UNIVERSITY OF CENTRAL OKLAHOMA

Edmond, Oklahoma

Jackson College of Graduate Studies

Development of an Artificial Intelligence Algorithm for the Analysis of Wheelchair Movements

A THESIS APPROVED FOR

THE DEGREE OF MASTER OF SCIENCE IN COMPUTER SCIENCE

Dr. Jicheng Fu

Dr. Hong Sung

Dr. Shuai Zhang

In partial fulfilment of the requirements

For the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

By

Daniel Williams

Oklahoma City, Oklahoma

2020

Development of an Artificial Intelligence Algorithm for the Analysis of Wheelchair Movements

Thesis Title

Daniel Williams

Author's Name

12/7/2020

Date

Jackson College of Graduate Studies at the University of Central Oklahoma

A THESIS APPROVED FOR

THE DEGREE OF MASTER OF SCIENCE IN COMPUTER SCIENCE

By

Jicheng Fu Digitally signed by Jicheng Fu Date: 2020.12.07 16:24:48 -06'00'

Committee Chairperson

Hongchi Shi

Committee Member

Committee Member

Committee Member

THESIS ABSTRACT

Monitoring wheelchair user movement is an essential task for assessing a wheelchair user's mobility and helping them maintain an active lifestyle. Research has shown that increased mobility leads to healthier overall lifestyles, and that people with disabilities are at an increased risk for sedentary lifestyles and the health problems associated with that lifestyle, including cardiovascular disease, obesity, and the development of pressure ulcers (WHO, 2014). Existing technology for analyzing wheelchair user mobility data requires the use of external sensors that must be purchased and maintained (Warms & Belza, 2004). To improve the ease by which mobility data is maintained and analyzed, a wheelchair user can utilize existing technology, such as smart mobile devices, to gather and analyze motion data. This study will focus on the development of a recurrent neural network (RNN) that is trained using wheelchair user data collected from smart devices attached to the wheelchair or wheelchair user. The benefit of collecting data this way is that it does not require the use of additional sensors or equipment, as most wheelchair users will already have access to a smart device capable of collecting movement data. The study found that it was feasible to meaningfully analyze data gathered from a smart device using an RNN. The raw data is analyzed with the RNN to gather information about the mobility of a wheelchair user. The final analysis includes the total time spent moving, number of bouts of movement, and the longest bout of movement. This resulting data could be used by a wheelchair user or healthcare professional to help assess healthy lifestyle habits.

KEYWORDS: *Artificial Intelligence (A.I.); Recurrent Neural Network (R.N.N.); Wheelchairs; Smart Devices; Mobility Assessment.*

Chapter I

Introduction: Wheelchair Movement Analysis Motivation

1.1 Background

Individuals with inactive lifestyles are at an increased risk for a variety of health issues. Some of these issues include cardiovascular disease, obesity, and hypertension (WHO, 2014). The World Health Organization reports that nearly two million deaths per year are due to causes associated with lack of physical activity. Wheelchair users in particular are more likely to lead inactive lifestyles that contribute to increased risk for these diseases along with other issues such as the formation of pressure ulcers (Nooijen et al., 2015). A study done with 197 sedentary overweight or obese adults concluded that self-monitoring physical activity using sensor-based technology with real-time feedback improved weight loss in those adults (Shuger et al., 2011). While self-monitoring has been proven to help increase the amount of physical activity individuals engage in, collecting and assessing movement data, especially for wheelchair users, is difficult to do manually and usually requires the use of sensors and technology to collect and analyze the data.

The research conducted in this study aims to collect and analyze movement data from wheelchair users without the need for extraneous sensors, by utilizing existing sensors in smart devices worn by the wheelchair users, e.g., smartwatch and smartphone. The study will build on current methods for collecting and analyzing wheelchair movement data through smart device sensors by incorporating the use of deep learning algorithms to classify the movement characteristics of the wheelchair. These deep learning algorithms will produce more accurate movement data, with the benefit of working in real-time and the ability to provide immediate feedback to the wheelchair user.

1.1.1 Traditional Movement Analysis

Capturing human movement for the purpose of data analysis has become more sophisticated as technology has improved. Marey (1873) and Muybridge (1878) were some of the first researchers to use photography to quantify and analyze human movement (Mündermann et al., 2006). Now it is possible to capture and analyze human movement data in real time with sensor-based technology. One such example is a pedometer, which is capable of counting human steps with electronics. Pedometers work by capturing the movement of a person's step by swinging a small metal pendulum wired into an electronic counting circuit (Woodford, 2020). These devices and similar ones are very practical for gathering movement data as they are generally small, inexpensive, and often integrated into other devices. Fitness applications for smart phones such as Google Fit and MyFitnessPal make use of a phone's internal sensors to gather movement data. These apps use the gyroscope and accelerometer to capture the lateral, longitudinal, and vertical movements to estimate steps (Wise & Hongu, 2009). Pedometers and similar technologies gather useful movement data for walking but are not viable options for individuals whose primary form of locomotion is in a wheelchair. The movements of a person propelling a manual wheelchair or operating a powered wheelchair are not able to be captured in the same way as a person walking, whose hip movements are used to capture movement data.

1.1.2 Wheelchair Movement Characteristics

The movement characteristics of wheelchair users are fundamentally different than the movements of an individual walking. Manual wheelchair users primarily propel themselves by using their arms. However, while during a traditional walking cycle there is always movement in the body, a wheelchair user can coast with the momentum imparted during a push to continue moving without moving any part of their body. Wheelchair propulsion involves two distinct phases, the propulsion phase where the extremities are activated to induce motion, and a recovery phase where hands are not directly engaged with the wheels (Woude et al., 2001). Some research has identified different propulsion patterns, that differ based on the level of impairment and skill of the wheelchair user. For powered wheelchair users, the limbs may have little or no movement while the chair is moving, as pressure only needs to be applied to the joystick by the hand to induce propulsion. These differences make traditional methods for gathering and analyzing movement data impossible for wheelchair users.

