nEMGNet: Voting based Deep Learning Needle Electromyography Diagnosis Prediction

Jaesung Yooa+, Ilhan Yoob+, Ina Younc, Sungmin Kimd, Ri Yue, Kwangsoo Kimf, Keewon Kime\*, and Seung-Bo Leeg\*

a School of Electrical Engineering, Korea University, Seoul, Republic of Korea

b Department of Neurology, Nowon Eulji Medical Center, Eulji University School of Medicine, Seoul, Republic of Korea

c Department of Computer Science, New York University, NY, USA

d Department of Neurology, Seoul National University Hospital, Seoul National University College of Medicine, Seoul, Republic of Korea

e Department of Rehabilitation Medicine, Seoul National University Hospital, Seoul, Republic of Korea

f Transdisciplinary Department of Medicine & Advanced Technology, Seoul National University Hospital, Seoul, Republic of Korea

g Office of Hospital Information, Seoul National University Hospital, Seoul, Republic of Korea

\*[keewonkimm.d@gmail.com](mailto:keewonkimm.d@gmail.com), \*[koreateam23@korea.ac.kr](mailto:koreateam23@korea.ac.kr)

+Authors contributed equally to this work

## Abstract

Needle electromyography (nEMG) is an electrophysiological test which measures electric signals generated from a muscle with an invasive needle. Characteristics of nEMG signals are used to diagnose types of peripheral neuropathy, which refers to the damage in the peripheral nervous system. An electromyographer manually inspects the nEMG signals in clinical diagnosis of peripheral neuropathy, making the process highly dependent on the experience of the electromyographer. To aid clinical diagnosis using nEMG, this study introduces nEMGNet and DiVote algorithm. Proposed nEMGNet is a one-dimensional convolutional neural network which extracts features from raw nEMG signals, and DiVote algorithm is a subject diagnosis prediction pipeline to integrate heterogeneous signal data structure for each subject. 376 nEMG signals measured from 57 subjects between June 2015 to July 2020 in Seoul National University Hospital were used to experiment our method. Results showed that nEMG segment prediction accuracy of proposed nEMGNet was 62.35%, and subject diagnosis accuracy of nEMGNet and DiVote algorithm was 83.69 % over 5-fold cross validation. Proposed nEMGNet outperformed all baseline nEMG classification models, and the features which causes nEMGNet to predict certain diagnosis labels were identified by applying feature visualization.

Keywords: Needle Electromyography, Clinical Diagnosis Prediction, Deep learning, Convolutional Neural Network, Data Heterogeneity, Explainable AI

# Introduction

Neurons are the electric circuits of the human body. However, the invaluable biological circuits may break down due to diabetes [1], chemotherapy [2], or even unknown reasons [3, 4] resulting in peripheral neuropathy [5]. In clinical diagnosis of peripheral neuropathy, needle electromyography (nEMG) has been widely used. nEMG is an electrophysiological test which records electrical activity generated from nerves, muscles, and neuromuscular junctions by inserting a needle into a muscle at rest or during muscle contraction [6-10]. Based on the abnormalities in the measured nEMG signal, a skilled electromyographer diagnoses types of peripheral neuropathy of the subject [6, 11]. While nEMG is an effective method in diagnosing the subtypes of peripheral neuropathy [12, 13], the current subjective method is highly dependent on the experience of the electromyographer, making it vulnerable to errors which can be identified from inter-rater reliability of 61-81% [14].

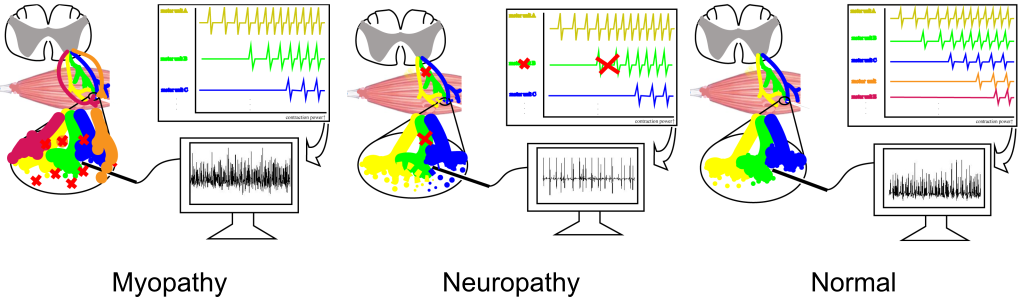
Among different types of machine learning models, deep learning [15] has shown outstanding performance by leveraging the power of large data in tasks that are hard to analyze mathematically [20-22]. There has been increasing movement to apply deep learning in medicine [16-19] including nEMG clinical diagnosis classification. Nodera et. al. [23] generated mel-spectrograms from nEMG signals and used pretrained image classification models to classify signals into six types of diagnosis labels. Nam et. al. [24] used plotted images of nEMG signals and used a pretrained image classification model to classify signals into three types of diagnosis labels.

While previous works have demonstrated the potential of deep learning applied in nEMG diagnosis classification, additional improvements are needed to apply deep learning in real nEMG clinical diagnosis. First, comparable performance is not guaranteed when using image classification models in signal classification tasks since the models were designed for natural image classification [25]. Second, previous works are focused on classifying the signals, not the individual subjects. However, in nEMG clinical diagnosis, signals of various lengths are measured from different types and number of muscles for each subject. To predict the diagnosis of subjects, an appropriate measure is necessary which integrates the heterogeneity in signal length, muscle sources, and number of signals. Finally, it is critical to identify how the machine learning model is making predictions, especially in medical applications [26, 27]. A machine learning model must be investigated to ensure the model is making predictions based on relevant features and not artifacts [28, 29], a topic which previous works did not deal with.

In this study, we propose nEMGNet, a one-dimensional convolutional neural network (CNN) model which can extract features from raw nEMG signals at improved accuracy compared to previous works. Motivation for nEMGNet was to build a domain specific deep learning model rather than to use models designed for other tasks. We also introduce the divide and vote (DiVote) algorithm, a method to predict the clinical diagnosis of a subject by integrating the heterogeneous muscle signals into a homogeneous form. By combining nEMGNet and DiVote algorithm, diagnosis of subjects was able to be predicted from heterogeneous data structure which allows practical adoption in clinical diagnosis. After the performance of nEMGNet and DiVote algorithm was verified, we identified how nEMGNet made predictions by applying feature visualization [30, 31], an explainable deep learning technique which shows the features a deep learning model has learned.

# Materials and Methods

## Data



**Figure 1** Needle electromyography signal for each peripheral neuropathy diagnosis types. Myopathy refers to the damage in the muscle fibers which results in motor unit action potentials with small amplitudes and short durations, and neuropathy refers to the damage in the peripheral nerves which shows motor unit action potentials with large amplitudes and long durations. nEMG signals from Normal muscles show medium amplitudes and durations compared to myopathy or neuropathy.

nEMG Data included a total of 57 subjects who visited Seoul National University Hospital from June 2015 to July 2020. Each subject was labeled with one of three diagnosis types by a certified electromyographer. Types of peripheral neuropathy were categorized into myopathy (M), neuropathy (N), and normal (NL). Characteristics of signals from each diagnosis type is presented in Figure 1. Single channel nEMG signals were sampled at 48kHz. Each subject contains a different number of signals acquired from different muscles, and each signal from each muscle was heterogeneous in length. The shortest signal was 0.41 second long and the longest signal was 4.00 seconds long. Each signal contained the muscle location information it was recorded from, which was one of proximal muscle (P) or distal muscle (D). Summary of the nEMG dataset is shown in Table 1. This study was approved by the Institutional Review Board (IRB) of Seoul National University Hospital (No. 2008-055-1147) and was conducted according to the Declaration of Helsinki and its later amendments. As the study is retrospective, informed consent was not obtained.

