Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: 24 Apr 2020
Article

EEG Data

Predicting Motion Regimes Through Machine Learning Methods with EEG Data

Ramezani, Sepehr [1,*], Goodwin, Derek [2] and Griswold, Luke [2]

[1] Mechanical and Aerospace Engineering, University of Central Florida, Orlando, 32816, USA and
[2] Computer Science, University of Central Florida, Orlando, 32816, USA.

[*] To whom correspondence should be addressed.

Abstract

Motivation: Electroencephalography (EEG) signals have been known as an important non-invasive tool in neuroscience. The vigorous classification of these signals is a significant contribution towards applying EEG in more practical applications. The previous studies of EEG have demonstrated the possibility of classification of walking speed. In this study, two different algorithms were developed based on the time and frequency domain to achieve better accuracy.
Results: Two models were very successful in learning the EEG patterns associated with an individual's motions with accuracy of .98 or greater, however both models were unsuccessful in predicting a novel individual's EEG patterns from other people's data.
Availability: https://github.com/Kickflip89/EEG-U-Time and https://github.com/Deriino/EEG_model_WIP
Contact: sepehr.ramezani@knights.ucf.edu

1 Introduction

Using Electroencephalography (EEG) signals to identify brain-behavior during different activities has been a topic under discussion in for a long time. EEG signals represent the electrical activity of the brain with analog time-series signals which are recorded by several electrodes placed on the scalp.

Research has been conducted on analyzing EEG signals for non-movement tasks in the order to restrict participants from extra movement during the trial and help to reduce the artifact signals. Classifying the EEG signals during listening to music (Lin *et al.*, 2008) and during a visual working-memory experiment (Artoni *et al.*, 2018) are a few examples.

However, looking at the EEG signal during movement has received researchers' attention in recent years. This field of study covered upper-limb motion like hand gesture detection (Olsson *et al.*, 2019; Li *et al.*, 2017) as well as the lower-limb body. Zhang et al proposed the investigation method of brain dynamics during human motion and the developed wearable mobile brain/body imaging (MoBi) methodology (Zhang *et al.*, 2017). Then, Gwin et al introduced a new method to process EEG data during walking and running (Gwin *et al.*, 2010). These methods provide a better understanding of EEG signals for scientists which put their focus on human electro-cortical brain dynamics during mobile activities. A study on the gate state of a small group of healthy and stroke patients during ground walking (Zhang *et al.*, 2017) and investigation of the possibility of EEG decoding for different locomotion task of developing children (Luu *et al.*, 2019) are examples.

The big challenge for analyzing EEG signals is an artifact associated with locomotion which is time-varying in sort-term and long-term variability (Hausdorff *et al.*, 1995; Jordan *et al.*, 2006, 2007). So, dealing with artifact signals which consist of non-cortical signals such as muscle activity, cardiac signal, and eye movement has been studied by different methods (Oliveira *et al.*, 2016; Nordin *et al.*, 2018). Using a machine learning algorithm has introduced as a high-performance approach for decoding the EEG signal as well (Lee *et al.*, 2017; Bengio *et al.*, 1994). There are some important factors addressed with EEG decoders such as functionality, feature extraction, and neural representations (Glaser *et al.*, 2017). To achieve these factors, artificial neural network (ANN) (Nazmi *et al.*, 2019) as well as long short-term memory (Ding *et al.*, 2018) are used to decode gait events from EEG. Rahrooh (Rahrooh, 2019) used different machine learning algorithms to classify and predict walking speed from EEG Data with moderate success.

In this paper, two different algorithms based on time domain and frequency domain are developed to improve the accuracy of state classification. Also, a novel data set is used in this study to make it resemble daily activity in comparison with other studies. Specifically, an adjustable treadmill is employed to create an incline and decline situation. The performance of these two models has been compared with each other and Rahrooh's study (Rahrooh, 2019).

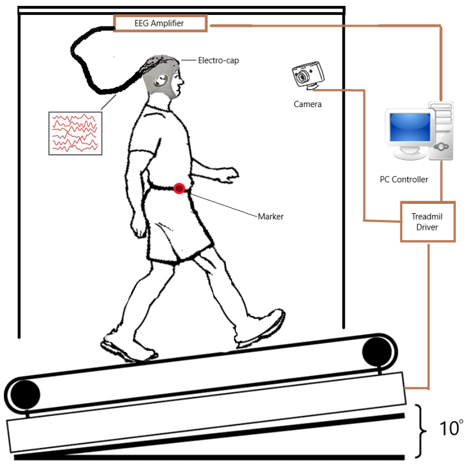

Fig. 1. Schematic of experimental setup: Subjects walked on a dual-belt treadmill with a force plate. EEG signals are recorded by electro-cap then amplified before entering in control unit system. One camera captures marker's position for self-pace mode.

