Classification of sEMG signals and Motion data using learning algorithms with application to post-stroke impairment

A Thesis Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Sadhu Sri Ravali

August 2019

This thesis is dedicated to my dad, I couldn't have done this without you.

Acknowledgments

I would like to thank first and foremost my advisor, Dr. Donatello Materassi, for guiding me through this project, introducing me to new topics in data science and for being patient with my unorganized thoughts about the thesis. I am also grateful to him for introducing me to my co-advisor Dr. Eric Wade. I am thankful to Dr. wade for accepting me as his student irrespective of being from a different department and letting me to be a part of his amazing research group at the Behavior, Robotics and Sensing group. I'm grateful for the great work environment that I had throughout my thesis as well as the people I got to know there. I am thankful to him for also guiding me and helping me in defining my thesis. Furthermore I would like to thank Dr. Hairong Qi for guiding me through the world of artificial intelligence and supporting me through my thesis. I must also thank my family and friends for supporting me to complete this thesis, hence moving a step closer to my goal in life. Thank You everyone!

Abstract

Stroke is the 3^{rd} leading cause of deaths in USA with an equally high number of survivors. Post-stroke rehabilitation is an important part of recovery after stroke as it helps to relearn skills lost due to the effect of stroke on a part of your brain. Stroke rehabilitation can help the subject regain independence and improve their quality of life. During their rehabilitation process, a subject is expected to perform a certain number of exercises and constantly try to use their affected limb. But due to social constraints and psychological stress, the subjects tend to perform the respective activities only in a closed environment and hence only in the presence of a physicist i.e. during their rehabilitation in an hospital, thereby dampening their process of rehabilitation.

This thesis aims at laying the foundation to develop a wearable device to track the recovery of a stroke patient while their in-hospital and at home upper limb rehabilitation processes depending on the muscular activity retrieved by a non-invasive sensors placed on the skin of the upper limb of the patient. To achieve this goal multiple steps are involved, but this thesis concentrates on the step which involves identifying the exercise performed by the subject using their bio & kinematic signals. To develop the ideal network for this goal, the thesis initially compares various classic learning techniques used to classify hand gestures using surface-Electromyography(sEMG). From the results obtained from these networks, a CNN is designed specific to the activities and input signals used in this thesis. The CNN is able to classify these activities with an accuracy of around 91%.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Muscle action potential is the electric potential that a muscle tissue conducts while performing an activity. When the muscles are at rest or during contraction these bio-electric potentials are generated which are recorded as an electromygram [14]. Researchers have been working with these signals since 1960 to improve the usage of these signals in fields like in wearable rehabilitation robots or in Human Machine interface or in exo-skeleton structures to guide patients during their rehabilitation process, in prosthetic devices and others. This chapter will explain the basic elements required for this along with laying out the motivation and objective of the thesis.

1.1 Background

Myoelectric signals are formed by physiological variations in the state of muscle fiber membranes during voluntary, involuntary or stimulated contractions [6]. They are complicated signals whose origin is from the nervous system and hence also controlled by the brain. Electrochemical transmission between nerves starting from the brain, produces action potential which propagates through nerve fibers and finally stimulates the skeletal muscle [6], which create muscle contraction and hence cause the limb movement. Each action potential stimulates an individual nerve, whereas to perform a limb movement many skeletal muscle fibers would be required, thus the electrical potential recorded as EMG from muscle movement is actually superposition of action potentials acting on skeletal fiber muscles [3].

These EMG signals differ from person to person as they majorly depend on the anatomical and physiological properties of the muscles of the respective human being. Currently the famous method of extracting these signals is a non-invasive way where surface electrodes would be used thereby being called as surface electromyography(sEMG).

Methods to collect, detection, pre-processing, interpretation and analysis of these sEMG signals has been the major part of the research done on them so far. With respect to the application of these signals, in the past decade these signals have been used as an indicator of force only. For example in EMG based exo - skeletons, the signals recorded from selected muscles are used as a trigger for robotic assistance when the value at a certain time increases above a threshold. But recently, pattern recognition based limb movement classification of these signals has gained considerable attention. The classifiers are used to identify the specific muscle that is being activated, the angular movement of the skeleton at that particular limb, the reason behind its activation and other purposes. The popular classifiers leveraged for gesture recognition with high classification accuracy are Support Vector Machine(SVM), Random Forest(RF), K-Nearest Neighbors(KNN) and Linear Discriminant Analysis(LDA). Neural Networks have also been used to classify these signals[4], but their results are a trade off between dataset size and the number of movements being classified. Very few researchers have implemented neural networks that would derive their own features instead of using engineered features, as a large quantity of data would be required to train the network.

In this thesis most of the focus would be on pre-processing and classification of these signals. The thesis's aims at laying a foundation for the development of an algorithm for a wearable device that can detect movements performed by the upper limb a post stroke patient during their rehabilitation process and track their rehabilitation process, depending majorly on their real time sEMG signals. The first step required to achieve this goal, as focused in this thesis is to build a classifier that can detect the exercises that an post stroke would be required to perform during their rehabilitation process.

1.2 Aim

As mentioned before pattern recognition algorithms have been applied on these signals to classify them but very less research has been done with NNs. The biggest concern while trying to use an NN as a classifier is the requirement of a huge dataset. But from early 2014 it has been feasible, with the development of the publicly available NinaPro database, a solid benchmark protocol for sEMG signals has been set, and also access to large datasets of sEMG signals has been easy. With this, a few papers have been published where NNs have been implemented to classify these sEMG signals without deriving the features beforehand but the highest accuracy accounted for these algorithms is 87.8% [1] only, while the benchmark accuracy attained by a pattern recognition algorithm is around 94% [14]. Along side, there hasn't been any prominent research on the implementation of recurrent neural networks(RNNs) to classify these signals though this algorithm is believed to perform best with time series data.

Hence the aim of this thesis is to classify the sEMG signals from the dataset 5 of the NinaPro repository using algorithms from both machine learning and deep learning. The results have been compared and analyze to explain the connection between the network used and the nature of the signal. 4 major networks have been used to classify the signals in this thesis, LDA, Multi-Layer Perceptron(MLP), CNN and RNN. The search for their respective hyper-parameters have also been explained.

1.3 Outline

This thesis is organized into five chapters. Beginning with a literature survey about the available algorithms and networks and their respective limitations or benefits. This is followed by a detailed description of the database, Nina Pro and the dataset of signals used from it. The following chapter explains the features extracted from the signals and the composition of the input matrix with respect to both the NN and pattern recognition algorithms. This is followed by a chapter where the models used are introduced. In this chapter along with the theory behind each model, the structure and the process for selecting their hyper parameters has been explained. This next chapter summaries the experiments conducted and results obtained, with the models and data defined. Lastly conclusions were drawn from the results obtained and discussed thoroughly.

Chapter 2

Theory

This chapter summarizes the performance of algorithms that have been used to classify upper limb movements of various subjects, along with recent developments in the field of post-stroke rehabilitation. After which the NinaPro repository is introduced and the database used is explained thoroughly.

2.1 Literature Survey

A example of an EMG signal is shown in figure 2.1 To collect a EMG signal an electrode is placed over a muscle and another reference electrode is placed near to a bone to set the baseline for the disturbances at that point of time. The signals attained from these