1.1.3 Movement Data from Smart Devices

Modern smart phones contain a variety of sensors for collecting movement data. Among these sensors are accelerometers, gyroscopes, and GPS trackers. GPS-enabled smart phones are typically accurate within 16 feet under open sky, with decreasing accuracy in buildings, under trees, etc. (van Diggelen & Enge, 2015). Accelerometers and gyroscopes measure device movement. These movements include rotation, shakes, swings, etc. and these measurements reflect not only the movements of the phone but also the movements of the environment that the phone is in. This makes it possible to record movement data for a person holding a smart phone, or the movement data of a wheelchair that the smart phone is mounted to.

1.2 Health Issues Associated with Wheelchair Users and Inactivity

The need for movement analysis is based on the understanding that wheelchair users who move more and for longer periods are less likely to develop sedentary lifestyles than users who have fewer periods of wheelchair movement. A study of women that took place over 12 years found that women with the highest amount of sedentary time had an increased risk of several diseases that would ultimately result in death (Seguin et al., 2014). The result of this study was that diet and exercise alone did not completely reduce the risk of disease, if a person spent a significant amount of time sedentary, they were still at a higher risk of disease. For wheelchair users who have no use of lower extremities, it is even more important to maintain active lifestyles to reduce the risks of these diseases.

1.2.1 Cardiovascular Disease, Obesity, and Diabetes

Cardiovascular disease is a blanket term used to describe a range of conditions that affect the heart. Some of the diseases that fall under this category include coronary artery disease, heart rhythm problems, and heart defects. Cardiovascular disease as a term is generally used when referring to conditions that involve impeded blood vessels, which are either narrowed or blocked, leading to a heart attack. Usually this is due to a buildup of fatty plaques in the arteries, inhibiting blood flow to organs and tissues. This is the most common cause of cardiovascular disease, and is caused by problems including unhealthy diet, lack of exercise, being overweight, and smoking. Because wheelchair users are more likely to be sedentary, their lack of exercise puts them at an increased risk for this disease.

The lack of exercise also puts wheelchair users at an increased risk for developing obesity. Studies show a rise in obesity among the general American populace but especially for people with disabilities. Adults with disabilities have an estimated obese population of 25% to 31% while adults without disabilities have an estimated obese population of 15% to 19% (Froehlich-Grobe & Lollar, 2011). Obesity is associated with increased risk for other diseases, including diabetes, high blood pressure, and certain cancers.

Weight gain due to lack of exercise, the increase of fatty tissue, and the way a person with disabilities body processes insulin also put them at an increased risk for diabetes. Diabetes occurs when the blood glucose is too high, due to the pancreas no longer supplying enough insulin to help the glucose make it to the body’s cells. High blood glucose from diabetes leads to other health issues including stroke, kidney disease, and eye problems.

All of the diseases discussed are related to sedentary lifestyles and lack of exercise among other factors. While it may not be possible to completely reduce the risks of developing these diseases by maintaining active lifestyles, it has been shown that active lifestyles help to reduce the risk of developing these diseases. It is for this reason that the need to properly track movement data for wheelchair users is so important. A wheelchair user that is tracking their movements and providing this data to their healthcare providers will be able to assess their current lifestyle and identify if any changes are necessary to ensure healthy levels of activity. For manual wheelchair users, the act of propelling the wheelchair can burn up to 120 calories in half an hour, three times more than someone performing the same action in a motorized wheelchair (Conger

& Bassett, 2011). While the act of moving itself is not enough to assess an individual's lifestyle, an increase in movement is associated with increased levels of physical activity. Working with a healthcare professional the mobility information can be helpful in determining how much a person's movement is contributing to their overall lifestyle.

1.2.2 Pressure Ulcer Formation

In addition to the risks associated with sedentary lifestyles, wheelchair users, especially powered wheelchair users, are at risk of developing pressure ulcers. Pressure ulcers are the result of prolonged pressure on the skin, causing damage to the skin and underlying tissue. Powered wheelchair users, many of whom do not have feeling of their lower bodies, are much more likely to develop pressure ulcers (James et al., 2020). Preventing pressure ulcers is largely done by exercises designed to relieve stress on the lower body and reduce pressure on specific areas.

Traditional tools for assisting in the prevention of pressure ulcers are largely based on assessments made by healthcare professionals and not applicable for home use by wheelchair bound individuals (AHRQ, 2012). There also exists expensive technology for automatic exercises used primarily in hospitals that involve seats or beds that move an individual automatically to assist in pressure ulcer prevention. This study will also attempt to help close the gap in pressure ulcer prevention for wheelchair users by allowing them to automatically monitor the amount of time spent doing exercises. This information will enable the user to identify whether or not they are spending adequate time performing exercises and can be used with a healthcare provider for further guidance.

Chapter II

Introduction: Artificial Intelligence and Machine Learning (ML)

2.1 Background

The process of analyzing the characteristics of the large data sets provided by the sensors requires the use of artificial intelligence. The data received from sensors is noisy and inconsistent. Artificial intelligence algorithms, specifically recurrent neural networks, are able to analyze the sequence of data from sensors quickly and provide meaningful output (Mitchell, 1997).

2.1.1 Overview of Artificial Intelligence

Artificial intelligence (AI) algorithms are algorithms that are designed to enhance their own efficacy by learning from input data. They differ from traditional algorithms in that they do not follow strict rules or give predetermined responses. AI algorithms take in information from multiple sources, analyze the data, and produce responses based on that analysis.

The benefit of using AI over other data processing algorithms is their ability to be used with data that is not the same every time, or data that is contaminated with noise. AI algorithms are also able to instantly classify large data sets, removing the need for tedious manual classification.

2.1.2 Introduction to Machine Learning

Machine learning (ML) is an application of AI in which systems automatically learn to improve based on past experience without being explicitly programmed to do so. ML algorithms learn by building a model based on sample data..

2.2 Machine Learning

ML algorithms learn by using a training set of data to construct a mathematical model, which is then used to make predictions or decisions on input data without explicitly being told what to do. ML algorithms are designed to evolve over time, so that the more data they are trained with, the more accurate their predictions are. ML algorithms are used in various applications, such as computer vision, image recognition, speech recognition, and self-driving cars (Mitchell, 1997). For the purposes of analyzing wheelchair movements, the goal of the ML algorithms is to take in large amounts of sensor data and learn how to classify that data to give characteristics about the movement of a wheelchair.