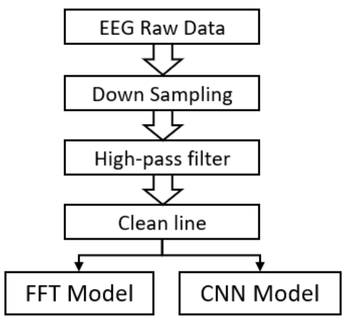

Fig. 2. Data preparation process

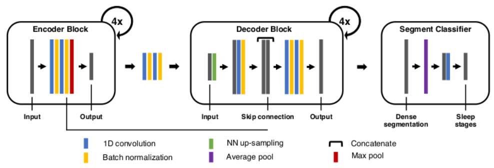

Fig. 3. CNN model Architecture (Perslev et al., 2019)

2 Methods

EEG data was recorded using a 128-channel system (BioSemi, ActiveTwo, Amsterdam, Netherlands) from young and healthy participants (n =10) during walking on an adjustable treadmill in different conditions Huang (2018). To achieve a stable signal, a conductive gel is used to fill the gap between electrodes and the skull. Participants walked on a dual-belt treadmill (Motek Forcelink M-Gait) for approximately 5 minutes at five different walking speeds 0.5 m/s, 0.75 m/s, 1.0 m/s, 1.25 m/s and self-paced speed. In the self-paced mode, the speed of treadmill and participants is synchronized by the controller unit to keep participants at the center of the treadmill which means they can change their speed during the test. One camera is employed (see figure 1) to measure the position of the body in self-pace mode by observing a marker that is attached to the body. Moreover, walking with a fixed speed of 0.75 m/s and self-pace is repeated for a 10 deg incline and decline setting. The sample rate of raw data is reduced from 500Hz to 150Hz to gain better validation accuracy of the decoding process (Nakagome *et al.*, 2020). Also, a high pass filter with a 0.5 cut off frequency is used to remove dc as well as clean line filter to reduce noise and remove AC power noise. (figure 2)

2.1 Encoder-Decoder Approach (CNN Model)

2.1.1 Model Architecture

The researchers surmised that EEG signals controlling our movement such as the gait of a person would originate from the motor cortex, and could therefore manifest in any, all, or some subset of the 128 channels. As such, an intuitive way to think of the problem was as a 1D convolution through all 128 channels searching for a learned pattern from different participants through the time domain. It was decided to use slices of 2.5 seconds as this should catch enough of the gait of an individual, and the only question was what kind of architecture would be required. Although several papers exist on using ML methods with EEG data, almost none are specifically trying to classify regimes of motion without other input (for example from EMG data (Olsson *et al.*, 2019)). The closest is a survey about gait decoding which is targeting hip angle and is not a classification problem (Nakagome *et al.*, 2020) and Rahrooh's master thesis (Rahrooh, 2019).

Perslev et. al did, however, solve a similar problem in classifying sleep stages through EEG data. The authors used a network they called U-Time (Perslev *et al.*, 2019). This network is an encoder-decoder type network that uses max-pooling in the encoder section with skip connections to the decoder's nearest-neighbor (NN) up-sampling to encode all the data within a set time window into a single chunk. A decoder then reconstructs the signal with skip connections from the encoded segment. The details can be found in figure 3. A standard classifier then outputs a probability distribution across the classes with either a categorical cross entropy or generalized dice loss function.

2.1.2 Data Preparation

After some consideration, the researchers decided to encode roughly 2.5 second time intervals using the encoder section. At 150Hz, this equated to Max Pooling layers of 8, 6, 4, and 2, which required a minimum size of 384 time steps, or 2.56 seconds. This ensures a full gait should be captured regardless of the start or end time of the steps and should therefore be present for the convolutional filters to find. Originally 9 filters of size 5 were chosen to mirror Perslev et. al, but once it became apparent a higher multiple of 9 might be required, the kernel size of 7 was used in order to have 2^7 possible combinations of zero and non-zero elements. The next question was whether to adopt strided or dilated convolutions. In order to explore slightly different frequency domains while not missing any time steps, it was decided to use dilated convolutions with a stride of 1.