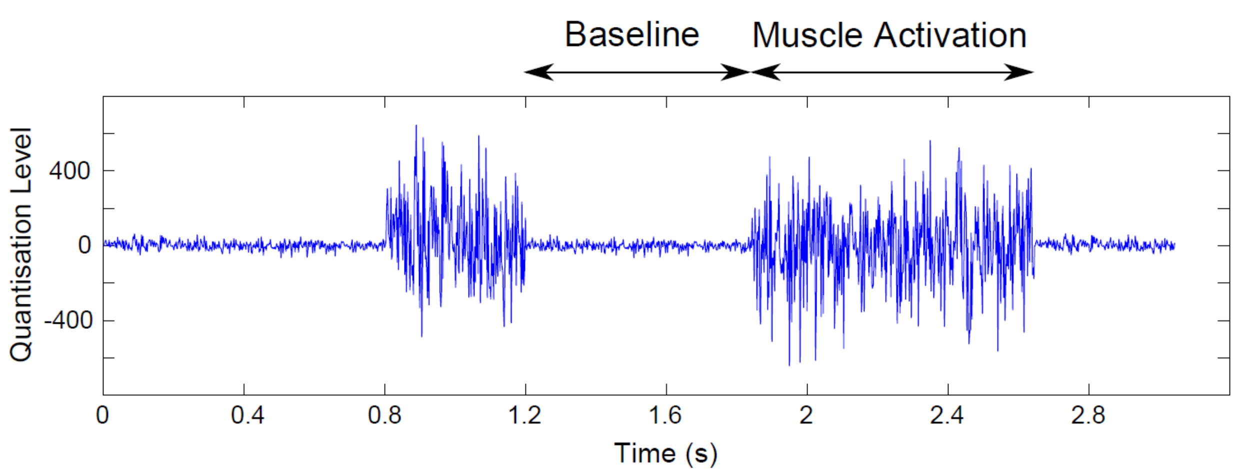


Figure 2.1: Sample sEMG signal

signals are then fed to a differential amplifier circuit which produces a clean EMG signal for that muscle. This whole setup constitutes a channel of the EMG signal. Usually there are multiple channels in an EMG collection setup as per limb movement there will be more then one muscle movement. Using these basics, EMG signals can be measured using invasive methods such as Peripheral Nerve Interfaces, Cortical Interfaces and Targeted Muscle Reinnervation(TMR) or a non- invasive methods called surface - EMG(sEMG). As the names suggests, in an invasive method, a surgery is performed and the electrodes are placed on the muscles directly giving clearer signals, to further understand, let's consider the TMR procedure which is a relatively famous procedure in the field of myoelctric control as it has shown promising results with respect to shoulder amputees. It is a surgical process where the spare muscle of the amputee is re-innervated with the residual nerves of their amputated limb. After this procedure the movement by the respective muscle can be recorded by the surface electrodes with lesser noise as the movement becomes more absolute. In another research, with the development of bio friendly electrodes, these electrodes have been placed on the nerves of the patient and wires are tapped out of the human body. The signals are then collected from these wires. Though this method of collecting signals gives stronger and cleaner signals, they are very stressful methods. Even when this survey [5] questioned the preference and compatibility level of upper-limb amputees with respect to the following four different techniques for detecting EMG signals: sEMG, TMR, Peripheral Nerve interfaces and Cortocal interfaces, participants expressed most interest in sEMG technique, thereby highlighting the importance of non-invasive techniques. In this method, electrodes are placed on the skin of the respective human and signals are collected from them. But the tricky part of this method is the placement of the signals, the electrodes have to reside over muscles whose effect can be felt through the skin. Irrespective of this, this method has be proven to be reliable and effective signals can be collected with the development of new technologies.

Movement intention detection using human bio-signals has gained considerable recognition thanks to its easy data collection nature with surface-EMG electrodes. Hence the classification just based on the non-invasive method has been a high priority for researches. Leading to many publications on smart algorithms that were developed to classify them. Han et al.[7] proposed fuzzy pattern classification and fuzzy min-max neural networks(FMMNN) technique for the classification of these signals which can then be used in rehabilitation robotic care system to assist disabled and elderly people. Though this algorithm was able to classify EMG signals of primitive arm motions with an average success rate of 83% for fuzzy pattern classifier and 90% for FMMNN with minimum dependency on the user, they still had to use engineered features. On the other hand, in the following paper [2] summarized the machine learning algorithms i.e. SVM and random Forests and KNN, used to classify 50 movements using just sEMG signals[2]. Though the accuracy obtained by these methods were less, the research sets a foundation to classify the sEMG signals for a lot of movements unlike the previous mentioned research where only 8 movements were being detected by the network.

The benchmark classification accuracy attained so far is 94% by [14]. They have evaluated 14 time domain features using scatter plots and extracted the best possible combination of features to get a high accuracy. They collected the data for 7 limb motions, where the person's data was collected in 4 sessions which were performed on separate days, with 6 trails per session i.e since data was being collected from just a single subject, the person alone had to provide a lot of data. The subject had to perform the exercise for 4 times within each trial with a duration of 3 seconds. Through their analysis based on scatter plot, they concluded that limb motions are significantly separable for willison amplitude and waveform length features. They have used a linear discriminant classifier to classify the signals while experiment with the feature reduction technique between ULDA and PCA. Based on the classification performance they concluded that a relatively simple pattern classification system using a dimension reduction technique can achieve a good classification accuracy, especially if the feature vector includes AR coefficients as features. They also concluded that the following feature vector shows the best classification accuracy: (4-AR Coefficients/WAMP) and (MAV1/WL/AAC/ZC/WAMP).

As the researches evolved, the necessity to automatize the project increased, i.e. the algorithms would require to derive a feature vector by themselves then being fed with a human formulated vectors. This paved way to the implementation of the popular deep learning networks on these signals. As their popularity is in the image classification field, hence the first documented application of CNN on sEMG was only in 2016. [15] is a single conference article which proposes a novel method of movement intention estimation based on the deep learning feature using sEMG signals from 27 subjects for 6 different hand movements. This paper also intended to tackle the problem of dependency on subjects. That is, to check if an algorithm cloud train on subject 1 while classify the signals of subject 2, hence 2 strategies were experimented in this paper. In the 1st strategy the proposed CNN structure in the paper was trained without the signals of the test subject while in the 2nd strategy, the CNN was retrained using a few of the labeled data of the test subject. They concluded that the user-adaptive decoding(2nd strategy) was able to classify with a higher accuracy of 90%. This was higher then their reference SVM model[8] which had an accuracy around 73%.

The delay in implementation of deep learning networks on sEMG signals is due to the requirement of huge data-sets of these signals from various backgrounds of people and also specificity in the collection procedure of the data. In 2014, NinaPro Project released the biggest publicly available benchmark repository of sEMG signal Thus paving path for future researches using big neural network structures. The following section will explain clearly the details of the repository as the dataset used for this thesis is from here.

2.2 Data

The Ninapro project is an ongoing work that aims to aid research on advanced hand myoelectric prosthetics or other feilds that can use sEMG signals, with publicly available datasets. From their webpage the data and related information can be downloaded. The project contains a repository of 7 databases currently which are obtained by jointly recording multi-modal data, including e.g. surface electromyography (sEMG) signals, hand kinematics, hand dyanamics while the subjects perform a predefined set of up to 53 movements.Each of these datasets vary in the sEMG signals acqusition setup or the type of patients and the number of patients from whom data has been collected. For this thesis, signals from NinaPro's dataset 5(DB5) has been used. The signals in this set are collected using two Myo Armbands from 10 intact subjects.

2.2.1 Acquisition Setup

In DB5 3 sensors are used in total, i.e. a 22-sensor Cyberglove II dataglove[13] and two Myo armbands. But for this thesis, only data from Myoarmbands are considered. It is a low cost wireless armband containing 8 single differential sEMG sensors and a 9 axis Inertial Measurement Unit (IMU) [11] developed by Thalmic Labs in 2013. The Thalmic Myo armband is only $199 which is almost 100 times less then the other setups used in the various other databases of NinaPro. Moreover the Myo configuration provides an extended uniform muscle mapping and it allows to analyze the data recorded from the multiple armbands together or separately. The Myo electrodes do not require the arm to be shaved and the armband tightens very firmly to the arm of the subject. All these characteristics of this database are the reasons to chose it for this thesis and also it is in accordance to the aim of the thesis, i.e. it uses a economical wearable device that can be used in clinical/bio-medical rehabilitation.

Two Myo Armbands were used to collect the data hence the dataset is called the Double Myo dataset (DB5). The subject wears two Myo armbands one next to the other as shown in the figure 2.2. The upper Myo armband is placed closer to the elbow with the first electrode on the radio humeral joint, following the Ninapro electrode configuration [2]. The lower Myo armband is placed just below the first, closer to the hand, tilted of 22.5°to fill the gaps left by the electrodes of the other Myo. Each Myo armband samples 8 sEMG sensors at a 200 Hz frequency with a resolution of 8 bit signed and streams the data through a Bluetooth low energy connection to the computer running the Myo Connection application. It also streams raw accelerometer data with 3 axes from the first Myo at a 50 Hz frequency

Figure 2.2: Acquisition setup for DB5 using Double Myo

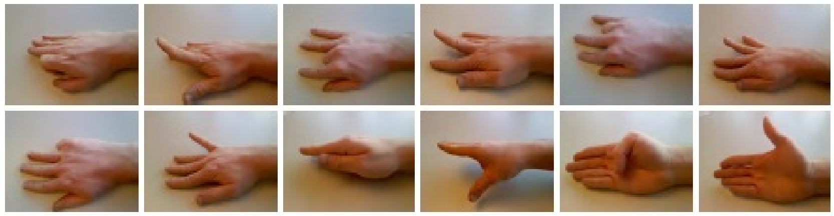
from the IMUs present in the Myo Armband. Though an myo armband has three IMUs data, i.e accelerometer, gyoroscope and megnotometer, DB5 only provides the accelerometer data. This data is also used in the thesis as Krasoulis et al. [9] concluded in his research that simultaneous use of EMG and IMU recordings increases the accuracy while classifying activities performed. The database also includes hand kinematics data, recorded using a 22-sensor CyberGlove II dataglove from CyberGlove Systems LLC[13]. As mentioned before, the data from this glove has not been considered as it violates the intention of this thesis to classify based on minimum number of wearable devices. In this dataset the average environmental temperature was approximately 22 Celsius degrees.