2.2.1 Machine Learning Approaches

There are several different types of ML algorithms, which differ based on their approach to classifying data, the type of data they work on, and the problem they are designed to solve. The different types of algorithms learn in different ways, which result in different applicational use. Each type of algorithm was considered, and the final decision was made based on the information that will be available for the given problem statement. In particular, the sensor data will be read from a device and output in a raw format. The data will be timestamped and organized sequentially. The data will likely contain noise as the available sensors are sensitive to small vibrations and are not perfectly accurate. There will also be a training set available, with the raw data correctly categorized as moving or not moving. With this information the available learning algorithms are discussed, with consideration on which algorithms are applicable for the study.

Unsupervised learning algorithms are one type of algorithm considered. Unsupervised algorithms are used on data sets where the input data is not labeled. The model created by the algorithm attempts to find structures present in the data and extract rules to organize new data based on the found structures. The K-means clustering algorithm (Hartigan, 1975), for example, can take a raw data set and organize it into objects based on each data points distance to other points.

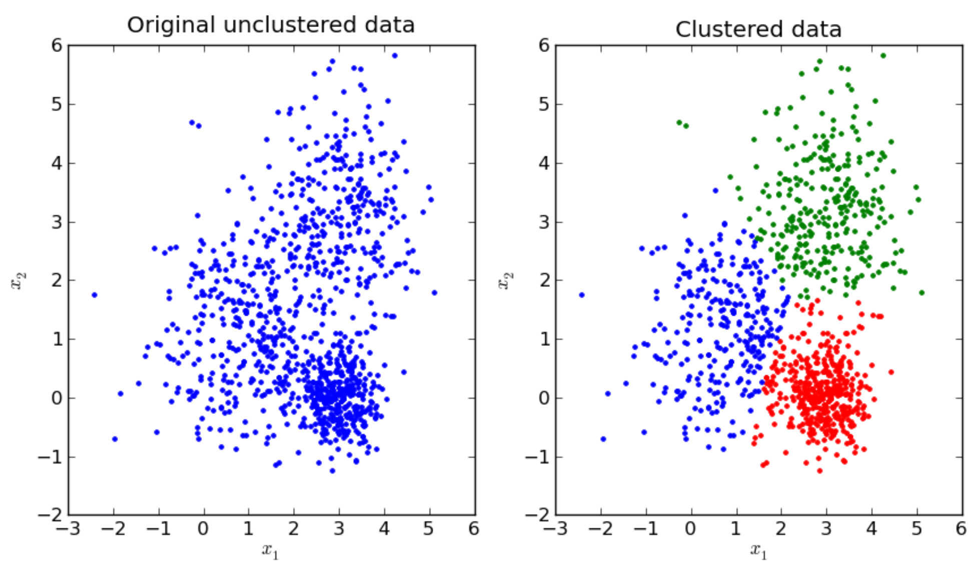

Figure 1. Example of K-means clustering (Viswarupan, 2017)

While this would in theory be able to organize the raw sensor data into groups based on the intensity of the sensor readings, it would sacrifice the sequential ordering of the data and make further analysis difficult.

Reinforcement learning algorithms are another type of algorithm considered. Reinforcement algorithms learn by attempting to maximize reward for a given task (Mitchell, 1997). The algorithm tries to make a sequence of decisions and is either rewarded or penalized for those decisions based on metrics specified by the programmer. The algorithm learns over time which set of actions will produce the highest reward. This form of learning is not well suited to analyzing data from a sensor as it is difficult to represent the data as a set of decisions, with performance-based metrics. For this reason, reinforcement learning algorithms were not considered for this study.

Supervised learning algorithms are the final type of algorithm considered. They learn by having a set of training data, with input data mapped to correct output data (Hastie et al., 2009). Supervised learning algorithms are trained iteratively, where an objective function is used to give a resulting output based on one or more inputs. If the output does not match the training data, the objective function is modified, and the process begins again. This continues until the objective function reaches a target accuracy, or the function fails to become more accurate after a set number of iterations. This form of learning fits the problem description, because a training set is available for use by the learning algorithm. Because supervised learning is able to learn from past experience with sequential data, and able to utilize a training set, it was chosen as the focus area for this study.

Review of Literature

2.3 Supervised Learning

While supervised learning was identified as the best candidate for this study, there are multiple implementations of supervised learning to consider. The "No Free Lunch" theorem states that there is no best algorithm for all problem statements (Wolpert & Macready, 1995). Choosing an appropriate supervised learning algorithm for the given problem statement involves looking at various metrics, including the variance of training data sets, the amount and complexity of training data available, the amount of noise in the output values, etc. Some previous studies using ML for the analysis were considered while choosing the appropriate algorithm.

2.3.1 K-Nearest Neighbor

For example, a 2013 study on wheelchair movement characteristics used the K-Nearest Neighbors (KNN) algorithm (Fu et al., 2013). The KNN algorithm uses the distance between points in a set of data points to organize the data. Specifically, the group uses a KNN classification algorithm that classifies each point based on the majority of its neighboring points. By using this supervised algorithm, the group was able to smooth the noise in the raw sensor data.

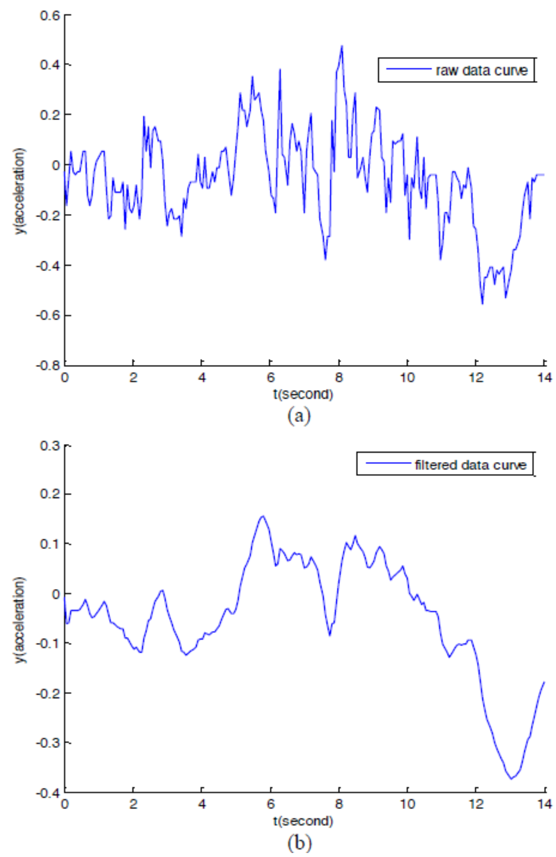

Figure 2. KNN smoothed data (Fu et al., 2013)