Following the considerations of Mrozik et. al, additional preparation was done on the raw data in the manner CNNs typically used for EEG signal (Mrozik *et al.*, 2017). First, they were normalized on a scale from 0 to 511 for each individual channel and cast as integers for faster training. The data were then characterized by the number of discrete time measurements in order to help with batch generation. The general training process would be to keep no more than nine files open at a time (one for each class) in order to avoid memory issues. At the beginning of a training session, the files would be split into as many size 384 chunks as possible and shuffled. For experiments that required multiple participants' data, a new file would be opened when all the usable 384 wide chunks had gone to batches. Training was then conducted in batches of 45 (five chunks per file / class). Information about which files and chunks had been used were then reset before the next epoch began. The major features of EEG signals that are difficult to process with convolution are the different frequency domains, so dilated convolution was used with the dilation rate as a hyperparameter

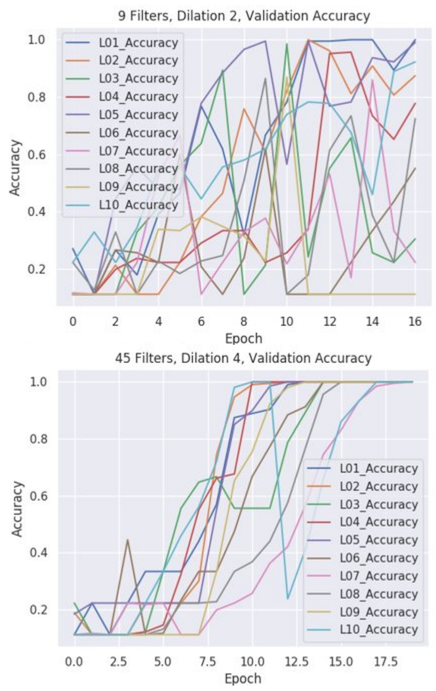

Fig. 4. Hyperparameter tuning by training individual CNN models for each participant: 45 Filters of kernel size 7 dilated by 4 to 30 had the best results with a categorical cross-entropy loss function.

to explore different frequency regimes (higher dilation tries to focus on lower frequencies). The general definitions of EEG signals are divided into different frequency ranges and are shown in table 1 (Mrozik *et al.*, 2017). As described in the Metrics section, the model architecture was tuned to predict with high accuracy an individual participant's regime's of motions. The following parameters were tested:

- Convolutional Filters: {9,18,27,45,90}
- Dilation: {1,2,4,8}
- Loss: Categorical Cross-Entropy

The best parameters were chosen as 45 filters with a kernel size of 7, dilated by 4 with a categorical cross-entropy loss function. The kernel size of 7 dilated to 4 spans about 0.2 seconds worth of data and should resolve frequencies below 19Hz. The training process was then optimized by the Adam optimizer. An example of the optimized model can be shown in figure 4.

2.2 Frequency Domain Approach (FFT Model)

2.2.1 Model Optimization

Knowing that relying on many learnable parameters to reduce the data's dimensionality may prove difficult Keogh and Mueen (2017), there was an assumed possibility before doing any experiments that learning from the signals directly would not work in the case of EEG signals for this project.

One way to attempt to overcome this is to use Fast Fourier Transform (FFT). FFT is a way of transforming data from the time domain to the frequency domain; this exposes the frequency ranges of the EEG signals displayed in 1, and would, in theory, expose what type of signals comes out of each participant. Craik et al stated that, on EEG data, recurrent neural network architectures outperformed architectures using elements from both RNNs and CNNs Craik *et al.* (2019). During testing and deciding the model, however, the use of two 1-dimensional convolutional layers

Table 1. EEG frequency

Wave	Frequency Range
Alpha	8 - 13 Hz
Beta	13 - 30 Hz
Gamma	Above 30 Hz
Delta	0.5 - 4 Hz
Theta	4 - 8 Hz

before the recurrent layer was found to be more effective on the validation set than the architecture with just the recurrent layer.

To generate one training sample for the data, 272 time slices – about 1.81 seconds of data with downsampled 150Hz input – are randomly selected from one of the participants' data. Ten chunks of 128 time slices are created from this; each chunk, other than the first, shares 112 time slices with the chunk before it. Then the chunks are generated, each chunk is transformed via FFT of length 128, of which the magnitudes of the 64 channels are used as input to the frequency domain model.