2.2.2 Signals and Subjects

The groups of subjects considered for this database are balanced and matched according to several parameters that may affect sEMG amplitude and classification accuracy[16]. Data was collected from the subjects while they repeat the movements represented by movies that were shown on the screen of a laptop. During the acquisition, the subjects were asked to repeat the movements with the right hand, each movement repetition lasted 5 seconds and was followed by 3 seconds of rest. The protocol includes 6 repetitions of 52 different movements. The movements were selected from the hand taxonomy as well as from hand robotics literature. The movements collected can be divided into three types of exercises:

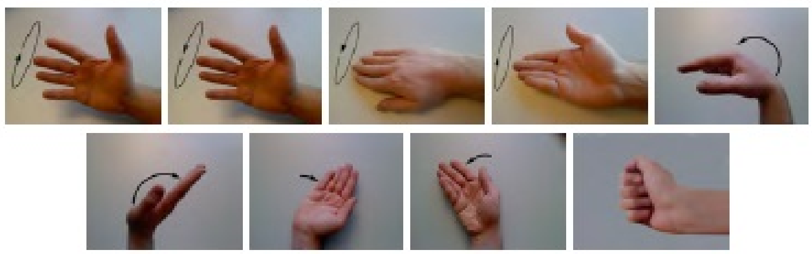
- Basic movements by fingers(12)
- Isometric, isotonic hand configuration and basic wrist movements(17)
- Grasping and functional movements(23)

Hence totalling to a total of 52 activities as shown in figure 2.3. From these activities, for the thesis only 10 activities, which are used as exercises in the rehabilitation process have been chosen. The following figure 2.4 consists of them.

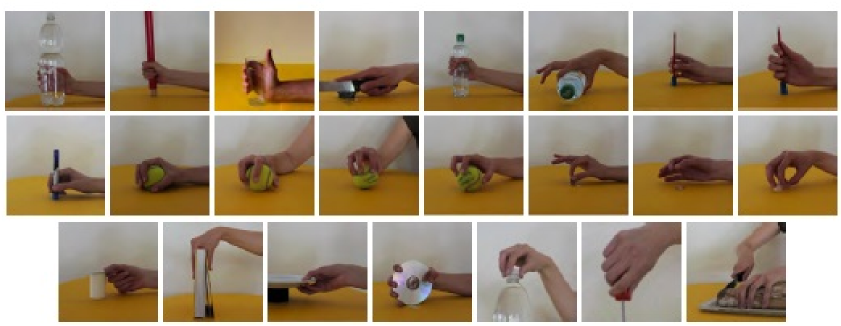


2a) Basic movements by fingers

2b.1) Isometric and isotonic hand configuration



2b.2) Basic Wrist movement



2c) Grasping and functional movements

Figure 2.3: 52 Movements by subjects

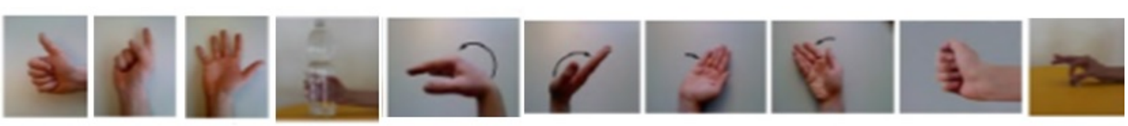


Figure 2.4: Activities classified using various algorithms in this thesis

2.3 Data Processing

Signal processing was performed before data analysis and classification. Usually power line interference can affect signal recording due to which filtering has to be done first. But the Thalmic Myo already has a notch filter at 50 Hz so no filtering was required for this sensor. The data from the two Myo armbands were recorded separately and merged afterwards on a timestamp basis. The movements performed by the subjects may not perfectly mirror the ones shown on screen, due to human reaction times. Movement detection algorithms were used to correct imperfect labeling[10].

This processed data is shared in the NinaPro repository. Each dataset contains files for each subject and exercise in Matlab format with filtered and synchronized data. For this thesis, this data was converted into an excel form on which analysis is performed.

Chapter 3

Data Analysis

Data analysis is a process where the data obtained is cleaned, transformed and modeled so that useful information and conclusions can be drawn from them. As Each classifier has its respective constraint, the analysis of the data helps chose the most appropriate classifier. In this chapter, we will understand the nature of the sEMG signals by which the features set is developed. Alongside the algorithm using which the channels of sEMG signals and accelerometer data were selected is explained.

3.1 Features

As mention before, action potentials created by neurons cause a muscular contraction which leads to the movement of limbs. The smallest functional unit to describe this neural control is called a Motor Unit. The signal measured by a electrode will consist of motor unit action potentials(MUAPs) of all the active motor units detectable under the electrode site which will be electrically superposed and observed as a bipolar signal with symmetric distribution about zero. sEMG signals are random in nature and Sherif in his dissertation has emphasized the non-stationary nature of the EMG and used an AR, integrated moving average (ARIMA) representation. He characterized the non-stationary nature of the EMG during different phase of muscle activity [17]. Apart from this, there aren't many statements defining the signal.

With this knowledge, the features can be extracted from the raw signals to represent the signals in an informative and non-redundant way. It is a technique to reduce the dimension of the input into manageable sets while still accurately and completely describing the original data. Using features reduces the chances to overfit especially in algorithms that do no have many weights. But on the contrary, features that best represent the data have to formulated for the algorithm to work best. Hence a reason why deep learning algorithms gained popularity over machine learning algorithms is the automation of this step. As deep learning algorithms have built- in feature extractors thereby reducing human interaction. Irrespective of this fact, researchers tend to use extracted features as the input to deep learning algorithms also. But in this thesis, features extracted are used to represent the signals to the machine learning algorithms only. Based on the literature survey performed, the following features were decided to be extrcted and used in this thesis.

Root Mean Square (RMS)

Root mean square is modeled as amplitude modulated Gaussian random process which relates to constant force and non-fatiguing contraction.

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \tag{3.1}$$

Mean Absolute Value (MAV)

MAV is normally used as an onset detection index in EMG non-pattern recognition and in clinical application [14]. MAV feature is an average of absolute value of the EMG signal amplitude in a segment, which can be defined as Equation:

$$MAV = \frac{1}{N}\sum_{i=1}^{N} |x_i| \tag{3.2}$$

Variance (VAR)

Variance is defined as an average of square values of the deviation of that variable.

$$VAR = \frac{1}{N-1}\sum_{i=1}^{N} x_i^2 \tag{3.3}$$

Waveform Length (WL)

Waveform length (WL) is a measure of complexity of the EMG signal [14]. It is defined as cumulative length of the EMG waveform over the time segment and can be expressed as:

$$WL = \frac{1}{N-1}\sum_{i=1}^{N-1} (x_{i+1} - x_i)^2 \tag{3.4}$$

Mean Frequency (MNF)

An average frequency which is calculated as sum of product of the EMG power spectrum and the frequency divided by total sum of the spectrum intensity .

$$MNF = \frac{\sum_{j=1}^{M} f_j P_j}{\sum_{j=1}^{M} P_j} \tag{3.5}$$

where f_j is the frequency of the spectrum at frequency bin j and p_j is the power spectrum at bin j.

Zero Crossing (ZC)

Zero crossing is a measure of frequency information of the EMG signal that is defined in time domain [14].

$$\begin{gathered} ZC = \sum_{i=1}^{N-1} [sgn(x_i x_i + 1) \cap |x_i - x_i + 1| \geq threshold] \\ sgn(x) = \begin{cases} 1 & \text{if } x \geq threshold \\ 0 & otherwise \end{cases} \end{gathered} \tag{3.6}$$

Multiple Hamming Windows (MHW)

Multiple hamming windows are an original version of multiple time windows method. An idea of the multiple time windows method is to capture the change of EMG signals energy with respect to time by various multiple windowing functions.

$$MHW = \sum_{i=1}^{N-1} (x_i^2 w_{i-k}); k = 1, 2, , , K \tag{3.7}$$

where w is the hamming window.