The group then use the smoothed data from both the accelerometer and gyroscope to roughly estimate the movements of a wheelchair. This study was effective in classifying wheelchair movements in batches collected and analyzed after the movements were performed. However, the KNN algorithm does not take into account the sequential ordering of the data other than to make determinations about each data points neighboring points, which were sequentially ordered in the referenced study. The data collected for this paper, however, is time-stamped which can be used with an algorithm that has a memory component in the learning process. The algorithm used for the referenced study was also only able to be used after data was collected in batches, and the algorithm had to relearn and smooth the curve of each batch of data. In order to get real-time data analysis for wheelchair movements, it is necessary to have an algorithm that can be taught beforehand and categorize data as it comes in from the sensors.

2.3.2 Linear Regression

In another study done for the purposes of analyzing wheelchair movement, a research group collected sensor data and ran the data through a curve fitting algorithm that produces a regression curve based on the sensor data (Fu et al., 2018). Linear regression algorithms work by creating a linear equation with a set of input values to produce a predicted output value. A simple regression equation could be in the form:

$$y = B_0 + B_1 x$$

Where y is the predicted output based on the coefficients B_0, B_1 and the input value x. A machine learned linear regression algorithm will change the coefficients to attempt to produce a line that best fits the data. More input variables can be used to potentially increase the accuracy of linear regression algorithm, however it should be noted that using too many inputs could result in overfitting the data, reducing the efficacy of the algorithm on new data. The group used a linear regression algorithm on a modified version of the sensor data to produce a line where peaks in the data represented periods of wheelchair movements.

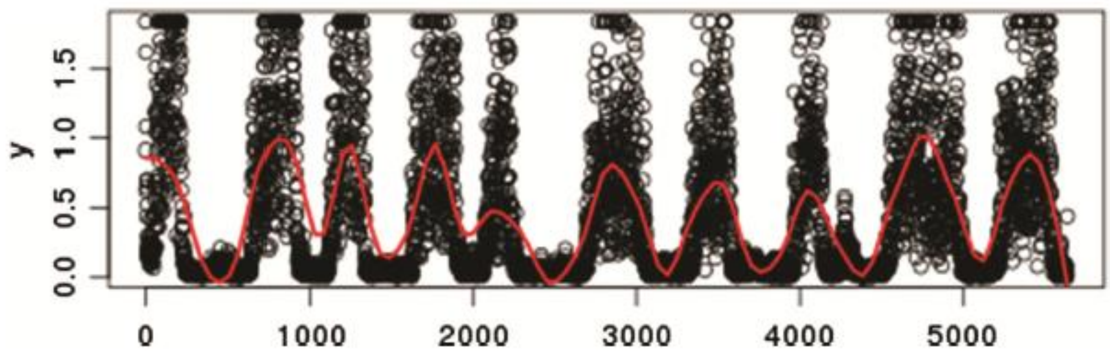

Figure 3. Linear regression curve fitting (Fu et al., 2018)

While this approach for analyzing wheelchair movement was largely successful, it had some error in correctly predicting the duration of movement in each bout, with an average error of 19%. This error was due to the fact that the duration of a bout was estimated based on the peaks and valleys of the modified data set. For this study, a more complex learning algorithm that is able to take into account the time-stamped data to produce more accurate bout duration readings will be considered.

2.2.2 Neural Networks

Artificial neural networks are supervised learning algorithms loosely based on biological neural networks like the human brain. They work by creating a collection of neurons, where each neuron takes one or more weighted inputs and produces an output.

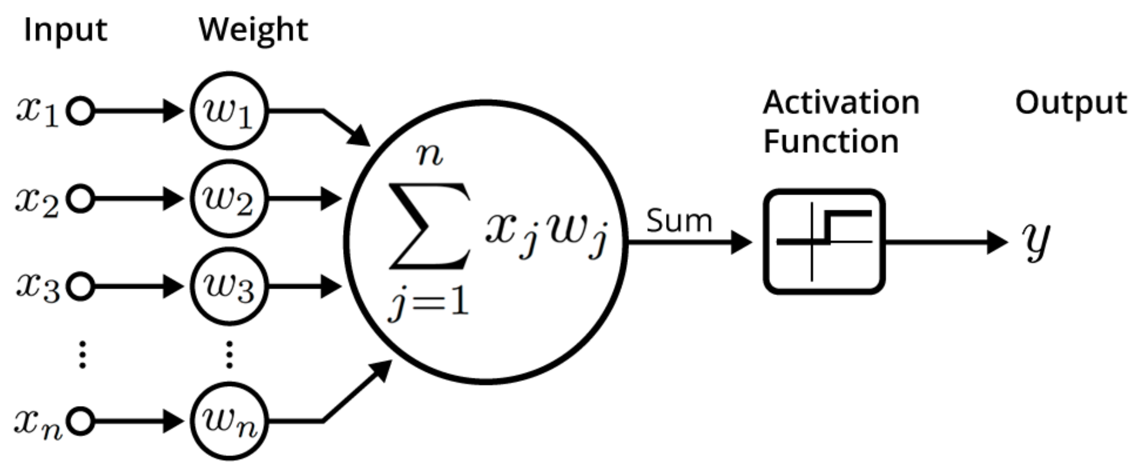

Figure 4. Artificial neuron (Saxena, 2017)

The neurons are grouped into layers with an input layer, output layer, and usually one or more hidden layers. The input layer accepts the data from the original source, that data is then passed through the hidden layers where they are acted upon by an internal weighing system that alters the values and produces an output. Neural networks learn by comparing their outputted values to the training values and adjusting the weights for the hidden layers to increase the accuracy of the network. This adjustment is done by backpropagation, where the outputted error is backward propagated through the network to adjust the weights.