The model takes the FFT halves and passes them through two convolutional layers, one fully-connected layer, one recurrent layer, and one fully-connected layer, respectively. The fully-connected layer outputs to LogSoftmax and trains using multi-class cross entropy loss.

Leaky ReLU of negative slope 0.2 was used for the activation function in each layer. Batch Normalization was applied before each leaky ReLU and after each convolutional block. Adam optimizer with the default betas were used.

Each batch contains 90 randomly-sampled training samples per batch - each with 10 FFT chunks. There are about 50 thousand samples per file in 81 training files in total. This means each "epoch" is about 165 batches in leave-one-out. To reduce training time, instead of using the entire test set for each evaluation, 270 samples were selected randomly for evaluation periodically during testing. The accuracy for a single batch is defined as the percentage of correct guesses on the last time chunk.

2.3 Metrics

Initial experiments were geared at two methods: Maximizing the accuracy by training a model on each participant's data, and maximizing the accuracy by training a model on a training set of all the data. Both the FFT and CNN approaches yielded high accuracy when a model was trained on an individual participant's data. Next, the researchers evaluated how the model performed on test sets sampled from the full data set. These experiments were used to tune the hyperparameters for the models. The next round of experiments were geared toward attempting to predict the regime motions of one participant by training on the other nine participants' data. Both the FST and CNN models encountered significant difficulty predicting motion regimes from participants that were not in the training set.

At this point, further effort was put into seeing if a model could improve on predicting a novel participant's data. Further training and hyperparameter tuning was done to attempt to improve the baseline CNN model. Another approach was to attempt to use weighted and unweighted ensemble classifiers by majority vote and aggregated activations. In all cases, accuracy was the metric used to evaluate the model performance. As there were 10 participants, it yielded well to using 10-fold cross-validation for each experiment.

Experiment	CNN Accuracy	FFT Accuracy
Individual	100.0%	98.18%
Leave-One-Out	21.0%	22.59%

Table 2. Experimental results with 10-fold cross validation. Individual trains a model on a single participants' data, Leave-One-Out trains a model on 9 participants' data and validates on the tenth. Mean accuracy is shown for both the CNN and FFT models.

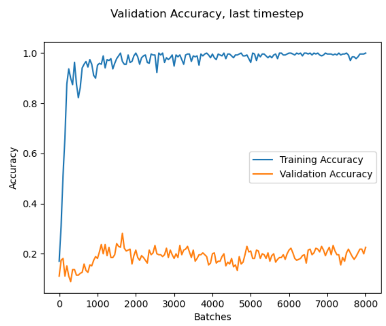

Fig. 5. FFT model's test and validation accuracies for leave-one-out experiment.

3 Results

Table 2 compares the accuracies of our models. The CNN model outperforms the FFT model on individual tests, while the FFT model outperforms the CNN model on LOO tests.

Both the frequency domain approach and the encoder-decoder approach encountered success at building models based one participant's data with high accuracy on the validation set. More trouble was encountered mixing all the data, as the validation loss and accuracy were not converging as the training loss and training accuracy improved. The decision was then made for the CNN model to use early stopping and model check pointing in order to save the model with the best accuracy on the validation set. In order to ensure this method resulted in actual training, a new validation set was generated from a single participant for the ten-fold cross validation.

The hardest problem was to achieve decent accuracy by withholding one participant's data from the training set and using a sample of that participant's data as validation. With 9 classes, the baseline for a random or constant classifier would be an accuracy of 11.1%.

With the baseline results, a few considerations still needed to be determined. The first question was if the LOO accuracy could be improved via ensemble methods. In order to find out, the CNN model was trained with 10 individual models, one for each participant. Then the other 9 models would attempt to predict a sample of the 10th participant's data via a majority vote. This method actually under-performed a random classifier with an average accuracy over the 10 participants of 9.7%. During the FFT model's individual experiments, the training data and validation data were separated into chunks so that they do not interfere with each other. Despite this, the FFT models of the individual experiments still show much higher validation accuracy than the FFT models of the LOO experiments. Figures 6 and 5 illustrate the experiments' performance.

The FFT model is a recurrent network; there are ten attempts for the model to guess the label of the data it is given. However, the model is supposed to use information from previous timesteps to estimate with a higher accuracy after each time step. This is why the model's accuracy score derives only from the tenth such output. Table 3 shows the test and validation accuracies on the last timestep on 270 random test cases from both the training and test sets. In general, and especially over the first few timesteps, this accuracy increases as the model gains more knowledge as to what it thinks the output is. Even on the training set, it starts out relatively low, but then it increases to 100 percent accuracy gradually. On the test set, the accuracy peaks on the fifth time step.