With respect to major pattern recognition algorithms, an important step in them is the feature engineering step. Here a trait unique to each category of data is found such that it is common to most of the data in that class and very different for the other classes. There can be multiple traits or a group of traits combined can be used as a single differentiate. These traits form the features of the data. Over the years of research, several efficient combinations of features from both the the time, frequency and power domains have been tested. In this thesis, we have evaluated the above features and will use them as input to the machine learning algorithm in the following chapter4.

3.2 Input Set

In this thesis both sEMG and accelerometer signals are used as input so there are total 19 channels of data per subject per activity, i.e. 8 channels form 1^{st} MyoArm band and another 8 channels form 2^{nd} MyoArm band and lastly 3 channels from the accelerometer in 1^{st} MyoArm band. For an exercise these are too many channels hence firstly the number of channels are reduced and the least co related channels are used to represent the activity. Futher details pertaining to the process of selection of channels is explained in the section 3.2.1.

While trying to classify these signals, the categories to classify them into can be viewed in three different perspectives. Using these perspectives different styles of training, testing and validation batches are designed. Based on which, further in the thesis we discuss the effectiveness of the selected algorithm.

3.2.1 Channel Selection

In statistics, correlation is defined as the dependence or association between two or more variables. In this thesis, we use cor-relationship as a basis to chose the channels as two MyoArm bands are used to collect data, leading to a chance of repetition in the measurement of sEMG signals. As explained before, a channel contains a sEMG signal produced by the muscle it is placed over. With the setup of DB5, there is a chance that another electrode from the 2^{nd} MyoArm band is also residing on the same muscle and thereby recording the same MUAP with a time lapse. This causes a duplication of signal and is better avoided. Alongside, for the accelerometer data, the exercises are very limited and hence can still be best represented with just two dimensions kinetics data. By reducing the number of channels fed as input, the computational power required to process will reduce drastically along with chance to reduce possibilities of over-fitting.

The Pearson correlation 3.8 is used to chose the channels that best represent an exercise. The Pearson correlation measures the linear correlation between two variables. For this thesis, we choose 8 least correlated channels from all of the sEMG channels pertaining to the exercise, while choosing two channels from the three channels of IMU data. Initially, the correlation between channels per exercise, with respect to each is calculated. This leads to the formation of a 16 x 16 x 52 correlation matrix for the sEMG signals. Similarly, a 3 x 3 x 52 matrix is developed for the accelerometer data. From these matrices the least co-related 8 channels for sEMG signals are selected while 2 of the least co-related accelerometer channels were chosen respective to the exercise. These selected channels of data are used in the future experiments.

$$A = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}} \tag{3.8}$$

3.2.2 Augmentation

The original dataset contains only 60 data samples per exercise. To increase the variance in the data within a category and to increase the size of the data, data augmentation was performed. A popular data augmentation technique used with signals is to add white

Gaussian noise. Experimenting was done by adding different signal to noise ratio(SNR) based noises to the original signal and finally concluding to add noise with SNR = 25.

3.2.3 Training, Testing and Validation

Though the data used in this project comes from a pre-collected repository, in the long term goal of this thesis, the is a step where the data has to be collected from post stroke patients. As these subjects tend to be socially awkward, there will be a limitation on the number of trails a person can be requested to perform. Hence the thesis also aims at developing an algorithm for the device which will depend least on the customer's signals. Based on this, categories to classify were looked upon in three different perspectives, i.e. the training, testing and validation batches were split in three different styles.

Style 1 forms the set 1 in table 3.1, where the principle aim is to test if the algorithm depends on the subjects being used. That is, the training set consists of only 8 subjects data while the test dataset consist of another subjects data. This will prove if the network is independent of the subject. Similarly Style 2 forms the set 2 in table 3.1, where the principle aim is to test the performance of the algorithm with a peak of the customer's signals. Therefore in this set few of the trails of the customer is used in the training process also. In the last Style, set 3 3.1, the aim is to see if the algorithm classifies based on exercises. The aspect of different subjects is totally ignored in this case, it can be called the classical style.

Set	Training	Validation	Test
Set1	8 subjects 6 trails/ subject	1 subject 6 trails/subject	1 subject 6 trails/subject
Set2	8 subjects 6 trails/ subject	2 subject 3 trails/subject	2 subject 3 trails/subject
Set3	10 subjects 4 trails/ subject	10 subject 2 trails/subject	10 subject 2 trails/subject

Table 3.1: Table of different styles of train, test and validation datasets

Chapter 4

Methods

Deep learning, machine learning and artificial intelligence algorithms can be thought of as a set of Russian dolls nested within each other, beginning with the smallest and working out. Deep learning is a subset of machine learning, and machine learning is a subset of AI, which is an umbrella term for any computer program that does something smart. In other words, all machine learning is AI, but not all AI is machine learning, and so forth. These learning algorithms are the most famous computational tools used to classify data in recent times. Machine learning algorithms are adaptive methods where they learn to predict from past observations while deep learning algorithms consists of interconnected adaptive processing elements called neurons. An deep learning algorithm's working process is usually inspired from a biological processes. For this thesis, 4 algorithms were implemented, linear discriminant analysis(LDA) from machine learning algorithms and Multi-layer perceptron(MLP), Convolutional Neural Networks(CNN) and Recurrent Neural Networks(RNN) from deep learning algorithms. In this chapter these network's respective theory and concepts are explained. Concluding with the details of the structures and hyper parameters of the models used in this thesis.

4.1 Machine Learning

A basic definition of machine learning is that its algorithms that parse data, learn from that data, and then apply what theyve learned to make informed decisions. Machine learning fuels all sorts of automated tasks and spans across multiple industries, from data security firms hunting down malware to finance professionals looking out for favorable trades. Theyre designed to work like virtual personal assistants, and they work quite well. There are many Machine learning algorithms like, K- nearest neighbours, SVM and others, But from literature survey in chapter 2, we can see that the LDA structure provided the highest accuracy with respect to sEMG signals. Hence in this thesis, this algorithm is used to compare and infer from to develop the required network.

4.1.1 Linear Discriminant Analysis

The LDA algorithm uses the probability distribution of database with respect to each of the response classes, along with Bayes theorem to estimate the probability of an entry belonging to a certain class, with the assumption that the dataset per category is normally distributed. The algorithm is based on the analysis of two scatter matrices: within-class scatter matrix and between-class scatter matrix. These can be computed as follows:
Given a set of samples $x_1, \ldots, x_n$, and their class labels $y_1, \ldots, y_n$:

The within-class scatter matrix is defined as:

$$S_w = \sum_{i=1}^{n}(x_i - \mu_{y_i})(x_i - \mu_{y_i})^T \tag{4.1}$$

Here, μ_k is the sample mean of the k^th class.

The between-class scatter matrix is defined as:

$$S_b = \sum_{k=1}^{m} n_k(\mu_k - \mu)(\mu_k - \mu)^T \tag{4.2}$$

Here, m is the number of classes, μ is the overall sample mean, and n_k is the number of samples in the k^th class.

Then, multi-class LDA can be formulated as an optimization problem to find a set of linear combinations (with coefficients w) that maximizes the ratio of the between-class scattering to the within-class scattering, as

$$\hat{w} = \underset{w}{\operatorname{argmax}} \frac{w^T S_b w}{w^T S_w w} \tag{4.3}$$

The solution is given by the following generalized eigenvalue problem:

$$S_b w = \lambda S_w w \tag{4.4}$$

4.1.2 Principal Component Analysis

This algorithm was used along with principal component analysis(PCA), a dimension reduction algorithm. PCA [18] aims at finding a projection that best represent the dataset. A high dimensional data can be represented using a set of basis vectors. For a d dimensional data $\vec{x}$, if the data is expressed by means of basis vectors $\vec{b_i}$ as follows:

$$\vec{x} = \sum_{i=1}^{d} y_i \vec{b_i} \tag{4.5}$$

Then, PCA tries to represent the data with reduced number of dimensions m, as follows, such that the error will be minimum.

$$\vec{x} = \sum_{i=1}^{m} y_i \vec{b_i} + \sum_{i=m+1}^{d} a_i \vec{b_i} \tag{4.6}$$

where, the error is,

$$\Delta x = \sum_{i=m+1}^{d} (y_i - a_i) \vec{b_i} \tag{4.7}$$

By using mean-square error to quantify the error in equation 4.7, it can be found that,

$$\epsilon^2(m) = \sum_{i=m+1}^{d} \lambda_i \tag{4.8}$$

where λ_i is the eigen values corresponding to the eigen vectors b_i. Thus to minimize the error rate, it is needed to select the eigen vectors for reduction purpose in way that the summation of the eigenvalues for those eigen vectors will be minimum. The optimal choice of basis vector is the eigen vectors of the co-variance matrix. To reduce the dimension, error is calculated following equation as a ratio of sum of all eigenvalues and then principal eigen vectors are selected based on a certain tolerance error rate. The original dataset is projected to the basis vectors with less dimensions.