One of the benefits of a neural network is that once the network has been trained it can very quickly classify new data in real-time without the need to further train the network. This is beneficial for this study where new sensor data can be automatically run through the network and classified for real-time movement analysis. In addition to real-time analysis the neurons are able to use activation functions to aid in data classification. This allows for the use of sigmoid functions to transform the output into binary classifications for moving or stationary. Several popular forms of neural networks exist, with the most popular being feedforward neural networks. These networks are the most basic form of neural network where data travels only forward through the network. Convolutional neural networks are another form of neural network that are used primarily in processes such as image classification (Krizhevsky et al., 2017).

2.2.3 Recurrent Neural Networks (RNN)

RNNs are a form of neural network that work especially well on sequential information. While a traditional neural network assumes that all inputs are completely independent, RNNs have a form of memory by performing tasks sequentially where the output for one node is dependent on the previous computations.

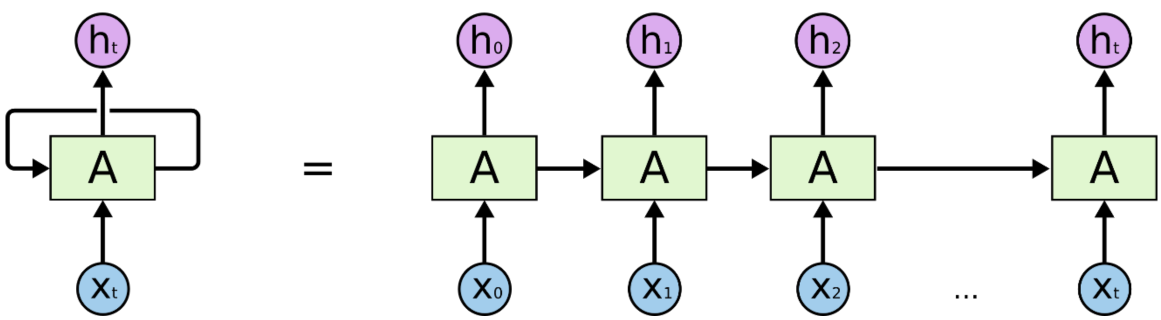

Figure 5. Unrolled recurrent neural network (Olah, 2015)

As shown in the figure above, data in a RNN is looped so that some information from the neuron is passed back into the neuron, essentially turning the network into a series of identical networks with different inputs and outputs, each of which passes along information to the next network. For the purposes of this study, the ability to pass along temporal information is useful because the current state of a wheelchair's movement characteristics influences the next state of the wheelchair's movement characteristics. The movement of the chair is affected by friction and the acceleration of the chairs motor or the arms propelling of the wheel. For instance, if accelerometer is reading a high acceleration value in the x direction, it is likely that the next value read by the sensor will be similar, either slightly higher or slightly lower depending on the movement state of the wheelchair.

2.2.4 LSTM Networks

Long short-term memory (LSTM) networks are a further refined form of RNN. While RNNs are able to hold memory about previous inputs in theory, in practice they generally cannot hold long term dependencies (Bengio, et al., 1994). LSTM networks solve this problem by using a module with a different structure than traditional RNN modules. Traditional RNN modules usually have a structure consisting of a single function, LSTM modules however have four interacting network layers.

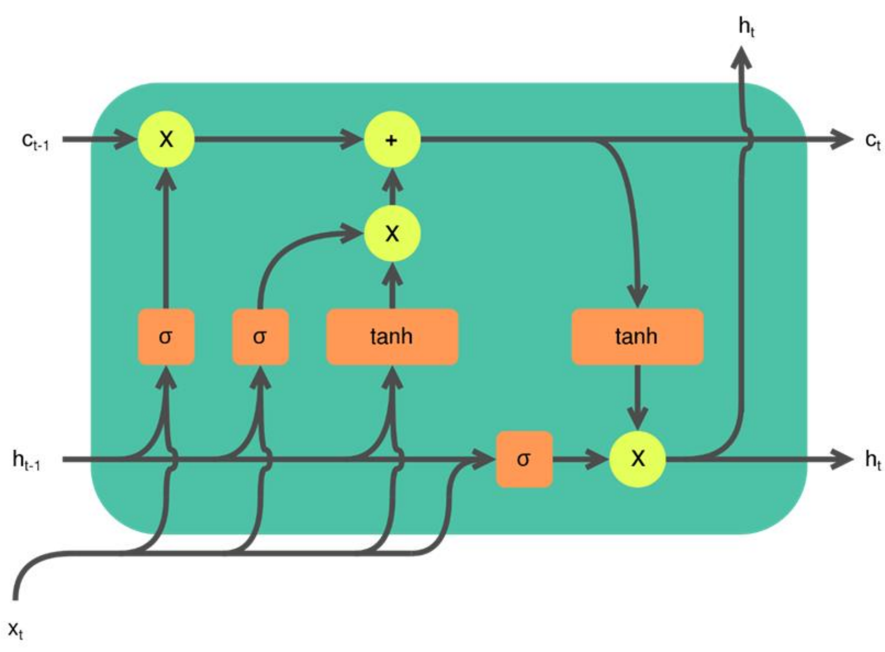

Figure 6. LSTM cell structure (Wikipedia, 2020)

LSTM modules are usually referred to as cells. The cell state c_t is the chain that runs through the entire network, and the LSTM adds or removes information to the cell state at each cell. The first step in the LSTM is to decide whether the information at the cell state should be kept or forgotten. The first slayer in the LSTM is called the "forget gate layer". It is a sigmoid layer that takes the hidden state vector from the previous cell h_{t-1} and the input vector for the current cell x_t, and outputs a value between 0 and 1. A 0 represents "forget this state completely" while a 1 represents "remember this state completely". In the context of this study, this would be where information about the current movement state is remembered or forgotten. If it appears that the wheelchair is no longer moving, the cell may forget the moving state that was passed into the new cell.

The next two layers of the cell update whatever is kept from the old cell state into the new cell state. The sigmoid layer called the "input gate layer" produces a list of which inputs to update and the tanh layer creates a vector representing the new candidate values. These are combined to update the cell state. This would correspond to the new movement state identified being added to memory, replacing the old movement state.