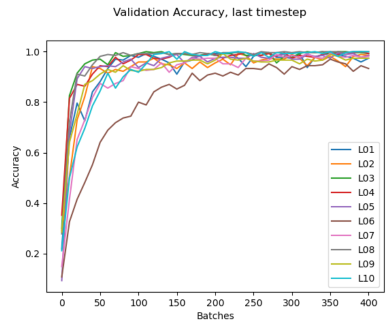

Fig. 6. FFT model's validation accuracies for individual experiments. Each participant in this graph is referred to as L01 thru L10.

Accuracy on step	1	2	3	4	5	...	10
Training	90.37	94.44	95.56	97.41	99.26	...	100.00
Validation	20.37	21.85	21.85	22.59	23.33	...	22.59

Table 3. The FFT model accuracy on the last evaluation of data on the LOO test. Each step represents shifting over the 128-sample window by 16 samples. Not all of the columns are shown.

Predicted Class	D1	DS	I1	IS	L1	L2	L3	L4	LS
D1	0	0	0	0	0	0	0	0	0
DS	0	23	0	0	0	0	0	0	0
I1	12	0	22	0	23	0	0	0	0
IS	0	0	0	0	0	0	0	0	0
L1	0	0	0	0	0	0	0	0	0
L2	0	0	0	23	0	23	0	0	0
L3	0	0	0	0	0	0	0	0	0
L4	0	0	1	0	0	0	23	23	23
LS	11	0	0	0	0	0	0	0	0

	D1	DS	I1	IS	L1	L2	L3	L4	LS
D1	23	0	23	18	5	0	0	23	0
DS	0	3	0	5	0	0	2	0	0
I1	0	17	0	0	18	0	12	0	22
IS	0	3	0	0	0	0	0	0	0
L1	0	0	0	0	0	0	0	0	0
L2	0	0	0	0	0	23	0	0	1
L3	0	0	0	0	0	0	0	0	0
L4	0	0	0	0	0	0	8	0	0
LS	0	0	0	0	0	0	1	0	0

Table 4. Confusion Matrices of CNN model with a sample of participant 5's data as a validation set. Predicted class on vertical axis, actual class on horizontal axis (D1: Decline 0.75m/s, DS: Decline Self-Pace, IS: Incline 0.75m/s, L2: Level 0.5m/s, L3:Level 0.75m/s, L4:Level 1m/s, L4:Level 1.25m/s LS: Level self-pace). TOP: Leave-One-Out method where a single model was trained on the other 9 participants. BOT: Weighted ensemble method where individual models' raw activations (pre-softmax) were added.

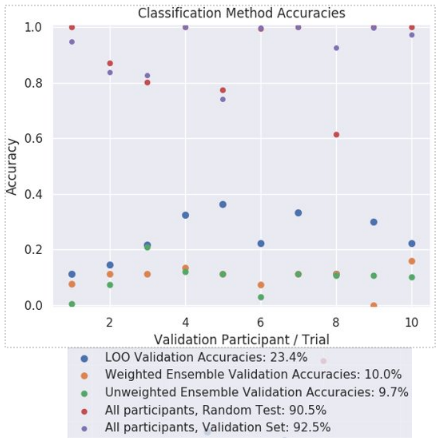

Fig. 7. CNN model results from 10-fold cross-validation using different training / model strategies. LOO is where the model is validated on the participant in the x-axis but trained on the others. Weighted and Unweighted Ensemble are majority-vote from 9 models trained on the other participants (weighted are aggregated activations, unweighted are aggregated distributions). All participants are trained on a random 80% of the data and validated on either the remaining 20% (validation) or a sample of a newly generated test set from a single participant (random test).

Because this ensemble method was aggregating probability distributions, it is an unweighted ensemble classifier by majority vote. An attempt to improve this method was to reconstruct the network by replacing the final softmax activation with a ReLU activation. In this way, it weights higher activations in different models before aggregating all of the models' activations. This did slightly improve the average accuracy over the 10 participants, but only to 10%, which still is below the expected accuracy of a random classifier.