In this work, an error rate of around 3% has been used to select eigen vectors which resulted in a 30% dimensionality reduction, i.e. for the 8 channels and 5 features selected, there were 40 features initially to represent a data. Upon applying PCA, only 12 features have been selected and used during classification.

4.2 Deep Learning

In practical terms, deep learning is just a subset of machine learning. It technically is machine learning and functions in a similar way, but its capabilities are different. Basic machine learning models do become progressively better at whatever their function is, but they still need some guidance. If an ML algorithm returns an inaccurate prediction, then an engineer needs to step in and make adjustments alongside, when using data of high dimensions, there is a requirement to reduce it with manual parameters. But with a deep learning model, the algorithms can determine on their own if a prediction is accurate or not and even extract its own features.

Deep is a technical term as it refers to the number of layers in a neural network. Multiple hidden layers allow deep neural networks to learn features of the data in a feature hierarchy, because simple features (e.g. two pixels) recombine from one layer to the next, to form more complex features (e.g. a line). Nets with many layers pass input data through more mathematical operations than nets with few layers, and are therefore more computationally intensive to train. Computational intensity is one of the hallmarks of deep learning, and also a setback as this would mean that these algorithms will require more data to train them.

4.2.1 Multi Layer Perceptron

The fundamental building block of any Neural Network(NN) algorithm is a perceptron. A perceptron is a linear classifier that forms a linear combination of its input and its input weights while obtaining the output through a nonlinear activation function. The equation 4.9 is the mathematics behind a perceptron and figure 4.1 is the structure of it.

$$y = \phi(W^T x + b) \tag{4.9}$$

where w denotes the vector of weights, x is the vector of inputs, b is the bias and ϕ is the non-linear activation function. The famous activation functions are sigmoid, Relu and Tanh.

With the perceptron as a building block, an MLP is built as a feedforward neural network which is trained using a supervised learning technique called Back-propagation. As the name suggests, a MLP consists of layers of perceptrons with a min of 3 layers i.e. an input, hidden and output layer. The hidden layer can consist of either a single layer or many layers. It is also called the universal aproximator. This layer consists of many perceptrons called as nodes. Each of these nodes in one layer are connected with a certain weight $w_{ij} w_{ij}$ to every node in the following layer as show in figure 4.2. Learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result. This is carried out through the famous

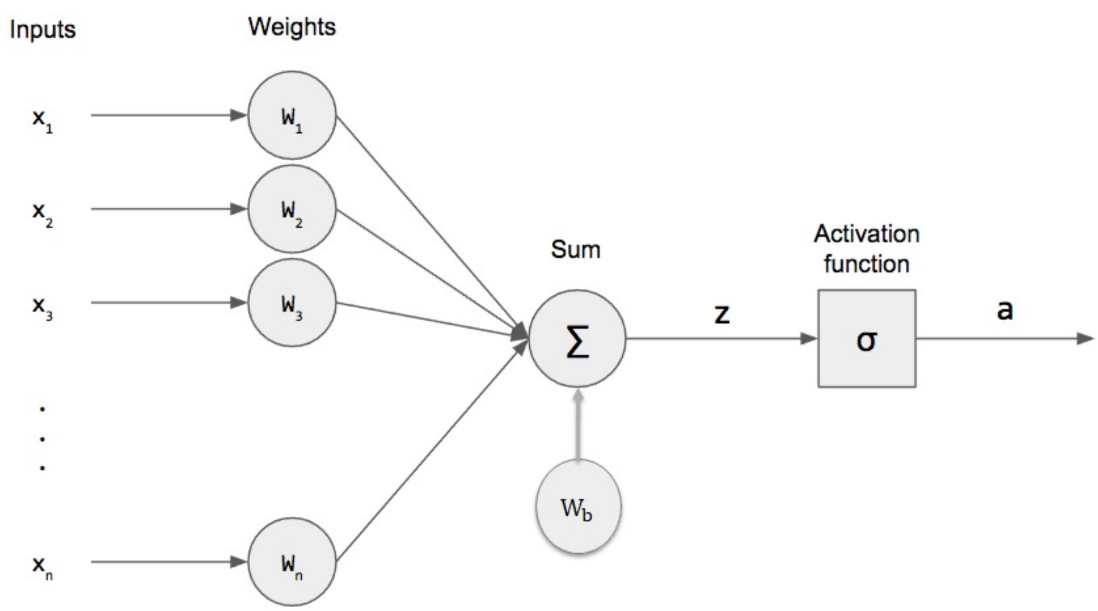


Figure 4.1: Simple perceptron structure

backpropagation algorithm which is explained in the subsequent subsection. Figure 4.2 is the structure and parameters of the MLP used in this thesis.

Backpropagation

The backpropagation algorithm was originally introduced in the 1970s, but its importance wasn't fully appreciated until a famous 1986 paper by David Rumelhart, Geoffrey Hinton, and Ronald Williams. It is the workhorse of learning in these NNs which gives us detailed insights into how changing the weights and biases changes the overall behaviour of the network. Backpropagation is shorthand for "the backward propagation of errors," since an error is computed at the output and distributed backwards throughout the networks layers. This error is calculated using a loss function which will be explained in the following section 4.2.1. In this algorithm, the chain rule of calculus and partial derivations of the error function with respect to the various weights and biases in the network are calculated to give

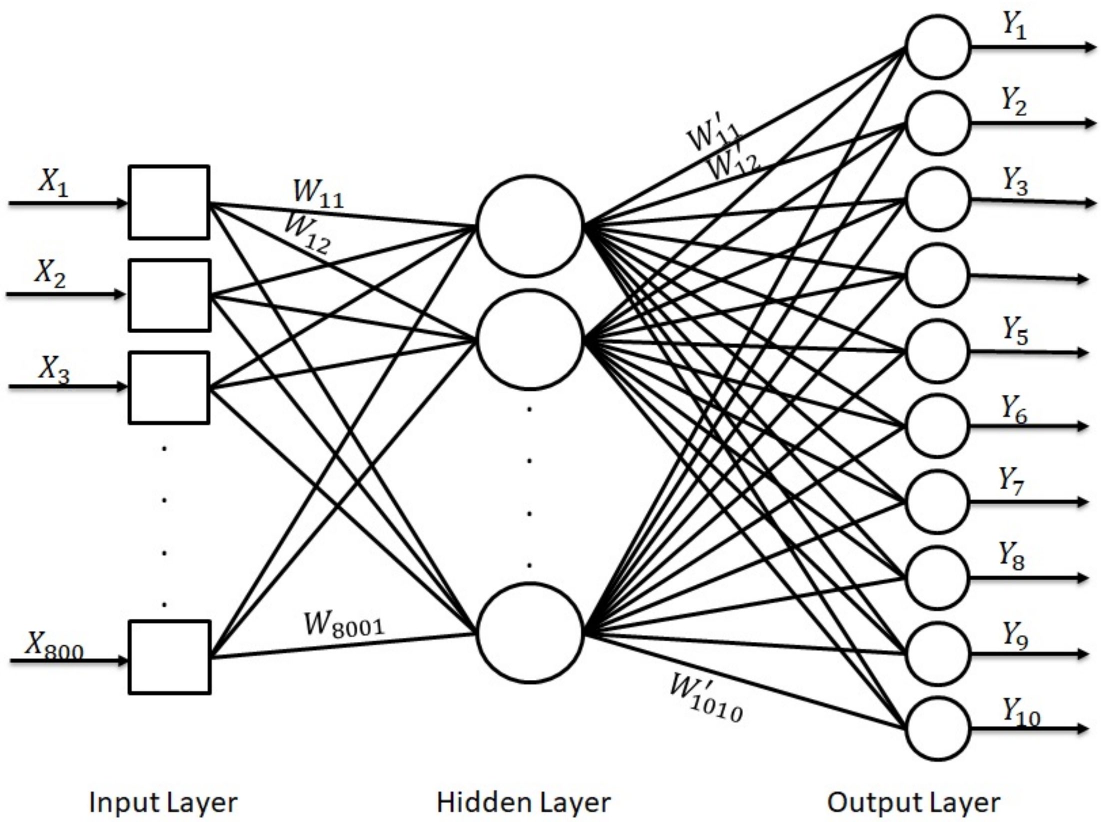


Figure 4.2: Structure of MLP used in thesis

the gradient. Which is a landscape of errors along which the parameters can be adjusted as the MLP moves one step closer to minimum error. This adjustment is performed by the optimization algorithms, which will be explained in section 4.2.1.