**Table 1** Summary of needle electromyography dataset

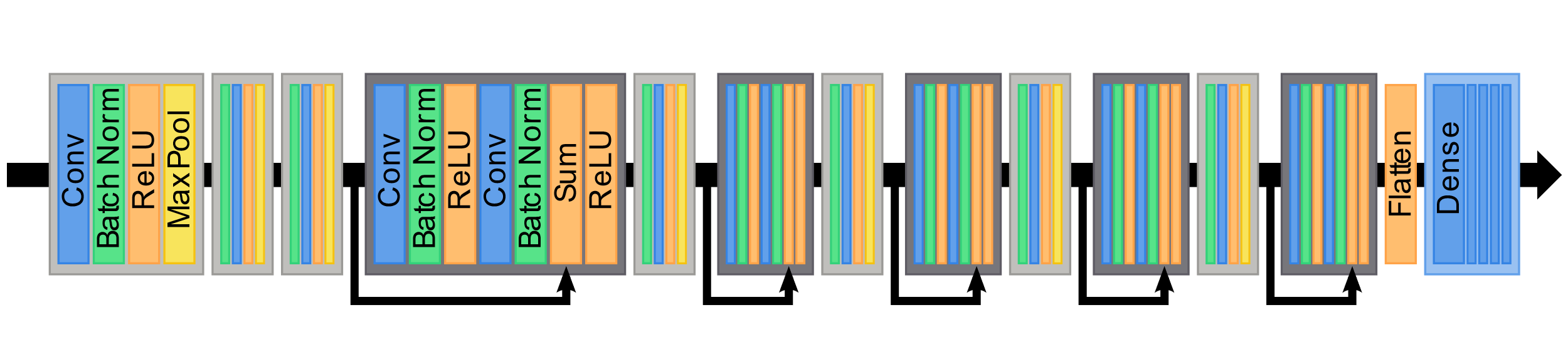
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Myopathy | Neuropathy | Normal | Total |
| Number of subjects | 19 | 19 | 19 | 57 |
| Number of signals | 122 | 160 | 94 | 376 |
| Proximal muscle signals | 64 | 63 | 17 | 144 |
| Distal muscle signals | 58 | 97 | 77 | 232 |
| Total signal length (sec) | 312.84 | 422.78 | 203.50 | 939.12 |

## Preprocessing

To avoid loss of information content [32] through transformations as well as to reduce the time taken in preprocessing, raw nEMG signals were used as input. To reduce computational complexity the signal was downsampled to 10kHz. Downsampled frequency was chosen after inspecting the morphology of downsampled signals under different frequencies. Each waveform was sliced into multiple segments by slicing the waveform with fixed window length and hop size . =0.4s, =0.1s was chosen empirically after experimentation. The resulting signal segment was 4000 samples in length. Each signal segment was labeled with the diagnosis label of the subject which the signal segment belonged to. After preprocessing, there were a total of 2700 myopathic signal segments, 3664 neuropathic signal segments, and 1706 normal signal segments.

## nEMGNet Model

As the task aims at capturing signal characteristics, one-dimensional convolutional neural network (1D-CNN) was designed. nEMGNet was designed based on the motives of VGGNet [33] and ResNet [34], neural network models which have demonstrated outstanding performance in image classification tasks. nEMGNet is composed of three types of blocks analogous to the building blocks of VGGNet and Resnet (Table 2). Spatial reduction block-1 (SR block-1) reduces the spatial resolution to 50%, and spatial reduction block-2 (SR block-2) reduces the spatial resolution to 25%. Residual block has residual connections which allows stable training in deep layers. [35]. As residual blocks can be repeated with arbitrary number of times, various versions of nEMGNet with different numbers of residual blocks were experimented. The configuration of nEMGNet versions is described in Table 3. In the fully connected layers that follow the convolutional layers, rectified linear unit (ReLU) activation function was applied for each hidden layer. Signal segments and diagnosis labels were used to train nEMGNet. Proximal and Distal muscle information was not used in training nEMGNet.



**Figure 2** nEMGNet model architecture. Blocks in light grey are spatial reduction blocks, and blocks in dark grey are residual blocks. The number of residual blocks between spatial reduction blocks vary in different types of nEMGNet tested.

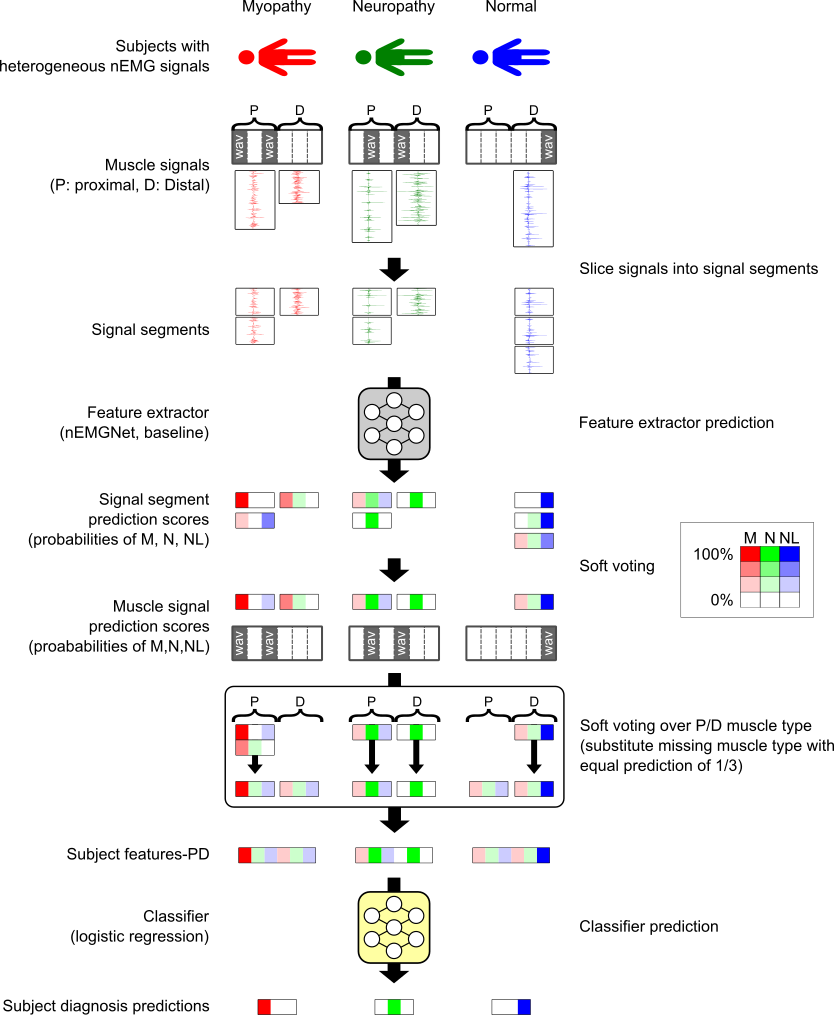
**Table 2** Convolutional blocks of nEMGNet. Spatial reduction block-1 reduces the spatial resolution to 50%, whereas spatial reduction block-2 reduces the spatial resolution to 25%. Residual blocks have residual connections which allows stable training in deep networks. In the title of each block, n and k inside the parentheses refers to the number of output channels and the size of the kernel.

|  |  |  |
| --- | --- | --- |
| **Spatial reduction block-1**  **(n,k)** | **Spatial reduction block-2**  **(n,k)** | **Residual block(n)** |
| Conv (k)-n,  Stride (1) | Conv (k)-n,  Stride (2) | Conv (5)-n,  Stride (1) |
| BatchNorm | BatchNorm | BatchNorm |
| ReLU | ReLU | ReLU |
| Max-pool (2),  Stride (2) | Max-pool (2),  Stride (2) | Conv (5)-n,  Stride (1) |
|  |  | BatchNorm |
|  |  | Sum (Residual connection) |
|  |  | ReLU |