The next question was why the models are classifying the participant's data in the LOO and ensemble methods so poorly. There is an obvious physical relevance to this question: people are highly variable, and a person's gait is likely based off of many physically relevant factors not present in the data set such leg length, etc. From a theoretical standpoint, it may simply be that 10 participants are not enough for a successful ensemble predictor as it does not sufficiently explore the space of motion among individual variability. One clue can be found in the confusion matrices, an example of which is shown in table 4. It appears that the LOO classifiers more often predict a validation class into a single predicted class. They both do not use all of the classification space (5 classes in each with zero or very few predictions).

Another question was if training on all participants' data, the model could successfully learn to classify every participant accurately. In order to accomplish this, the CNN model was used with model checkpointing and early stopping. In order to avoid picking a random good version of the model, since the loss and accuracy weren't smoothly converging, the best versions of these models were tested with a randomly generated test set (this was repeated 10 times, with the random test set generated from a different participant each time. The results of these experiments can be seen in figure 7.

4 Discussion and Comparison

The frequency domain approach and the CNN-based encoder-decoder approach offer a good coverage of the EEG interpretation problem. The CNN-based approach is locked in to certain frequencies and time windows based on the dilation and kernel size but is able to recognize certain shapes in the waveform in each channel. In contrast, the FFT-based approach is able to explore the entire frequency domain while individual waveforms are only recognized by their contribution to frequency amplitudes. In this way, the authors felt an overarching approach to the problem was achieved that would complement each other. The fact that both models are able to predict with high accuracy an individual's motion regimes when trained on that individual's data shows that this is probably not a difficult problem to solve. The problem of predicting an individual's motion regimes based on *other people's data* remains a difficult and unsolved problem with both models.

An idea to improve accuracy was to analyze if any of the 128 EEG channels had a disproportionate effect on training the models. In order to do this, a test set was generated for each participant's individual model. For one channel at a time, the model predicted the classes while that channel's data was replaced with random numbers on the interval [0,511] for 50 trials. There were a few channels that were in the lowest 10 mean accuracy for more than one model, but the accuracy only fell by at most .04, with at most a standard deviation of .002. Admittedly, the encoder-decoder approach is probably robust to input modification of this nature so this technique may still be useful for a different model, but it appears there is not a single channel contributing greatly to the model's performance for each participant.

It is likely that 9 participants are not enough to sufficiently generalize to an individual's motion regimes. This is evident in the failure of the Leave-One-Out methods and the weighted and unweighted ensemble methods to achieve much more than correctly predicting one or two classes correctly. When training on all data, a moderate amount of success was achieved with the EEG model of about 90% accuracy.

(Rahrooh, 2019) performed most recent study by predicting walking speeds from EEG data. However, this study differs from the current approach in that all participants were on level treadmills and there were 7 vice 10 participants, but Rahrooh tried many different traditional ML methods such as random forrest, naive Bayes, bagging, and boosting. They were able to achieve precision and sensitivity between 60% and 80% with these methods while trying to predict based on a subset of all participants' data. A direct comparison is difficult because Rahrooh predicted five classes with seven participants using precision, sensitivity, and recall, but the authors feel this method surpasses those results based on the accuracy of 9 classes over 10 participants.

5 Conclusion

Human beings are highly variable creatures and EEG data is no exception. The authors had good success in predicting motion regimes for an individual person, and even moderate success in predicting them for several people when everyone's data was in the training set. The problem of predicting motion for an individual based on a community of other people's data remains largely unsolved. It could be that nine participants is not enough data to successfully generalize motion regimes to a stranger, or it could be that machine learning algorithms will need data from an individual in order to learn their EEG motion patterns. However,results of proposed algorithm shows improvement in validation accuracy by 10% in compassion of previous study.

Regardless, this was a good first step and paired with more information (such as from EMG sensors or even height/weight/age information) these networks may be able generalize more successfully.

References

Artoni, F., Delorme, A., and Makeig, S. (2018). Applying dimension reduction to EEG data by Principal Component Analysis reduces the quality of its subsequent Independent Component decomposition. *Neuroimage*, **175**, 176–187.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, **5**(2), 157–166.

Craik, A., He, Y., and Contreras-Vidal, J. (2019). Deep learning for electroencephalogram (eeg) classification tasks: a review. *Journal of Neural Engineering 16*.

Ding, Z., Yang, C., Xing, K., Ma, X., Yang, K., Guo, H., Yi, C., and Jiang, F. (2018). The real time gait phase detection based on long short-term memory. In *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, pages 33–38.