Loss function

Loss function is a performance metric on how well the NN manages to reach its goal of generating outputs as close as possible to the desired values. The loss function is a function that maps values of one or more variables onto a real number intuitively representing some "cost" associated with those values. For backpropagation, the loss function calculates the difference between the network output and its expected output, after a training example has propagated through the network. The most intuitive loss function is simply loss = (Desired outputactual output). There are several various loss functions that can be employed to train a network. The choice depends on the problem at hand. For this thesis, based on literature survey and experimental results, the final algorithms use categorical cross entropy. This would require the the output to be encoded into binary arrays with size equal to the number of classes i.e. a one-hot vector.

Optimisation Algorithm

There are various algorithms to optimize the NN, with each working in different ways. But their common goal is to update the weights and biases to reduce the error based on the error map derived by the backpropogation algorithm. They can be divided into two categories based on whether the algorithm is using a constant learning rate is or not. Learning rate is a hyperparameter i.e. a value that has to be fined tuned by a human. As its name suggests, it is the rate at which the error is learned by the network. This thesis experiments with both types of algorithms and the results are explained in 5.

4.2.2 Convolution Neural Networks

CNN algorithm revolutionized the deep learning field due its ability to efficiently process data by resorting to local connections, shared weights, pooling and layers. Although CNNs have the biggest impact in computer vision applications using 2D/3D images, relatively there hasn't been many implementations on time-series data. In this thesis we develop a CNN to classify the sEMG and IMU signals.

CNNs, like MLPs, are made up of perceptrons/neurons with learn-able weights and biases. Each neuron receives several inputs, takes a weighted sum over them, pass it through an activation function and responds with an output. The whole network has a loss function and propagates the loss in training through back propagation.

The input to a CNN is usually raw data unlike machine learning algorithms i.e. the sEMG and IMU signals are fed to CNN directly after pre-processing. The initial layers of the CNN extracts these features from the input layer and then the last few layers classify based on these features. CNNs compare images piece by piece to find features that would match in roughly the same positions in two images, CNNs get a lot better at seeing similarity than whole-image matching schemes.

As the CNN wont know the exact location of these features initially, so it will searching for them everywhere, in every possible position. To calculate the match to a feature across the whole image, the feature is made into a filter. These filters are then convoluted with the input to find a match. Hence they are called convolution neural networks. A map of matches location is created which is a filtered version of our original image. If the values it consists are close to 1 show it means they are a strong match, close to -1 shows strong matches for the photographic negative of our feature, and values near zero show no match of any sort. Each filter will produce an image really the shape of the input. All these processes are performed in the convolution layer.

CNN also has a pooling layer. Pooling is a way to take large images and shrink them down while preserving the most important information in them. It consists of stepping a small window across an image and taking the maximum value from the window at each step thereby also reducing the size of the image i.e. about a quarter pieces will be left as to from what it started with. The output of this layer will still consist of the same number of pictures as the previous convolution layer but with fewer pixels.

An important player in all this process is the activation function. As described previously, its purpose in a CNN is to introduce non-linearity. The layers explained till now are independent of each other, i.e. unlike MLPs, every neuron of the previous layer isn't connected with next layer. The cumulative result from one layer is transferred to the following layer and hence forth. But the fully connected(FL) layer the CNN is a feedforward layer hence structured like in MLPs. Fully connected layers are the primary building block of traditional neural networks, as Its the layer where classification is done majorly. They take the high-level filtered images and translate them into votes. In practice, several fully connected layers are often stacked together, with each intermediate layer voting on phantom hidden categories. In effect, each additional layer lets the network learn ever more sophisticated combinations of features that help it make better decisions.

Dropout layer is an optional layer in a CNNs structure. This layer is used only during training process and is omit during prediction. While training,g half of neurons on a particular layer will be deactivated as this will improve generalization. Dropout is a technique used to improve over-fit in neural networks.

The CNN used in this thesis consists of 8 layers. It uses the Relu activation function and Binary category loss function with a stochastic gradient descent(SGD) optimizer. The following figure 4.3 is the structure of the CNN used.

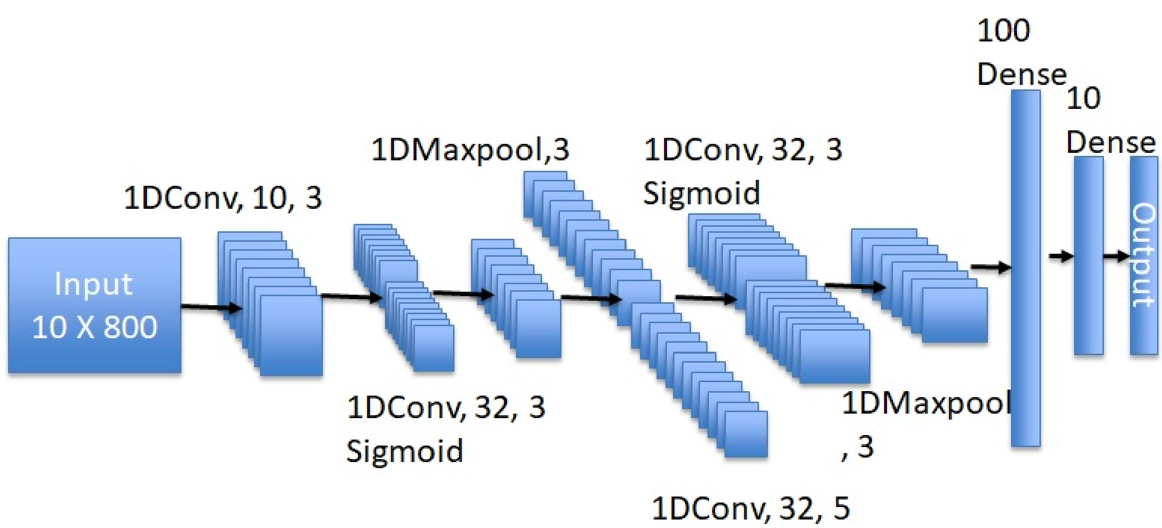


Figure 4.3: CNN structure

4.2.3 Recurrent Neural Networks

When the backpropagation algorithm was introduced, its most exciting application was to train RNNs. These networks differ substantially from MLPs and CNNs, since they possess dynamic memory, in the form of an internal state which can be altered by recurrent connections.

In this thesis we have used Long short-term Memory model of RNN. The architecture of a vanilla LSTM cell is depicted in Figure 4.4. As shown in the schematic of the LSTM cell, the output is connected back to the block input and all of the gates, through recurrent (lagged) connections. The architecture presented here consists of a revised version of the initial LSTM, which did not yet include a forget gate. With the addition of this gate, the network becomes able to learn continuous tasks, since it can at times forget and thus release memory. The formulas for the forward pass in a LSTM are given in equations following:

$$Z^{current} = g(W_z x^{current} + R_z y^{previous} + \theta_z) \tag{4.10}$$

$$i^{current} = g(W_i x^{current} + R_i y^{previous} + \theta_i) \tag{4.11}$$

$$f^{current} = g(W_f x^{current} + R_f y^{previous} + \theta_f) \tag{4.12}$$

$$c^{current} = i^{current} \odot z^{current} + f^{current} \odot c^{previous} \tag{4.13}$$

$$o^{current} = g(W_o x^{current} + R_o y^{previous} + \theta_o) \tag{4.14}$$

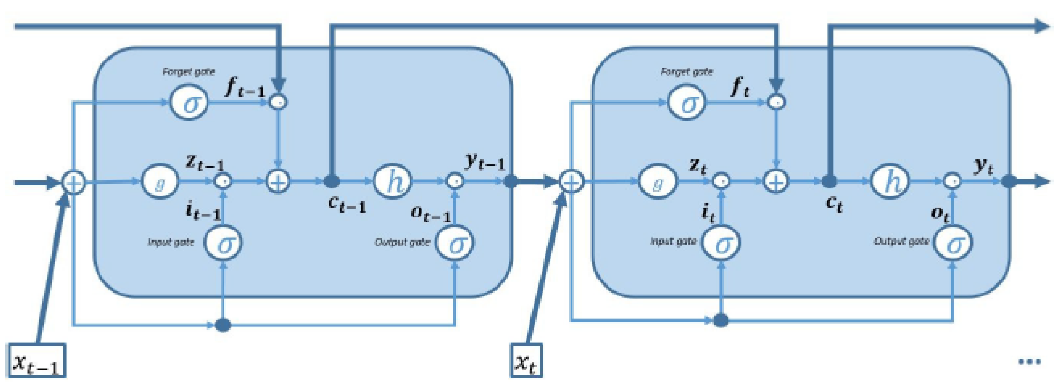


Figure 4.4: Vanilla LSTM cell

$$y^{current} = o^{current} \odot h(c^{current}) \tag{4.15}$$

Here $\odot$ denotes point-wise multiplication of two vectors. W and R are the weights matrices. They are noted differently since W consists of the input weights rectangular matrices and R are square recurrent weight matrices. In this thesis we use just a single layer of RNN consisting of 10 LSTM cells.