**Table 3** Configuration of nEMGNet. Different numbers of residual blocks are experimented. The numbers inside the parentheses of the blocks refer to the hyperparameters defined in Table2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **nEMGNet-A** | **nEMGNet-B** | **nEMGNet-C** | **nEMGNet-D** |
| Block 1 | SR block-2  (64, 11) | SR block-2  (64, 11) | SR block-2  (64, 11) | SR block-2  (64, 11) |
| Block 2 | SR block-2  (64, 7) | SR block-2  (64, 7) | SR block-2  (64, 7) | SR block-2  (64, 7) |
| Block 3 | SR block-2  (64, 5) | SR block-2  (64, 5) | SR block-2  (64, 5) | SR block-2  (64, 5) |
| Block 4 |  | Residual block  (64) 2 | Residual block  (64) 4 | Residual block  (64) 6 |
| Block 5 | SR block-1  (128, 5) | SR block-1  (128, 5) | SR block-1  (128, 5) | SR block-1  (128, 5) |
| Block 6 |  | Residual block  (128) 2 | Residual block  (128) 4 | Residual block  (128) 6 |
| Block 7 | SR block-1  (256, 5) | SR block-1  (256, 5) | SR block-1  (256, 5) | SR block-1  (256, 5) |
| Block 8 |  | Residual block  (256) 2 | Residual block  (256) 4 | Residual block  (256) 6 |
| Block 9 | SR block-1  (512, 5) | SR block-1  (512, 5) | SR block-1  (512, 5) | SR block-1  (512, 5) |
| Block 10 |  | Residual block  (512) 2 | Residual block  (512) 4 | Residual block  (512) 6 |
| Block 11 | SR block-1  (1024, 5) | SR block-1  (1024, 5) | SR block-1  (1024, 5) | SR block-1  (1024, 5) |
| Block 12 |  | Residual block  (1024) 2 | Residual block  (1024) 4 | Residual block  (1024) 6 |
|  | FC-512 | FC-512 | FC-512 | FC-512 |
|  | FC-256 | FC-256 | FC-256 | FC-256 |
|  | FC-64 | FC-64 | FC-64 | FC-64 |
|  | FC-16 | FC-16 | FC-16 | FC-16 |
|  | FC-3 | FC-3 | FC-3 | FC-3 |
|  | Softmax | Softmax | Softmax | Softmax |

## Divide and Vote (DiVote) algorithm



**Figure 3** DiVote algorithm. Position and colors in the signal segment prediction scores, muscle signal prediction scores, subject features, and subject diagnosis predictions correspond to myopathy, neuropathy, and normal. The intensity in color indicates the magnitude of the value. Subject features-PD aggregates muscle signal prediction scores over proximal and distal muscle separately. Only subject features-PD is described and subject features-all is omitted.

In nEMG clinical diagnosis, each subject contains different number of signals from different muscles and the length of each muscle signal is heterogeneous. We propose the “Divide and Vote (DiVote)” algorithm (Figure 3) in order to predict the diagnosis of a subject given this heterogeneous data structure. At the inference stage, the heterogeneous muscle signals are divided into signal segments of homogeneous shape, which are the preprocessed signal segments in our study. Each signal segment is then converted by a feature extractor into three class signal segment prediction scores which contains probabilities for each diagnosis class label. The signal segment prediction scores from the same muscle signal are soft voted to create muscle signal prediction score. Muscle signal prediction score contains prediction probabilities for each of the three diagnosis labels. Since each subject possesses different numbers and types of muscle signals, each subject also possesses different numbers and types of muscle signal prediction scores. To integrate this heterogeneity, muscle signal prediction scores are soft voted to create subject features which are the final features assigned to each subject. Two types of subject features were experimented: soft voting over all muscle types which we refer to as “subject feature-all”, and soft voting over proximal and distal muscles respectively and concatenating the results which we refer to as “subject feature-PD”. When deriving subject feature-PD for a subject with no muscle signal prediction score in some muscle type, equal muscle signal prediction score of 1/3 was substituted (Figure 3, myopathy and normal subject). Therefore, subject feature-all is a three-dimensional vector whereas subject features-PD is a six-dimensional vector. Finally, a logistic regression classifier predicts the final three class diagnosis label of the subject based on subject features. Note that nEMGNet is used as a feature extractor in this study. Also, feature extractor is trained from signal segments and logistic regression classifier is trained from subject features after when the feature extractor is trained.

## Experimental Setup

Evaluation results from 5-fold cross validation were reported as final performance. Each fold was repeated 3 times with different nEMGNet weight initialization since deep learning performance is dependent on weight initialization [36]. When training nEMGNet, learning rate was set to 1e-3, batch size was set to 32, and epoch was set to 100 with early stopping on validation set. Adam [37] optimizer was used to train nEMGNet. Cross entropy was used as an objective function for model optimization, and class weights were applied in inverse proportion to the number of signal segments per diagnosis label to prevent bias in prediction. Hyperparameters were selected empirically. To compare the predictive performance of nEMGNet, baseline experiments were conducted with nEMG classification models from prior works. The baseline nEMG classification models were used as feature extractor from the DiVote algorithm. In the baseline experiment using a model from Nam et al. [24], the downsampled signal segments were plotted as images which were used to train Inception-v4 [38] image classification model. When using baseline models from Nodera et al. [23], mel spectrograms were generated from the same signal segments and trained Resnet50, Resnet152, VGG16, VGG19, and Inception-v3 [39] image classification models. All experiments were run on NVIDIA V100 GPUs.

## Feature Visualization

Feature visualization is an explainable deep learning technique which extracts the features a neural network has learned to accomplish a task [30, 31]. Feature visualization was applied to trained nEMGNet to visualize the features nEMGNet has learned. Three types of augmented signal segments were generated by optimizing random noise with respect to each output node of nEMGNet, which corresponds to each diagnosis label. Initial signal segments were generated from . Random jitter was applied by 12.5% of signal segment length in order to prevent over optimizing the signal segments. Learning rate was set to 1e-2 and gradient descent was applied for 1500 steps. Adam [37] optimizer was used to optimize the signal segments.

## Evaluation Metrics

Evaluation was performed at two stages. First, the signal segment classification was evaluated. Next, the subject diagnosis classification was evaluated. Both evaluations were three class classification tasks. Accuracy, precision, recall, F1-score, area under receiver operating characteristic curve (AUROC), Mathew’s correlation coefficient (MCC) [40] were computed. Performance metrics were computed from the following Eqs. (1)-(5).

()

()

()

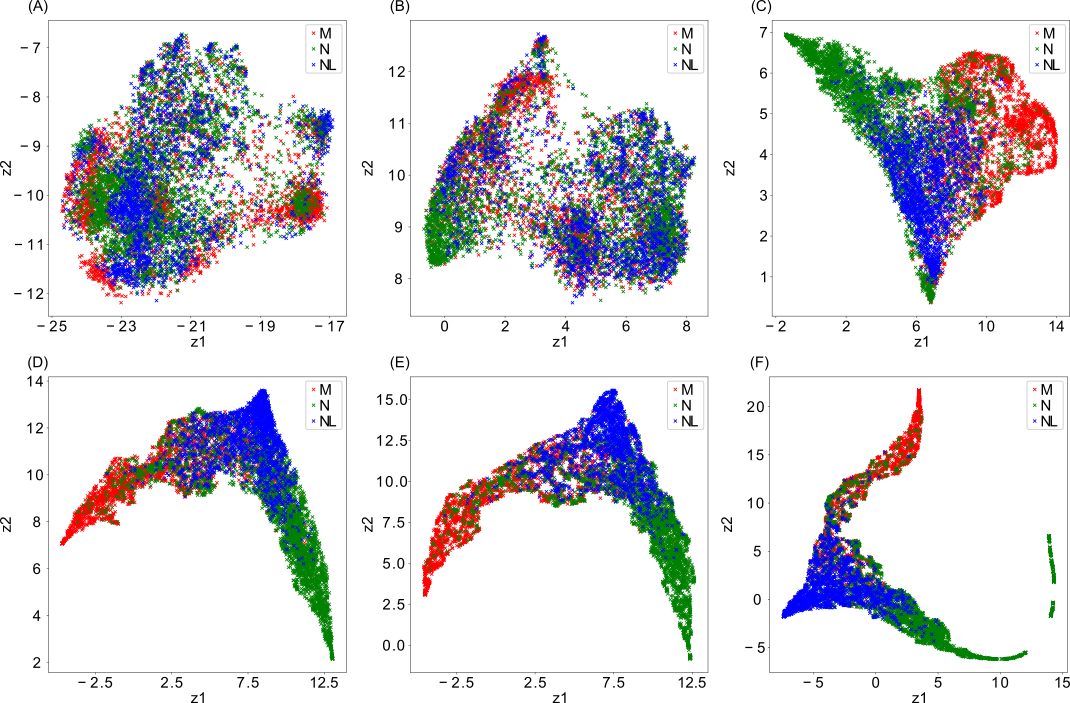
()