Glaser, J. I., Chowdhury, R. H., Perich, M. G., Miller, L. E., and Kording, K. P. (2017). Machine learning for neural decoding. In *2008 IEEE 10th Workshop on Multimedia Signal Processing*, pages 127–130.

Gwin, J. T., Gramann, K., Makeig, S., and Ferris, D. P. (2010). Removal of movement artifact from high-density EEG recorded during walking and running. *J. Neurophysiol.*, **103**(6), 3526–3534.

Hausdorff, J. M., Peng, C. K., Ladin, Z., Wei, J. Y., Goldberger, A. L., and Goldberger, A. L. (1995). Is walking a random walk? Evidence for long-range correlations in stride interval of human gait. *J. Appl. Physiol.*, **78**(1), 349–358.

Huang, H. J. (2018). Biomechanics, rehabilitation, and interdisciplinary neuroscience (brain) lab. University of Central Florida.

Jordan, K., Challis, J. H., and Newell, K. M. (2006). Long range correlations in the stride interval of running. *Gait Posture*, **24**(1), 120–125.

Jordan, K., Challis, J. H., and Newell, K. M. (2007). Walking speed influences on gait cycle variability. *Gait Posture*, **26**(1), 128–134.

Keogh, E. and Mueen, A. (2017). *Curse of Dimensionality*, pages 314–315. Springer US, Boston, MA.

Lee, K., Liu, D., Perroud, L., Chavarriaga, R., and del R. Millán, J. (2017). A brain-controlled exoskeleton with cascaded event-related desynchronization classifiers. *Robotics and Autonomous Systems*, **90**, 15 – 23. Special Issue on New Research Frontiers for Intelligent Autonomous Systems.

Li, X., Samuel, O. W., Zhang, X., Wang, H., Fang, P., and Li, G. (2017). A motion-classification strategy based on semg-eeg signal combination for upper-limb amputees. *Journal of NeuroEngineering and Rehabilitation*, **14**(1), 2.

Lin, Y., Wang, C., Wu, T., Jeng, S., and Chen, J. (2008). Support vector machine for eeg signal classification during listening to emotional music. In *2008 IEEE 10th Workshop on Multimedia Signal Processing*, pages 127–130.

Luu, T. P., Eguren, D., Cestari, M., and Contreras-Vidal, J. L. (2019). Eeg-based neural decoding of gait in developing children. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 3608–3612.

Mrozik, K., Kostek, B., Kurowski, A., and Czyzewski, A. (2017). Comparison of selected electroencephalographic signal classification methods. In *Signal Processing Algorithms, Architectures, Arrangements, and Applications*.

Nakagome, S., Luu, T., He, Y., Ravindran, A., and Contreras-Vidal, J. (2020). An empirical comparison of neural networks and machine learning algorithms for eeg gait decoding. *Scientific Reports*.

Nazmi, N., Rahman, M. A. A., Yamamoto, S.-I., and Ahmad, S. A. (2019). Walking gait event detection based on electromyography signals using artificial neural network. *Biomedical Signal Processing and Control*, **47**, 334 – 343.

Nordin, A. D., Hairston, W. D., and Ferris, D. P. (2018). Dual-electrode motion artifact cancellation for mobile electroencephalography. *Journal of Neural Engineering*, **15**(5), 056024.

Oliveira, A. S., Schlink, B. R., Hairston, W. D., K?nig, P., and Ferris, D. P. (2016). Induction and separation of motion artifacts in EEG data using a mobile phantom head device. *J Neural Eng*, **13**(3), 036014.

Olsson, A., Sager, P., Andersson, E., Bjorkman, A., Malesevic, N., and Antfolk, C. (2019). Extraction of multi-labelled movement information from the raw hd-semg image with time-domain depth. *Scientific Reports*.

Perslev, M., Jensen, M., Darkner, S., and Jennum, P. (2019). U-time: A fully convolutional network for time series segmentation applied to sleep staging. In *33rd Conference on Neural Information Processing Systems*.

Rahrooh, A. (2019). *Classifying and Predicting Walking Speed From Electroencephalography Data*. Ph.D. thesis, University of Central Florida.

Zhang, D., Yao, L., Zhang, X., Wang, S., Chen, W., and Boots, R. (2017). Eeg-based intention recognition from spatio-temporal representations via cascade and parallel convolutional recurrent neural networks. *ArXiv*, **abs/1708.06578**.