The last step is to create the output for the cell and update the hidden state h_t. A sigmoid layer controls which parts of the cell state to output, the current cell state is pushed through a tanh function to get the values between -1 and 1, then multiplied with the output of the sigmoid gate so only the relevant parts of the cell state are transformed and pushed to the output. This is where relevant information about the movement characteristics would be identified to move into the next cell state.

LSTM networks fit the data available for this study well. They take into consideration the context for the data, the current state of the wheelchair, and after being trained can very quickly categorize new data in real-time. Training the LSTM network will take some time, but this can be done once and then used for all subsequent bouts of movement. For these reasons, LSTM networks will be used in the study to train an RNN on wheelchair movement characteristics.

Chapter III

Research Purpose

3.1 Purpose Statement

Existing technologies for analyzing wheelchair movements and maneuvers, and for assisting in necessary exercises to promote healthy lifestyles such as sensors placed on the wheels of the chair or automatic seat adjustment hardware are more costly and complicated than working with technology already present for the wheelchair user. The purpose of this study is to use the wheelchair users' existing technology, namely their smartphone or smart watch devices equipped with accelerometers in conjunction with an RNN to provide useful data about wheelchair movements and use that data to construct an application to assist wheelchair users and healthcare professionals in assessing wheelchair user mobility.

3.2 Method

In order to assess wheelchair user mobility, this study looks at whether or not it is possible to gather data from a smartphone or smart watch and train a recurrent neural network to analyze that data to identify bouts of movement. For the purpose of this study, the total number of bouts, the longest bout, and total amount of time spent moving are used to assess the wheelchair user's mobility.

3.2.1 Equipment and Technology Used

The majority of the data used in this study was collected from powered wheelchairs. The accelerometer data was collected from a Google Nexus 5 and Google Pixel 2 running Android version 11.0. The phone was either mounted in a phone holder attached to the arm of the wheelchair or was placed flat on an inclined table attached to the wheelchair. Some data was also obtained from the Fossil Sport Smart Watch (DW9F2) running Wear OS version 2.5.0.

3.2.2 Python Development Environment

The data preprocessing and RNN programs were developed in Python 3.6.7 using Google's Colab notebook development environment. The Colab development environment allows developers to use rich text formatting along with executable code in a single document. Google Colab also provides libraries for importing and saving files.

The Python Pandas 1.1.0 library was used to read the comma separated data files containing the accelerometer data. Pandas was also used to construct the data frames, which are 2-dimensional labeled data structures. These data frames hold the labeled rotational axis data and the timestamp data obtained from the smart devices.

The Python Numpy 1.19.0 library was used to apply functions to the matrices necessary for correct formatting of the accelerometer data. The Seaborn and MatPlotLib libraries were used to create graphs inside the Colab notebook. The Tensorflow 2.3.0 library was used to create the RNN.

3.2.2 Method for Gathering Movement Data

In order to measure wheelchair user mobility, movement data from wheelchair users was collected using the accelerometer of the user's smart devices. Two applications were created to collect the data, one for a smartphone running the Android operating system, and one for a smart watch running the WearOS operating system. The functionality of the applications were very similar. The applications connected to the device's accelerometer using the SensorManager class. The polling rate for the data was set to be the same as the refresh rate for the Android UI mode (14-16 Hz). This polling rate provided data quickly enough to account for movements in the wheelchair without having a large impact on the battery of the devices.

The sensors output data as three-dimensional vector representing acceleration along each axis, excluding gravity. This data, along with the timestamp in milliseconds was saved to a csv file for further processing.

3.2.3 Factors Affecting Data

For this experiment multiple factors that could affect the accuracy of the result were identified and suitable mitigation techniques were implemented. For each risk factor identified, at least one solution was created to mitigate the risk. Factors negatively affecting this experiment could produce errors in the training set of data, the resulting accuracy of the RNN algorithm, or misrepresent the population that is the subject of the experiment. Because this experiment is intended to assess the mobility of a wheelchair user for the purposes of providing the results to a healthcare professional in order to create a personalized exercise program, it was important for risks to be adequately mitigated.

Different types of wheelchairs, either manual or powered, would produce different types of accelerometer data, and could affect the outcome of the experiment. For this experiment only powered wheelchair data was collected, however the resulting RNN was tested against manual wheelchair data and the results are discussed later in this study. Individuals in the same wheelchair also move differently to one another, and so data was collected from multiple people controlling the powered chair.

The placement of the device, the type of device used, and the accuracy of the device's accelerometer are also factors that could affect the outcome of the experiment. To mitigate these risks, different devices were tested on, with different placements on the wheelchair. Two different phones with different hardware were used, namely a Google Nexus 5 and Google Pixel 2. The different hardware on these two phones ensure that the accuracy of the sensor will not affect the results. The device was also placed using different holders to ensure different device orientation did not negatively affect the results. The devices were placed on a table mounted to the chair and in a phone holding mount attached to the arm of the chair. These different mounting solutions ensured that the device was in different orientations and could experience different types of motion that would be typical for a device mounted on a wheelchair.

3.3 Experimental Study: Analyzing Data with an RNN

The device accelerometer reads the acceleration of the device along each axis, excluding the acceleration from gravity. The sensor measures the acceleration applied to the device $A_d$by measuring the forces applied to the sensor itself F_s using the relation:

$$A_d = -g - \sum F_s/mass$$

where g represents the acceleration due to gravity. For the linear acceleration, the effect of gravity is accounted for and the equation can be simplified to:

$$A_d = -\sum F_s/mass$$

This equation simply represents the negative sum of all forces applied to the sensor divided by the mass of the device (AndroidDev, 2020).