()

where TP, TN, FP, FN are the number of true positives, true negatives, false positives, and false negatives. C is the confusion matrix from n-class classification results with columns indicating true labels and rows indicating predicted labels. Note that accuracy and MCC are computed as three class classification metrics, whereas all other methods are weighted averages of metrics from one-versus-rest classification method [41], weighted according to the number of samples per class.

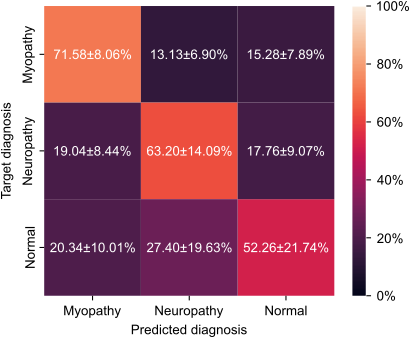
# Results

## EMG Segment Classification



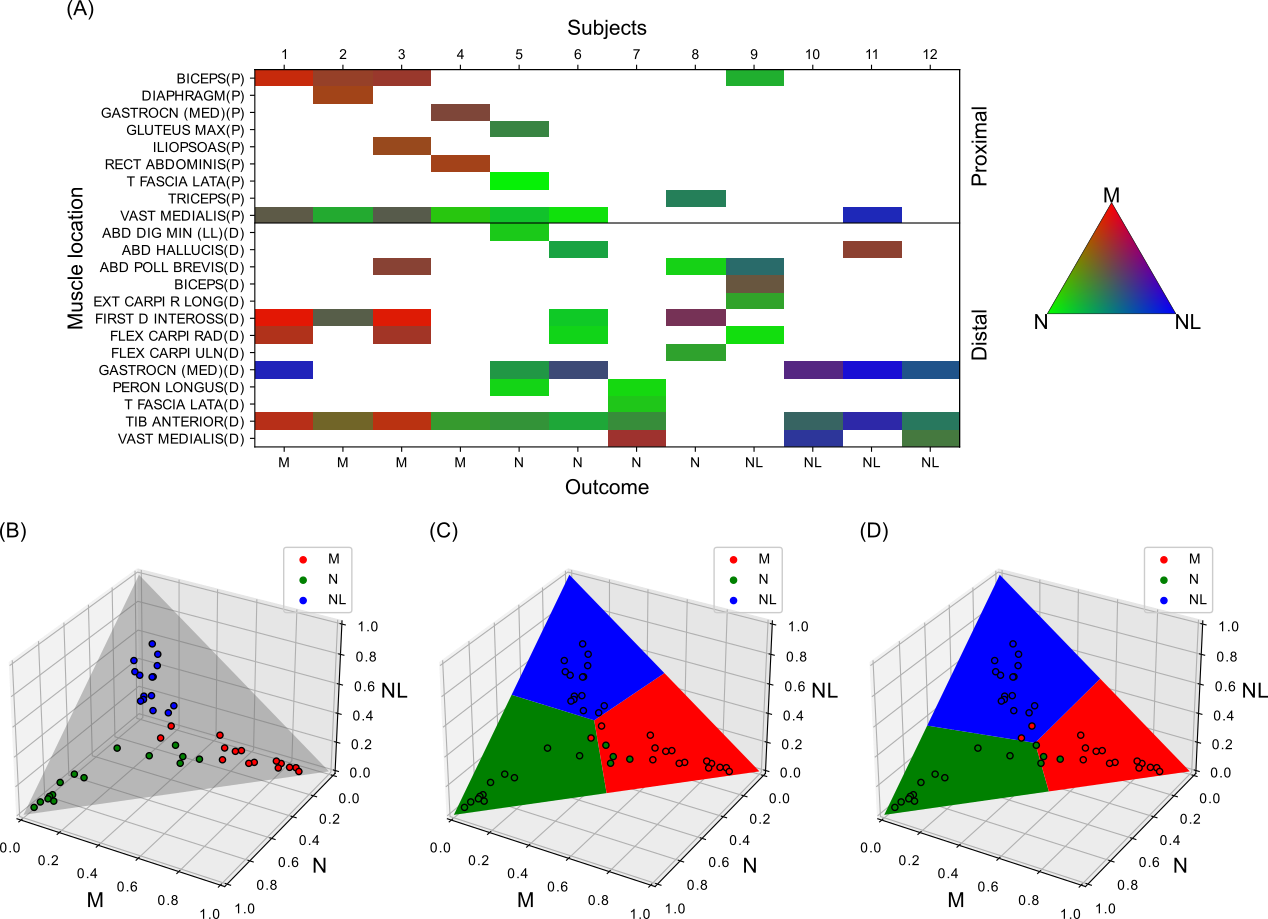
**Figure 4** Signal segments filtered through convolutional layers of nEMGNet. Filtered signals after passing through different convolutional blocks of the nEMGNet are plotted by reducing the dimensionality to two dimensions using Uniform Manifold Approximation and Projection. (a) Initial signal, (b) 2nd block, (c) 4th block, (d) 6th block, (e) 9th block, (f) 12th block. z1 and z2 correspond to reduced dimensions. M stands for myopathy, N stands for neuropathy, and NL stands for Normal signal segments.

To visualize the filtering effect of nEMGNet, signal segments filtered after different convolutional blocks were plotted. Uniform Manifold Approximation and Projection [42] was used to reduce the dimension of the filtered signal segments to two dimensions. Among different nEMGNet versions, best performing nEMGNet-B was used to plot Figure 4. Since the fully connected layers of nEMGNet are not convolutional operations and thus they do not filter the signal segments, tensors after fully connected layers were not used to plot Figure 4. As the signal segments are filtered through deeper layers of nEMGNet, signal segments from each label are farther apart in the reduced dimension. Additionally, the filtered signals of normal signal segments tend to be in between myopathic and neuropathic filtered signal segments.



**Figure 5** Confusion matrix of signal segment classification accuracy. The values indicate mean and standard deviation.

Confusion matrix of signal segment classification accuracy is plotted to show the classification performance of nEMGNet per diagnosis label (Figure 5). Results from best performing nEMGNet-B were used to plot the confusion matrix. Classification results over each fold and random repetitions were normalized. Total segment prediction accuracy over all classes was 62.354.60%. Myopathy and neuropathy signal segments are classified with prediction accuracy of 71.58% and 63.20% respectively, whereas normal signal segments are relatively more misclassified with prediction accuracy of 52.26%.



**Figure 6** Process of DiVote algorithm. (a) Heatmap of muscle signals prediction scores for subjects from the test set. Each square in the heatmap is computed by majority voting the signal segment prediction results within a muscle signal, predicted by nEMGNet. White blocks indicate no signals. The color is represented by assigning probability scores for myopathy, neuropathy, and normal to red, green, and blue. (b) Subject features-all plotted on a 3d plane. Each point in the plot corresponds to each subject feature from the train set, which are aggregated through soft voting the muscle signal prediction scores over all muscle types. The same scatterplot is described in (c) and (d). (c) Decision boundary of simple argmax function. (d) Decision boundary of classifier trained with subject features of (b). M stands for myopathy, N stands for neuropathy, and NL stands for Normal signals and subjects.

