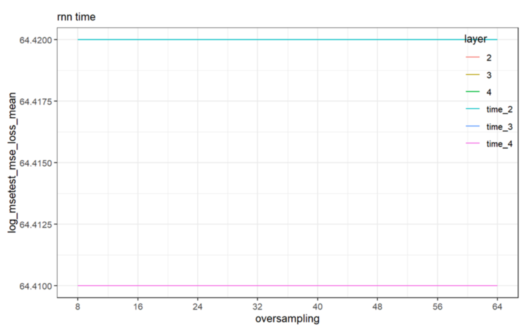

Fig. 6 Comparison of Settings in RNN

For LSTM (Fig. 5), four time zones are added as time sequence information, which makes the loss larger. Because each time zone is characterized by 50000 time samples, compared with the average value of 200000 time data in four time zones, it cannot reflect the characteristics of each sample well. Because the LSTM model preserves the information of the previous time series data, the inaccurate information will accumulate step by step, which makes the loss larger. Especially in level 3, the amount of inaccurate information increases, leading to a large increase in loss.

In contrast, the RNN model cannot learn the long-distance dependency well, so adding inaccurate time sequence information has little effect on loss. On the contrary, due to the transmission of time sequence information, RNN learns more features of training samples, which reduces loss. (Fig. 6)

5.4 Comparison of LSTM and RNN in Processing Time Sequence Data

The two losses are almost equal when the timing information is added, and the loss is insensitive to the oversampling number.

LSTM model can solve the gradient disappearance in the long sequence training, which makes the recurrent neural network have stronger memory performance to better deal with long sequence problems. However, only four time zones of data are selected in this study and the LSTM model does not play a good role in dealing with long sequences, so the results of the two models are similar.

6. Conclusion

Relevant features from the data of electroencephalogram and electromyogram of dystonia are extracted in few-shot samples to predict the score of the preoperative scale.

In this study, features of the time domain and frequency domain are extracted from EEG and EMG signals. This study innovatively proposes to extract brain lateralization features from local field potentials (LFP) of STN and GPi nuclei of left and right brains as inputs of neural networks, predicting dystonia scores through the linear model, RNN, and LSTM models respectively. At the same time, four features of time domain are added to RNN and LSTM models as time sequence information for comparison. It is manifested that the addition of lateralization features influences Linear Regression and LSTM model training to some degree, which shows the effectiveness of lateralization feature extraction. Besides, LSTM and RNN models have more advantages in predictions related to time sequence data.

Due to the limited training set, this study adopts a random oversampling method to expand the training set, which may lead to over-fitting. We can add noise to enhance the data of the training set to obtain better training results. In the aspect of adding time sequence information, we can sample more time sequence intervals and increase the influence of time sequence information on RNN and LSTM. In the aspect of feature extraction, we can conduct big data mining to further extract signal features. Predicting the score of the preoperative scale is to extract the relevant features from the data of electroencephalogram and electromyogram of dystonia in few-shot samples.

The use of deep learning in predicting muscle tone disorder scores has shown promising results. However, there are opportunities for further expansion in this area. One potential avenue is to incorporate additional data sources, such as patient demographics or medical histories, to enhance the accuracy of the prediction model. Another possibility is to explore the use of different deep learning architectures, such as ResNet and U-Net, to improve the predictive power of the model. Additionally, the application of transfer learning, where a pre-trained model is fine-tuned for a specific task, may also be useful in improving the accuracy of the prediction model. Overall, the potential for using deep learning to predict muscle tone disorder scores is vast and there are many avenues for further exploration and development in this area.

References

1. Balint, B., et al., Dystonia. Nat Rev Dis Primers, 2018. 4(1): p. 25.
2. Biering-Sorensen, F., J.B. Nielsen, and K. Klinge, Spasticity-assessment: a review. Spinal Cord, 2006. 44(12): p. 708-22.
3. Oh, S.L., et al., A deep learning approach for Parkinson’s disease diagnosis from EEG signals. Neural Computing and Applications, 2018. 32(15): p. 10927-10933.
4. Cho, K., et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. 2014. arXiv:1406.1078 DOI: 10.48550/arXiv.1406.1078.
5. Cristian Borges Gamboa, J. Deep Learning for Time-Series Analysis. 2017. arXiv:1701.01887 DOI: 10.48550/arXiv.1701.01887.
6. Kiranyaz, S., et al. 1-D Convolutional Neural Networks for Signal Processing Applications. in ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2019.
7. Lawhern, V.J., et al., EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces. J Neural Eng, 2018. 15(5): p. 056013.
8. Schirrmeister, R.T., et al., Deep learning with convolutional neural networks for EEG decoding and visualization. Hum Brain Mapp, 2017. 38(11): p. 5391-5420.
9. Xia, P., J. Hu, and Y. Peng, EMG-Based Estimation of Limb Movement Using Deep Learning With Recurrent Convolutional Neural Networks. Artif Organs, 2018. 42(5): p. E67-E77.
10. Xiong, D., et al., Deep Learning for EMG-based Human-Machine Interaction: A Review. IEEE/CAA Journal of Automatica Sinica, 2021. 8(3): p. 512-533.
11. Vinyals, O., et al. Matching Networks for One Shot Learning. 2016. arXiv:1606.04080 DOI: 10.48550/arXiv.1606.04080.
12. Ren, M., et al. Meta-Learning for Semi-Supervised Few-Shot Classification. 2018. arXiv:1803.00676 DOI: 10.48550/arXiv.1803.00676.
13. Corballis, M.C., Left brain, right brain: facts and fantasies. PLoS Biol, 2014. 12(1): p. e1001767.
14. Fisher, R.A., Statistical Methods for Research Workers. 1992, Springer New York. p. 66-70.
15. Rumelhart, D.E., G.E. Hinton, and R.J. Williams, Learning representations by back-propagating errors. Nature, 1986. 323: p. 533-536.
16. Hochreiter, S. and J. Schmidhuber, Long Short-Term Memory. Neural Computation, 1997. 9(8): p. 1735-1780.