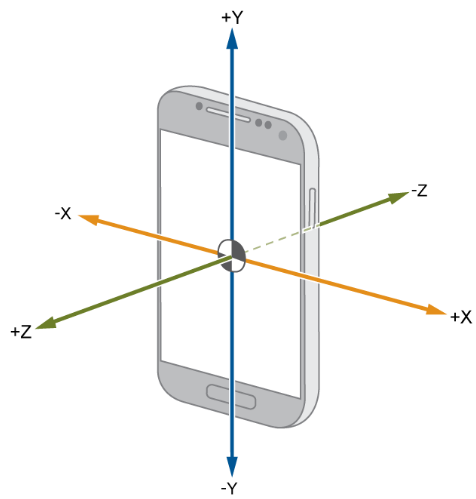

Figure 7. Illustration of linear acceleration axes (Mathworks, 2014)

The acceleration data collected from the devices was in a raw format. Each line of the file contained the *x*, *y*, and *z* axis data along with the timestamp represented by milliseconds since January 1st, 1970 UTC. The collected data can be represented as a sequence of accelerations:

$$D_C = \{c_1, c_2, \ldots, c_n\}$$

where $c_i = \langle a_i^x, a_i^y, a_i^z, t_i \rangle$ with ($1 \leq i \leq n$ and $t_1 \leq t_i \leq t_n$) and a_i^x, a_i^y, a_i^z represent the accelerations of the device measured along each axis, at time t_i.

In order to obtain meaningful data from this information, the raw acceleration data is transformed into a set of jerk data. Jerk is the derivative of action with regard to time:

$$f = \frac{d\alpha}{dt}$$

where f is the jerk while α is the acceleration.

If we consider jerks in three axes, we can obtain the overall jerk using the equation:

$$D_f = \left\{ f_1, f_2, \dots, f_{n-1} \middle| f_i = \sqrt{f_{ix}^2 + f_{iy}^2 + f_{iz}^2} \ and \ 1 \leq i \leq n \right\}$$

Where f_{ix}, f_{iy}, f_{iz} are defined as the difference between two consecutive accelerations since the difference in time between two consecutive accelerations is small enough to be negligible. Using this equation provides a single jerk value to represent each sensor reading, which is used to determine if movement was read from the wheelchair.

3.3.1 Python RNN Programs

To analyze the data, two python programs were created. A data preprocessing program was created to organize and clean the raw data, and an RNN data processing program to create an RNN, train it with labeled data, test the accuracy of the RNN against data that has already been labeled, and to run the algorithm on unlabeled datasets and analyze the results.

3.3.2 Data Preprocessing Program

The data preprocessing program was created to apply the transformation to the raw data and clean the data to remove erroneous elements that occurred to due to noise in the sensor. The data manually is read into the program using the files library as part of the google.colab suite. A data frame is constructed with four columns, one for each acceleration axis, and one for the timestamp. Once the data is loaded into the program, the accelerations are transformed into a series of jerk values with associated timestamp.

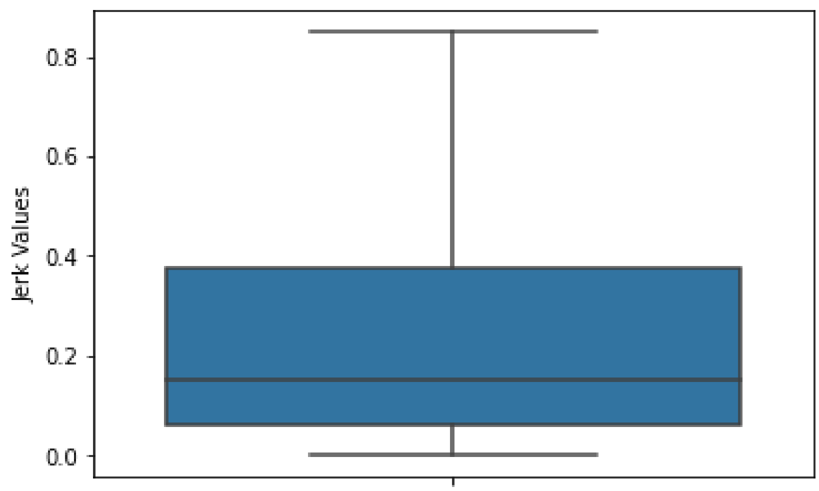

Figure 8. Example boxplot showing data collected from a series of bouts

In order to remove extreme outliers in the data that would likely be caused by errors in the sensor, the inner quartile range, upper and lower quartiles was calculated for each set of data, as shown in Figure 8.. Zero is defined for the lower quartile since the values can never fall below zero, and any values identified above the upper quartile are replaced with the value of the upper quartile. While the wheelchair is in motion, the acceleration along one or more of the axes would increase, resulting in a higher jerk value. A low value indicates the wheelchair is stationary.

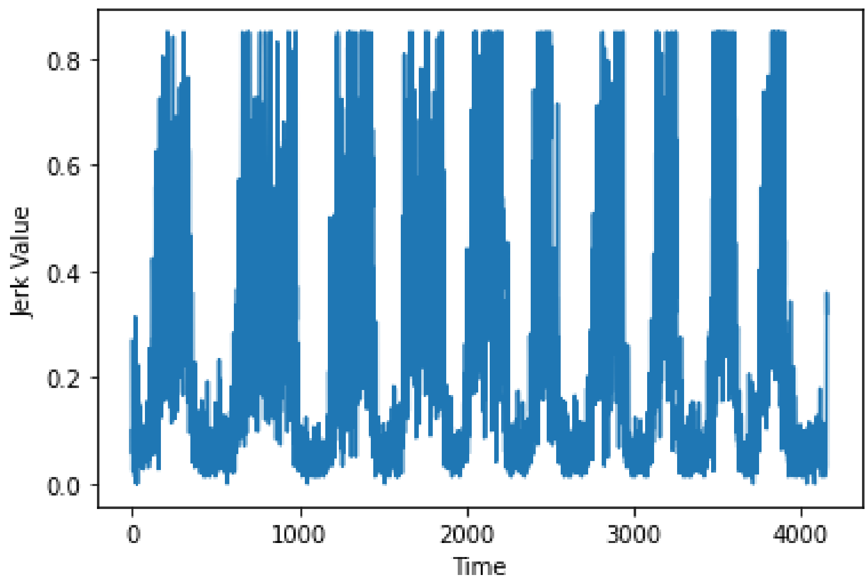

Figure 9. Graph of all adjusted jerk values for one data set

As shown in Figure 9, the data collected from the sensors contains noise making it difficult to analyze. There are clear patterns in the jerk data, but the variation makes it difficult to precisely tell where the movement starts and stops. The data sets used for training also contain the correct labels for each sensor reading, so it was possible to do a manual analysis to find the approximate threshold to use to determine whether the jerk value should be categorized as moving or not moving, in this case a value of 0.25 was used.