The process of DiVote algorithm after signal segment prediction is shown in Figure 6. Train and test set from the first fold was used to plot the figures. Figure 6A shows that each subject possesses different types and number of muscle signals. In addition, signals from myopathic or neuropathic subjects tend to be predicted as their respective labels, whereas signals from normal subjects tend to be ambiguous which can be identified from the colors of the heatmap (Figure 6A). Note that subject features-PD cannot be plotted in the same way as Figure 6B since subject features-PD are six-dimensional which cannot be plotted in three-dimensional space. Due to ambiguous prediction results of normal muscle signals, normal subject features are relatively biased towards myopathic and neuropathic features (Figure 6B). Elementary approach in deriving the final subject prediction scores is to use the subject features from Figure 6B as the final subject prediction scores which is equivalent to the argmax function (Figure 6C). To mitigate the bias of normal subject features, a classifier is trained using subject features from Figure 6B to move the decision boundary in feature space (Figure 6D).

## Subject Diagnosis

**Table 4** Subject diagnosis accuracy of nEMGNet and DiVote algorithm. Different types of nEMGNet and different types of subject features within the DiVote algorithm are compared. Subject features-all refers to subject features created from soft voting the muscle signal prediction scores over all muscle types. Subject features-PD refers to subject features created from soft voting the same scores over proximal and distal muscle types respectively and concatenating the results. Best performance is described in bold font. Results are expressed in meanstandard deviation.

|  |  |  |  |
| --- | --- | --- | --- |
| nEMGNet | Accuracy (%) | | |
| Simple averaging (No classifier) | Subject features-all | Subject features-PD |
| A | 67.1710.75 | 76.064.90 | 76.5710.23 |
| B | 73.647.27 | 81.924.83 | **83.695.28** |
| C | 69.957.77 | 81.266.35 | 81.876.80 |
| D | 75.356.93 | 81.266.35 | 80.815.31 |

Subject diagnosis accuracies of different nEMGNet versions and various prediction methods within DiVote algorithm were compared (Table 4). Simple averaging setting used average of muscle signal prediction scores as the final subject prediction score, which is equivalent to using subject features-all as the final subject prediction score. As there was no additional classifier, the decision boundary of simple averaging in feature space was equal to the standard argmax function (Figure 6C). Regardless of subject feature type, using an additional classifier always resulted in better performance. When training a classifier with subject features, subject features from soft voting over proximal and distal muscles resulted in better performance than soft voting over all muscles for most nEMGNet models. Among different versions of nEMGNets, nEMGNet-B which contains two residual blocks between SR blocks performed the best. nEMGNet-A which does not have any residual blocks performed worst. Increasing the number of residual blocks did not consistently increase model performance.

**Table 5** Subject classification result of nEMGNet and baseline feature extractors. nEMGNet was replaced with baseline feature extractors within the DiVote algorithm to compare performance. Metrics are mean values of each fold and random repetitions. Best evaluation metrics are described in bold font.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Reference | Feature extractor | Evaluation Metrics (%) | | | | | |
| Accuracy | F1 | Precision | Recall | AUROC | MCC |
| Current study | nEMGNet-A | 76.57 | 74.67 | 79.17 | 76.57 | 89.43 | 68.33 |
|  | nEMGNet-B | **83.69** | **83.59** | **87.96** | **83.69** | **91.45** | **77.70** |
|  | nEMGNet-C | 81.87 | 81.61 | 85.74 | 81.87 | 91.21 | 74.66 |
|  | nEMGNet-D | 80.81 | 80.56 | 86.65 | 80.81 | 90.53 | 74.16 |
| Nam et al. [24] | Inception-v4 | 57.47 | 51.38 | 56.53 | 57.47 | 78.39 | 42.03 |
| Nodera et al. [23] | ResNet50 | 73.84 | 72.59 | 81.94 | 73.84 | 81.27 | 64.76 |
|  | ResNet152 | 75.61 | 74.90 | 81.11 | 75.61 | 85.30 | 65.50 |
|  | VGG16 | 68.23 | 65.47 | 66.86 | 68.23 | 79.85 | 55.42 |
|  | VGG19 | 72.32 | 69.66 | 74.71 | 72.32 | 81.10 | 62.91 |
|  | Inception-v3 | 71.92 | 70.64 | 79.29 | 71.92 | 83.48 | 61.60 |

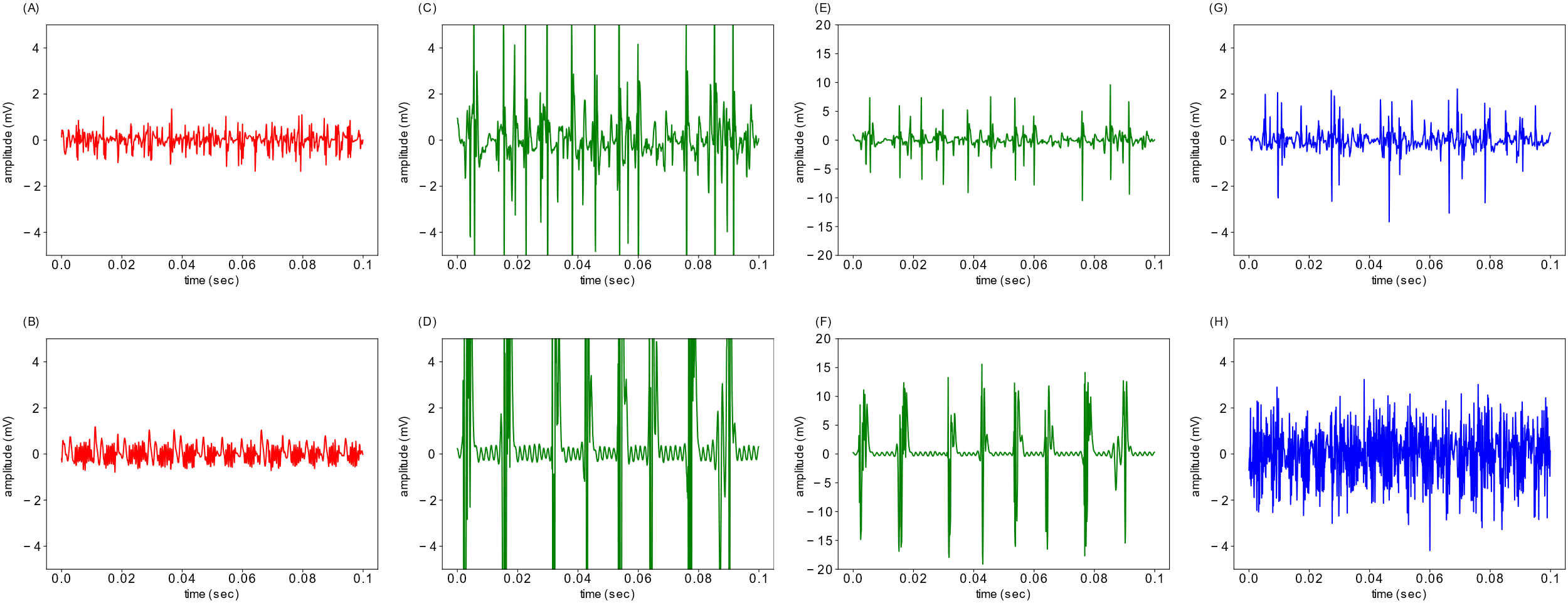
The performance of nEMGNet was compared to previous works by replacing nEMGNet with baseline feature extractors from previous works [23, 24] within the DiVote algorithm. Table 5 describes subject diagnosis prediction results of proposed nEMGNet and baseline feature extractors. Subject feature-PD was created after extracting muscle signal prediction scores using each feature extractors. Feature extractors from Nodera et al. [23] perform better than feature extractor from Nam et al. [24]. Among different models from Nodera et al. [23], ResNet152 performed the best in classification accuracy. It can be identified that nEMGNet outperforms all previous works in all metrics. Among different nEMGNet versions, nEMGNet-B performed the best in all metrics.