4.2.4 Performance Measures

The most straight forward way of evaluating the performance of a learning algorithm is to measure the classification accuracy, which is equal to the fraction of correct classifications with respect to the total number of classifications. However, this measure alone can give rise to deceivingly high performance results.

To introduce a few more important measure concepts lets look at the confusion matrix in table 4.1 initially. For binary classification problems, the confusion matrix gives a clear summary of the algorithms performance. The terms TP, TN, FP and FN refer to true positives, true negatives, false positives and false negatives, respectively. From these values many different accuracy measures can be extrapolated, such as precision (true positive rate) and recall (true negative rate) and others.

		Prediction outcome		
		p	**n**	**total**
actual value	**p′**	True Positive	False Negative	P′
	n′	False Positive	True Negative	N′
	total	P	N	

Table 4.1: Confusion Matrix

For this thesis we measure accuracy, recall, precision and F1 Score. Recall measures the number of correct samples selected as seen in equation 4.17 and precision calculates how many of the selected were selected samples as seen in equation 4.18. These equations can further be used to develop a more complex measure the F1-score 4.19 which is a harmonic mean of precision and recall.

$$Accuracy = \frac{TP + FN}{TotalNo.samples} \tag{4.16}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.17}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.18}$$

$$F1Score = 2 * \frac{1}{\frac{1}{pricision} + \frac{1}{recall}} \tag{4.19}$$

Chapter 5

Results And Conclusions

This chapter summarizes the results obtained from each model as well as the hyperparameter search. Initially the various models implemented on sEMG and Acc signals are compared to identify the best structure to use. With the network chosen, hyperparameter search was conducted leading to the development of the CNN structure with highest classification accuracy of 91%.

5.1 Comparison

From the literature survey discussed in chapter 2, 4 reference models were chosen and classified with, to compare the learning algorithms performances. The LDA structure which provided the highest accuracy was used to represent the machine learning domain. For deep learning algorithms representation, a simple single hidden layer MLP is used along with the CNN from [15] and modified RNN structure based on [12]. These algorithms with respective structures from their papers were implemented on various styles of train, test and validation datasets as explained in section 3.2.3. The following graph 5.1 plots the accuracy attained by each of the structures for the various styles of training, testing and validation datasets. When comparing the performance of machine learning algorithms vs deep learning algorithms, it can be clearly seen that the machine learning algorithm performs better. But to attain this accuracy, though the structure is borrowed from literature survey, the features used had to be extensively studied and fine tuned for this particular objective. Alongside, in the following paper [14], it can be seen that when the number of activities to be classified increases, the accuracy decreases from an ML algorithm while the complexity in the feature set increases. On the other side, without a fine tuned structure, the CNN algorithm has performed comparably good w.r.t the ML algorithm. Hence this thesis further develops this algorithm to classify the exercises with higher accuracy. In here it can be clearly seen that the CNN structure produced the highest accuracy of 91% for style 2 dataset. With respect to other datasets too, the CNN performed relatively well. Hence from this graph we can conclude that the CNN can classify this combination of signals for the application with a high accuracy.

5.1.1 Hyperparameters

While the CNN structure taken as reference contains only 7 layers while the final structure explained in figure 4.3 is the CNN proposed by this thesis has 10 layers. This structure is built upon exploring the various hyper parameters in a CNN structure. The learning rate, momentum and training batch size. The final learning rate used is 0.001 with a momentum of 0.8 using SGD optimizer with batch wise training with each batch containing 100 training samples. Over-fitting is the most likely occur-able training problem in deep learning algorithms, where during training the error of the training set keeps decreasing,

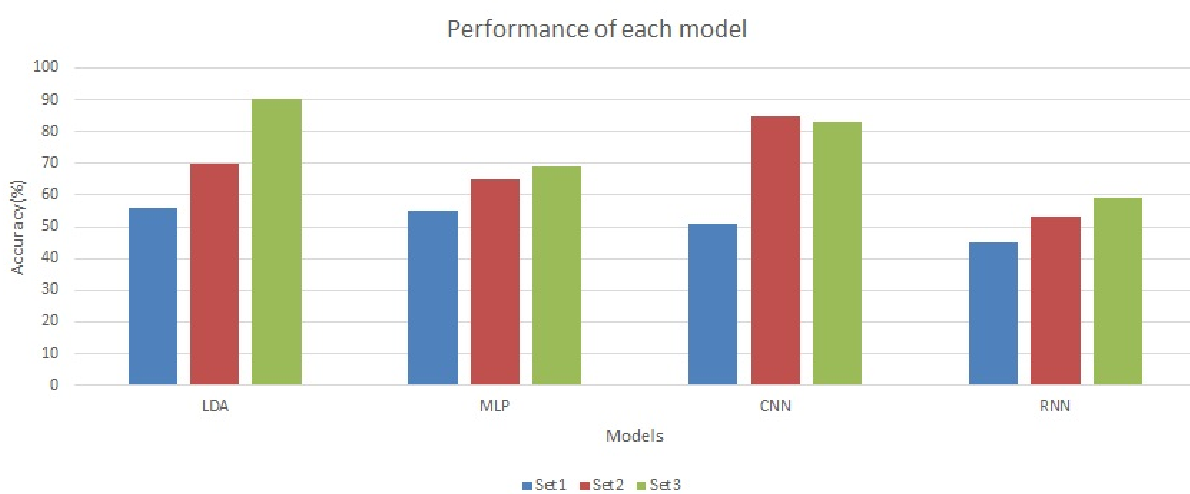
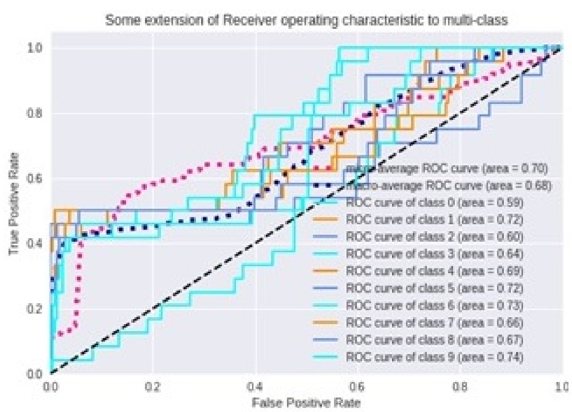


Figure 5.1: This graph plots the accuracy attained by each model w.r.t each of the inputs data set

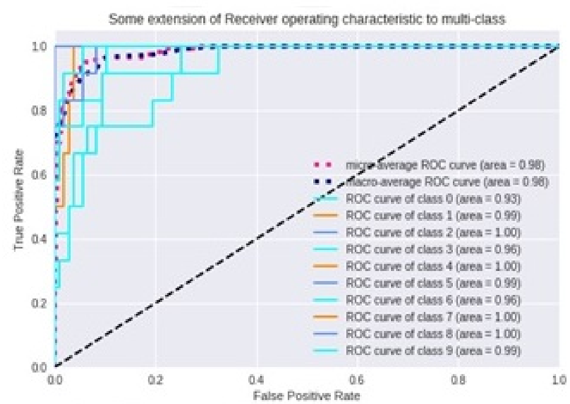
but the validation error stagnates or even rises. To prevent this early stopping is used which simply checks when or if this occurs and according stops training then. With these modifications the 4.3 is the CNN structure developed in this thesis that classifies these exercises with the highest accuracy of 91

5.2 Discussion

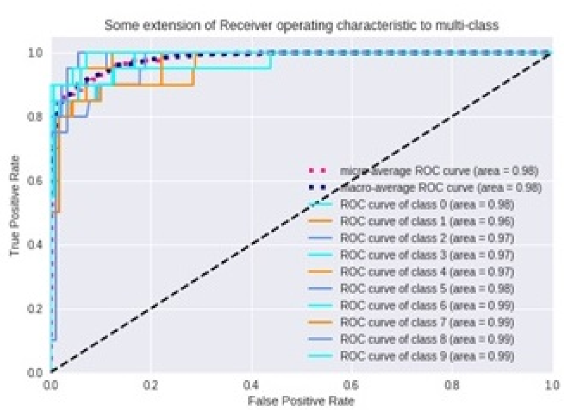
With the defined CNN structure, training and testing using the various styles of data was performed. The following figures 5.2 & 5.3 are these experiments respective ROC curves and performance matrices graphs.



((a)) ROC of Style 1 batch



((b)) ROC of Style 2 batch



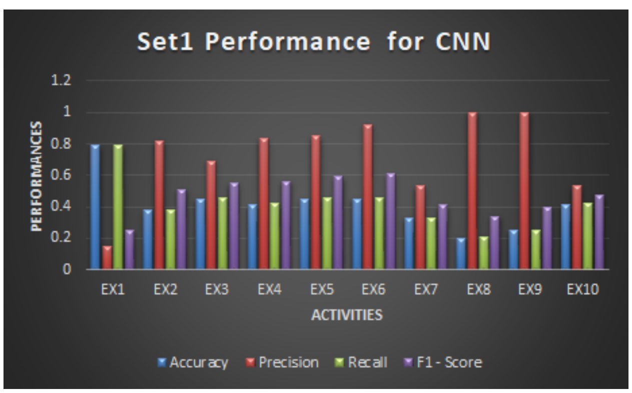
((c)) ROC of Style 3 batch

Figure 5.2: ROCs of the CNN for various styles of data

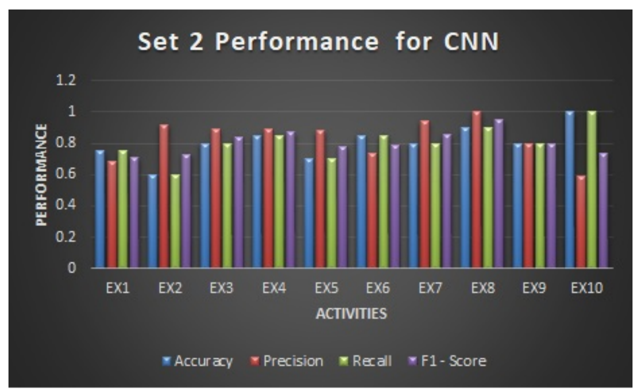
From the graphs it can be inferred that to classify the exercises used during stroke rehabilitation process, the described CNN structure provides the highest accuracy. With the various styles of training, testing and validation batches it can be seen that the CNN performs best for Style 2 hence the CNN would require few signals from the subject to train on before identifying them.

5.3 Future Work

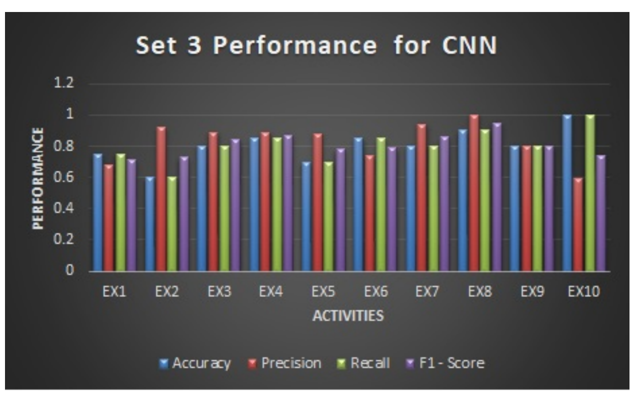
Irrespective of the high accuracy recorded, in the ROC it can be seen that the performance per activity isn't similar. This situation has to be explored further to find a solution. A likely cause would be the small size of dataset. This size issue could also have lead the RNN to perform poorly as RNN has more weights compared to CNN.



((**a**)) Style 1 batch



((**b**)) Style 2 batch



((**c**)) Style 3 batch

Figure 5.3: Performance graph of the CNN for various styles of data

Further on, the next steps to build the wearable device has to be performed which involves testing and improvising the network on real time data and on stroke patients sEMG signals. In addition to identifying the improvement of the stroke patient recovery process.

5.4 Conclusion

To conclude, the thesis aims at finding the learning network that could classify the exercise performed by a subject based on their sEMG and IMU data. These exercises are required in the rehabilitation process of stroke patients. This is the initial research to further develop an wearable device for stroke patient to monitor their recovery process.

This thesis makes use of the Database 5 from the NinaPro repository available online. It compares 4 of the highly performing learning algorithms and concludes that the CNN performs with the highest accuracy of 91% against a RNN, MLP and LDA. Alongside the dependency of the network on the subjects is also tested. Concluding that CNN's work best with the given dataset size and the required exercises with the condition that the CNN structure would require a small quantity of prior knowledge of the subject.

Bibliography

[1] Atzori, M., Cognolato, M., and Müller, H. (2016). Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands. *Frontiers in neurorobotics*, 10:9. 3

[2] Atzori, M., Gijsberts, A., Castellini, C., Caputo, B., Hager, A.-G. M., Elsig, S., Giatsidis, G., Bassetto, F., and Müller, H. (2014). Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Scientific data*, 1:140053. 7, 9

[3] Basmajian, J. V. (1962). Muscles alive. their functions revealed by electromyography. *Academic Medicine*, 37(8):802. 1

[4] Côté-Allard, U., Fall, C. L., Drouin, A., Campeau-Lecours, A., Gosselin, C., Glette, K., Laviolette, F., and Gosselin, B. (2019). Deep learning for electromyographic hand gesture signal classification using transfer learning. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2

[5] Engdahl, S. M., Christie, B. P., Kelly, B., Davis, A., Chestek, C. A., and Gates, D. H. (2015). Surveying the interest of individuals with upper limb loss in novel prosthetic control techniques. *Journal of neuroengineering and rehabilitation*, 12(1):53. 6

[6] Goen, A. and Tiwari, D. (2013). Review of surface electromyogram signals: its analysis and applications. *World Academy of Science, Engineering and Technology, International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 7(11):1429–1437. 1

[7] Han, J.-S., Song, W.-K., Kim, J.-S., Bang, W.-C., Lee, H., and Bien, Z. (2000). New emg pattern recognition based on soft computing techniques and its application to control of a rehabilitation robotic arm. In *Proc. of 6th International Conference on Soft Computing (IIZUKA2000)*, pages 890–897. 6

[8] Khushaba, R. N. (2014). Correlation analysis of electromyogram signals for multiuser myoelectric interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(4):745–755. 8

[9] Krasoulis, A., Kyranou, I., Erden, M. S., Nazarpour, K., and Vijayakumar, S. (2017). Improved prosthetic hand control with concurrent use of myoelectric and inertial measurements. *Journal of neuroengineering and rehabilitation*, 14(1):71. 10

[10] Kuzborskij, I., Gijsberts, A., and Caputo, B. (2012). On the challenge of classifying 52 hand movements from surface electromyography. In *2012 annual international conference of the IEEE engineering in medicine and biology society*, pages 4931–4937. IEEE. 12

[11] Labs, T. (1999). Thalmic labs myo armband description. 9

[12] Laezza, R. (2018). Deep neural networks for myoelectric pattern recognition. *Chalmers University of Technology*. 31

[13] LLC, C. S. (1999). Cyberglove systems llc. 9, 10

[14] Negi, S., Kumar, Y., and Mishra, V. (2016). Feature extraction and classification for emg signals using linear discriminant analysis. In *Advances in Computing, Communication, & Automation (ICACCA)(Fall), International Conference on*, pages 1–6. IEEE. 1, 3, 7, 14, 15, 32

[15] Park, K.-H. and Lee, S.-W. (2016). Movement intention decoding based on deep learning for multiuser myoelectric interfaces. In *2016 4th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–2. IEEE. 7, 31

[16] Pizzolato, S., Tagliapietra, L., Cognolato, M., Reggiani, M., Müller, H., and Atzori, M. (2017). Comparison of six electromyography acquisition setups on hand movement classification tasks. *PloS one*, 12(10):e0186132. 10

[17] Sherif, M. H. (1982). A stochastic model of myoelectric signals for movement pattern recognition in upper limb prostheses. 13

[18] Smith, L. I. (2002). A tutorial on principal components analysis. Technical report. 21

Vita

Sadhu Sri Ravali was born in Machilipatnam, Andhra Pradesh, India, to the parents of Sadhu Krishnayya and Sadhu Bhagyalakshmi. She studied till her high school in the Singaporean education system before shifting and graduating with an undergraduate degree of BTech in Electrical Engineering from M.S.Ramaiah Institute of Technology from the Indian educational system. With her passion in robotics engineering she continued to pursue her masters degree in University of Tennessee in electrical engineering. Here she ventured into the field of Artificial Intelligence.