**Table 6** Elapsed times for subject diagnosis prediction and model parameter size. nEMGNet-B and baseline feature extractors are compared. The time spent in predicting subject diagnosis of the test set is measured. Elapsed times are the mean value of each fold and random repetitions. Elapsed time for classifier prediction is not described since the classifier is identical across all methods, and the time measured was less than 0.01 seconds. Total elapsed time refers to the time taken to preprocess signal segments, load the model parameters, extract features from feature extractor, and classify the subject diagnosis prediction scores. Best metrics are described in bold font.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Reference | Feature extractor | Elapsed time (sec) | | | | Number of parameters |
| Preprocessing | Load model parameters | Feature extractor prediction | Total |
| Proposed | nEMGNet-B | **0.01** | **0.43** | **0.54** | **0.99** | 33,238,686 |
| Nam et al. [24] | Inception-v4 | 68.92 | 1.17 | 5.32 | 75.42 | 41,210,744 |
| Nodera et al. [23] | ResNet50 | 19.08 | 0.61 | 1.21 | 20.90 | 23,567,352 |
|  | ResNet152 | 19.20 | 1.45 | 2.25 | 22.92 | 58,301,534 |
|  | VGG16 | 19.61 | 2.63 | 1.60 | 23.87 | 134,272,835 |
|  | VGG19 | 19.53 | 2.71 | 1.78 | 24.04 | 139,582,531 |
|  | Inception-v3 | 19.42 | 2.60 | 1.38 | 23.40 | **21,826,241** |

The time spent in the complete subject diagnosis prediction pipeline and the number of parameters for each model are measured for nEMGNet-B and baseline feature extractors (Table 6). Among different nEMGNet versions, best performing nEMGNet-B was compared. Each test set of five-fold cross validation was used in measurement, and the average number of signal segments, muscle signals, and subjects in test sets were 1614, 75.2, and 11.4 respectively. As there was no preprocessing for nEMGNet, the 0.01 second is measured from loading the signal segments. Also, preprocessing for baseline models from Nodera et al. [23] are identical and the slight variations in preprocessing time are empirical variances. The proposed nEMGNet spends significantly minimal time in preprocessing compared to other baseline methods since no preprocessing was applied. Note that the time spent in loading model parameters is related to the number of parameters but is not directly proportional. The time spent in feature extractor prediction was smallest of 0.54 seconds for the proposed nEMGNet. The total time required from preprocessing to subject diagnosis prediction was the fastest in the proposed nEMGNet, resulting in 0.99 seconds. The number of parameters includes the weights and biases of layers as well as the running mean and variance of batch normalization layers. In terms of model size, Inception-v3 model was the smallest. Compared to ResNet152 model which was the best performing model among baseline feature extractors, nEMGNet-B was smaller in model size.

## Feature Visualization of trained nEMGNet



**Figure 7** Learned features of nEMGNet. Top row are real signals and bottom row are generated signals through feature visualization. nEMGNet from the 1st fold was used to plot the figure. (a) Real myopathic signal. (b) Generated myopathic signal. (c) Real neuropathic signal. (d) Generated neuropathic signa€(e) Real neuropathic signal with 20mV y-axis limit. (f) Generated neuropathic signal with 20mV y-axis limit. (g) Real normal signal, (h) Generated normal signal. Note that (a), (b), (c), (d), (g), (h), is plotted with a 5mV y-axis limit for better comparison between signals of different labels. (e), (f) is plotted with a 20mV y-axis limit to show the overall shape of the neuropathic signals. (c)&(e) and (d)&(f) are identical signals.

Feature visualization results of nEMGNet are presented in Figure 7. The bottom row of Figure 7 are the signal segments generated through feature visualization, which the nEMGNet best perceives as myopathy, neuropathy, and normal. The maximum absolute amplitude for Figure 7A,B was 1.35mV and 1.27mV, Figure 7C-F was 10.50mV and 19.10mV, Figure 7G,H was 3.54mV and 4.19mV respectively. Amplitudes of the signal segments generated through feature visualization resembles that of each corresponding real signal segment.

# Discussion

In clinical trials of peripheral neuropathy diagnosis, the electromyographer records nEMG signals from various muscles and considers the signal characteristics and the muscle location in diagnosis. Significant information regarding the pathology of the patient is nested within the signals and the muscle types, which needs to be extracted in order to successfully classify the given subjects into diagnosis types. This study proposed nEMGNet, a one-dimensional CNN suitable for nEMG signal classification to extract features from raw signals. Four types of nEMGNet were experimented, and results showed that the proposed nEMGNet was capable of learning signal characteristics of each type of nEMG signals. Comparing of nEMGNet with previous works on nEMG diagnosis classification showed that nEMGNet outperforms all previous works. Also, the DiVote algorithm was introduced to mitigate data heterogeneity in subjects as well as leverage relevant information within proximal and distal muscle types. DiVote algorithm was able to handle data heterogeneity and bias in subject features, which improved subject classification performance compared to the case when there was no additional classifier. Finally, the learned features of nEMGNet were visualized which resembled each type of diagnosis signal characteristics by applying feature visualization to a trained nEMGNet.

## Attributes of nEMGNet

Throughout the signal segment classification results (Figure 5), muscle signal prediction scores (Figure 6A) and the soft voted subject features (Figure 6B), the values corresponding to normal subjects tend to have mixed prediction scores. Also, it can be identified from filtered signal segments (Figure 4F) that normal signals are in between myopathic and neuropathic signals in latent space, and that the latent features are not distinctly clustered per diagnosis labels. This can be explained with normal signal segments having similar latent features with myopathic and neuropathic signals. Since not all muscles from myopathic or neuropathic subjects show typical pathologic signal characteristics, some signals labeled as myopathy or neuropathy exhibit signal characteristics of normal muscles. As a result, signal segments labeled as myopathy or neuropathy contain signal characteristics of normal subjects, while signals from normal subjects do not contain any features that indicate myopathy or neuropathy.

This imbalance in feature distribution among different labels leads the nEMGNet to classify normal signals in a more ambiguous manner than other labels, as it was trained to classify normal features as myopathy and neuropathy as well. Additionally, the amplitude of the signal is the largest for neuropathic signals, medium for normal signals, and smallest for myopathy signals which might also induce intermediate prediction scores of normal signals. Furthermore, not only was nEMGNet trained using noisily labeled signal segments, but the signal segment test set also consisted of noisy labels which lowered the prediction accuracy of signal segments. This overlapping labels in terms of signal segment feature has caused signal segment prediction accuracy of normal signal segments to be 52.26% (Figure 5) which is the lowest among three diagnosis types.

Nevertheless, the loss value of nEMGNet had converged to a value close to zero during training, which indicates that the model was fully capable of learning features from the data. Also, filtering the signal segment through deeper layers of nEMGNet resulted in distant groups in latent space (Figure 4) which implies that nEMGNet has learned to distinguish signals of different diagnosis labels. Thus, the relatively ambiguous signal segment accuracy of normal labels was explained by signal segment data with ambiguous labels which consist the train set and the test set, and not by the capacity of nEMGNet to learn relevant features. Since the ultimate goal of this study is to classify the subjects, not the signal segments, segment prediction results are useful as long as they are divergent enough to generate distinct subject features.

There are several advantages of using 1D-CNN over image classification models in signal processing applications as the baseline experiment results indicate. 1D-CNN computes gradients during training and predicts at inference in one-dimensional direction whereas image classification models must operate in two dimensions. This results in significant reduction in train and inference time under similar number of model parameters (Table 6). During training and inference, gradients and hidden layer activation tensors are smaller which consumes smaller memory at runtime. Also, the number of parameters within a deep learning model is much smaller in 1D-CNN under identical number of layers and kernel size configurations (Table 6). Furthermore, it is harder to optimize neural architecture in image classification models since the search space is larger due to more hyperparameters in the model such as the width and height of the convolution kernel.

Additionally, nEMGNet may outperform image classification models because of its proximity to raw signals. Whenever a preprocessing operation is applied to a signal, the amount of information content is bound to be equal or smaller due to data processing inequality [32]. Preprocessing operations must be applied carefully to prevent loss of important features within the data. However, conventional signal features such as mel spectrogram or discrete wavelet transform needs human selection of preprocessing hyperparameters. In contrast, the proposed nEMGNet makes predictions based on raw signals. A convolution operation with a linear filter is equivalent to frequency filtering [43, 44], and training linear convolutional layers optimizes a frequency filter to extract features from a given signal. The nEMGNet can similarly be interpreted as a nonlinear filter based on 1D-CNN which extracts features from the raw signals directly. There are also other works on using 1D-CNN for signals from medicine [45, 46] which implicates the potential in leveraging 1D-CNN in signal processing applications.

## Significance of DiVote algorithm

In terms of the proposed DiVote algorithm, there are several significance. First, the DiVote algorithm is a robust pipeline for subjects with heterogeneous data structures. When subjects possess different types and number of signals which also vary in length, a pipeline is needed which converts the heterogeneous data into a homogeneous form. The division, feature extraction, and majority voting process successfully extracts and integrates meaningful features from varying data structures in a homogeneous manner.

Second, the DiVote algorithm can classify subjects with inconsistent features. As discussed previously, features and labels are not well labeled in terms of signal segments since all muscle signals from the same subject are labeled as the diagnosis of the subject. Inaccurately labeled signal segments not only limits the nEMGNet from learning acute features, but also limits the validity of test results since signal segments from the test set are also roughly labeled. This limitation can be mitigated with DiVote algorithm since subject features are generated by filtering muscle signals which contain different features through the nEMGNet, which is then classified using a classifier. Not only do most muscle signals from a subject follow the characteristics of the subject diagnosis label, but also the additional classifier shifts the decision boundary of subject features for classifying the subject diagnosis label (Figure 6B). This can be identified from Figure 6 which using an additional classifier better predicts subject diagnosis. Thus, even though some muscle signals of each subject contain irrelevant features for the diagnosis of the subject, the DiVote algorithm successfully classifies the given subject.

Furthermore, generating subject features from the DiVote algorithm can prevent undesired inductive bias in the deep learning model. Metadata such as the proximal or distal muscle information could be used as a feature vector to train the deep learning model [49, 50], but the model may learn irrelevant features from the metadata. For instance, if the muscle type information was used as a feature vector in nEMGNet, the model may predict proximal muscles as non-normal regardless of the signal segment characteristics, because the number of proximal muscle signals in normal subjects is small (Table 1). Applying metadata in the form of subject feature may allow the feature extractor to learn the desired features as well as leveraging the information from the metadata to improve task performance.

In medical applications the same subject often visits medical institutions multiple times, which results in longitudinal data. While homogeneous data structure is preferred in developing machine learning models, each measurement is likely to be heterogeneous. Such as the longitudinal data or the data from this study, measurements may be recorded from different time and space resulting in heterogeneous data structure. In medical applications where the data structure is heterogeneous within a given subject [47, 48], DiVote algorithm may leverage all data from the subject in training and integrate the heterogeneity at inference to predict the diagnosis of the subject. In addition, metadata within the heterogeneous data may be used to generate new types of subject features which may improve task performance.

## Learned features of nEMGNet

It is observable from Figure 7 that the signals which a trained nEMGNet is most likely to classify as each diagnosis label is similar in terms of magnitude. In addition, signal characteristics of each diagnosis label are similar to the feature visualization results. Characteristic of myopathic signals is small amplitude and higher frequency [10] which signal from Figure 7B similarly exhibit. Characteristic of neuropathic signals is large amplitude and lower frequency [10] which signal from Figure 7D, F is also similar. Learned features of normal signals are quite different from real normal signals since latent features for normal signals were not as concise as myopathic or neuropathic signals. Meticulous details of real signals are not simulated in feature visualization results which indicates that very fine details were not necessary to classify the normal signals from myopathic or neuropathic signals. Heuristic analysis by electromyographers suggested that the generated signals for myopathy and neuropathy exhibit typical pathologic characteristics. From this result it was verified that nEMGNet has learned the desired signal characteristics of each diagnosis type, addressing the causality of the deep learning model [51-53].

## Limitation and future works

There are several limitations of this study. First, there were only 57 subjects analyzed in this study which is insufficient to validate the evaluation results. Cross validation was performed due to small subject size. As a result, additional nEMG subjects must be collected in order to increase the sample size and validate the study using external validation. Second, in this study we focused only on distinguishing between normal, neuropathy and myopathy by analyzing nEMG waveforms. However, there are subtypes of myopathies which include inflammatory myositis, myotonic dystrophy, muscular dystrophies, and congenital myopathies. While these subtypes share common major myopathic motor unit action potential (MUAP) features, they also exhibit subtle differences in MUAP at the same time [10]. Likewise, there are subtypes for neuropathy which include compressive neuropathies, Guillain-Barre syndrome, radiculopathies and amyotrophic lateral sclerosis which share common neurogenic MUAP features but show subtle differences [10]. If a larger dataset including various subtype labels is available in the future, the proposed method may be able to determine a detailed diagnosis based on nEMG patterns.

# Conclusion

This study proposed nEMGNet, a 1D-CNN model to extract features from raw nEMG signals, and a DiVote algorithm to mediate data structure heterogeneity in predicting peripheral neuropathy diagnosis. Single channel needle electromyography signals of 57 subjects from Seoul National University Hospital were used to verify the proposed methods. Experimental results showed that the proposed nEMGNet and DiVote algorithm resulted in subject diagnosis accuracy of 83.69%, outperforming all other baseline nEMG deep learning feature extractors. Using an additional classifier in the DiVote algorithm increased subject diagnosis performance by shifting the decision boundary to mitigate bias in subject features and leveraging metadata information from proximal or distal muscle type increased performance. The typical signal segment features that the model predicts as each diagnosis type were identified using feature visualization, and it was verified that the model was making predictions based on valid features. By improving feature extraction performance, introducing a method to mediate the heterogeneous data structure of subjects, and suggesting the causality of deep learning model, this study is making a step closer to applying deep learning in nEMG clinical diagnosis. With future work when nEMG data from enough subjects are acquired to further verify the performance stability, the proposed method may be utilized in real clinical diagnosis of peripheral neuropathy.

# Acknowledgements

This work was supported by the Technology Innovation Program (20016225, Development and dissemination of Standard Reference Data) funded By the Ministry of Trade, Industry & Energy (MOTIE, Korea)

# CRediT authorship contribution statement

Jaesung Yoo: Conceptualization, Methodology, Software, Formal analysis, Writing – Original Draft, Writing – Review & Editing, Visualization

Ilhan Yoo: Conceptualization, Resources, Data Curation, Writing – Original Draft, Writing – Review & Editing, Visualization

Ina Youn: Data Curation

Sungmin Kim: Resources, Data Curation

Ri Yu: Writing – Review & Editing

Kwangsoo Kim: Resources, Writing – Review & Editing

Keewon Kim: Conceptualization, Resources, Writing – Review & Editing, Supervision

Seung-Bo Lee: Conceptualization, Methodology, Writing – Review & Editing, Supervision

# Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

# References

1. Said, G., *Diabetic neuropathy—a review.* Nature clinical practice Neurology, 2007. **3**(6): p. 331-340.

2. Bae, E.H., M.K. Greenwald, and A.G. Schwartz, *Chemotherapy-Induced Peripheral Neuropathy: Mechanisms and Therapeutic Avenues.* Neurotherapeutics, 2021: p. 1-13.

3. Smith, A.G. and J.R. Singleton, *Idiopathic neuropathy, prediabetes and the metabolic syndrome.* Journal of the neurological sciences, 2006. **242**(1-2): p. 9-14.

4. Smith, A.G., K. Rose, and J.R. Singleton, *Idiopathic neuropathy patients are at high risk for metabolic syndrome.* Journal of the neurological sciences, 2008. **273**(1-2): p. 25-28.

5. Martyn, C. and R. Hughes, *Epidemiology of peripheral neuropathy.* Journal of neurology, neurosurgery, and psychiatry, 1997. **62**(4): p. 310.

6. Daube, J.R. and D.I. Rubin, *Needle electromyography.* Muscle & Nerve: Official Journal of the American Association of Electrodiagnostic Medicine, 2009. **39**(2): p. 244-270.

7. Mills, K.R., *The basics of electromyography.* Journal of Neurology, Neurosurgery & Psychiatry, 2005. **76**(suppl 2): p. ii32-ii35.

8. Rubin, D.I., *Needle electromyography: Basic concepts.* Handbook of clinical neurology, 2019. **160**: p. 243-256.

9. Whittaker, R.G., *The fundamentals of electromyography.* Practical neurology, 2012. **12**(3): p. 187-194.

10. Kimura, J., *Electrodiagnosis in diseases of nerve and muscle: principles and practice.* 2013.

11. Oh, S.J., *Clinical electromyography: nerve conduction studies*. 2003: Lippincott Williams & Wilkins.

12. Torpy, J.M., J.L. Kincaid, and R.M. Glass, *Peripheral Neuropathy.* JAMA, 2008. **299**(9): p. 1096-1096.

13. Leblhuber, F., et al., *Diagnostic value of different electrophysiologic tests in cervical disk prolapse.* Neurology, 1988. **38**(12): p. 1879-1879.

14. Kendall, R. and R.A. Werner, *Interrater reliability of the needle examination in lumbosacral radiculopathy.* Muscle & Nerve: Official Journal of the American Association of Electrodiagnostic Medicine, 2006. **34**(2): p. 238-241.

15. LeCun, Y., Y. Bengio, and G. Hinton, *Deep learning.* nature, 2015. **521**(7553): p. 436-444.

16. Abd-Elnaby, M., M. Alfonse, and M. Roushdy, *Classification of Breast Cancer Using Microarray Gene Expression Data: A Survey.* Journal of Biomedical Informatics, 2021: p. 103764.

17. Faris, H., et al., *Medical speciality classification system based on binary particle swarms and ensemble of one vs. rest support vector machines.* Journal of Biomedical Informatics, 2020. **109**: p. 103525.

18. Hammami, L., et al., *Automated classification of cancer morphology from Italian pathology reports using Natural Language Processing techniques: A rule-based approach.* Journal of Biomedical Informatics, 2021. **116**: p. 103712.

19. Paragliola, G. and A. Coronato, *An hybrid ECG-based deep network for the early identification of high-risk to major cardiovascular events for hypertension patients.* Journal of Biomedical Informatics, 2021. **113**: p. 103648.

20. Dosovitskiy, A., et al., *An image is worth 16x16 words: Transformers for image recognition at scale.* arXiv preprint arXiv:2010.11929, 2020.

21. Amodei, D., et al. *Deep speech 2: End-to-end speech recognition in english and mandarin*. in *International conference on machine learning*. 2016. PMLR.

22. Brown, T.B., et al., *Language models are few-shot learners.* arXiv preprint arXiv:2005.14165, 2020.

23. Nodera, H., et al., *Deep learning for waveform identification of resting needle electromyography signals.* Clinical Neurophysiology, 2019. **130**(5): p. 617-623.

24. Nam, S., et al., *Development of artificial intelligence to support needle electromyography diagnostic analysis.* Healthcare informatics research, 2019. **25**(2): p. 131-138.

25. 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. IEEE.

26. Amann, J., et al., *Explainability for artificial intelligence in healthcare: a multidisciplinary perspective.* BMC Medical Informatics and Decision Making, 2020. **20**(1): p. 1-9.

27. Lauritsen, S.M., et al., *Explainable artificial intelligence model to predict acute critical illness from electronic health records.* Nature communications, 2020. **11**(1): p. 1-11.

28. Lapuschkin, S., et al., *Unmasking Clever Hans predictors and assessing what machines really learn.* Nature communications, 2019. **10**(1): p. 1-8.

29. Anders, C.J., et al., *Finding and removing clever hans: Using explanation methods to debug and improve deep models.* Information Fusion, 2022. **77**: p. 261-295.

30. Olah, C., A. Mordvintsev, and L. Schubert, *Feature visualization.* Distill, 2017. **2**(11): p. e7.

31. Erhan, D., et al., *Visualizing higher-layer features of a deep network.* University of Montreal, 2009. **1341**(3): p. 1.

32. Cover, T.M., *Elements of information theory*. 1999: John Wiley & Sons.

33. Simonyan, K. and A. Zisserman, *Very deep convolutional networks for large-scale image recognition.* arXiv preprint arXiv:1409.1556, 2014.

34. He, K., et al. *Deep residual learning for image recognition*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

35. Zaeemzadeh, A., N. Rahnavard, and M. Shah, *Norm-preservation: Why residual networks can become extremely deep?* IEEE transactions on pattern analysis and machine intelligence, 2020.

36. Yam, J.Y. and T.W. Chow, *A weight initialization method for improving training speed in feedforward neural network.* Neurocomputing, 2000. **30**(1-4): p. 219-232.

37. Kingma, D.P. and J. Ba, *Adam: A method for stochastic optimization.* arXiv preprint arXiv:1412.6980, 2014.

38. Szegedy, C., et al. *Inception-v4, inception-resnet and the impact of residual connections on learning*. in *Thirty-first AAAI conference on artificial intelligence*. 2017.

39. Szegedy, C., et al. *Rethinking the inception architecture for computer vision*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

40. Gorodkin, J., *Comparing two K-category assignments by a K-category correlation coefficient.* Computational biology and chemistry, 2004. **28**(5-6): p. 367-374.

41. Bishop, C.M., *Pattern recognition.* Machine learning, 2006. **128**(9).

42. McInnes, L., J. Healy, and J. Melville, *Umap: Uniform manifold approximation and projection for dimension reduction.* arXiv preprint arXiv:1802.03426, 2018.

43. Ifeachor, E.C. and B.W. Jervis, *Digital signal processing: a practical approach*. 2002: Pearson Education.

44. Openheim, A.V., A.S. Willsky, and S. Nawab, *Signals and systems.* Prentice Hall, 1983.

45. Acharya, U.R., et al., *Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network.* Information sciences, 2017. **405**: p. 81-90.

46. Galloway, C.D., et al., *Development and validation of a deep-learning model to screen for hyperkalemia from the electrocardiogram.* JAMA cardiology, 2019. **4**(5): p. 428-436.

47. Lee, G., et al., *MildInt: deep learning-based multimodal longitudinal data integration framework.* Frontiers in genetics, 2019. **10**: p. 617.

48. Gao, L., et al. *Brain disease diagnosis using deep learning features from longitudinal MR images*. in *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*. 2018. Springer.

49. Xiong, C., et al. *Conditional convolutional neural network for modality-aware face recognition*. in *Proceedings of the IEEE International Conference on Computer Vision*. 2015.

50. Yoo, I.-C., et al., *Speaker Anonymization for Personal Information Protection Using Voice Conversion Techniques.* IEEE Access, 2020. **8**: p. 198637-198645.

51. Arrieta, A.B., et al., *Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI.* Information Fusion, 2020. **58**: p. 82-115.

52. Tjoa, E. and C. Guan, *A survey on explainable artificial intelligence (xai): Toward medical xai.* IEEE Transactions on Neural Networks and Learning Systems, 2020.

53. Holzinger, A., et al., *Causability and explainability of artificial intelligence in medicine.* Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2019. **9**(4): p. e1312.