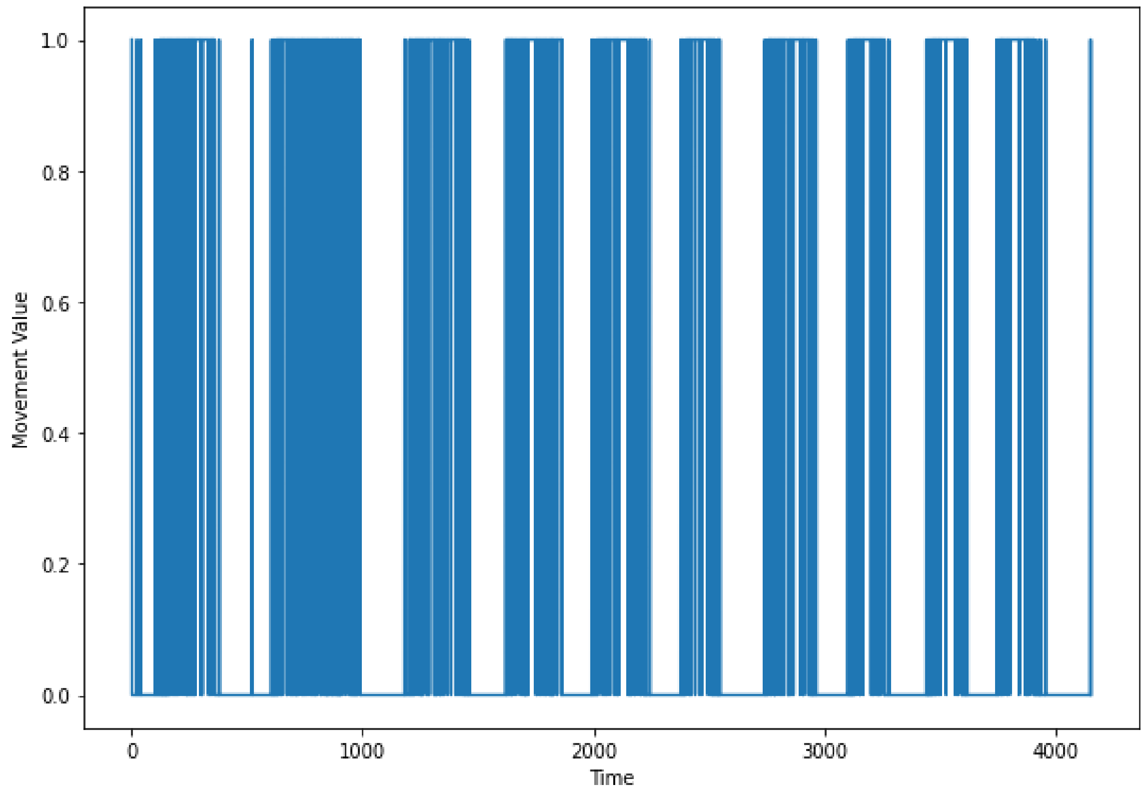

Figure 10. Movement Data after threshold is applied

A min-max scalar was applied to the data using the threshold resulting in a dataset where all values are marked either zero or one as shown in Figure 10. While the patterns are clearer in this dataset, there is still too much noise for a proper analysis. To further refine the set so that it can be used as training data for the RNN, the set was split into groups of 10, where the most common label for each group was assigned as the label for the entire group. This change resulted in a data set where with very clearly marked bouts of movement. There were occasionally noise values that made it into the final training set, so a small algorithm to further clean the data was developed that looks at the neighbors for each value to determine if a value was incorrectly classified. It is assumed that since the sensor data is gathered at a fast rate and bouts of movement last long enough for many sensor outputs to be present in each bout that a single differing value in the middle of set is determined to be incorrectly classified and automatically updated to the correct value.

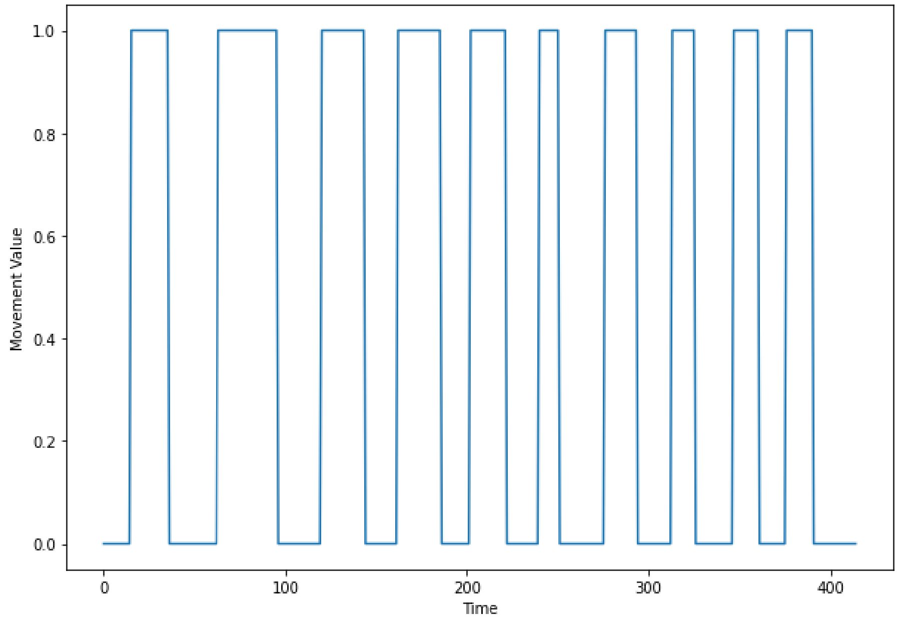

Figure 11. Correctly labeled training set data

This data set is then used to train the RNN which will then be able to automatically classify sensor data as it is received from the sensors. The final step to prepare the training data is to convert the data from a single movement value into a set of features for the RNN LSTM cells. Each group contains ten elements, all with the same label, so a new data frame is created with ten values, one for each movement value in the group. The final data frame format contains twelve columns, one for each weight $x_0, x_1, \ldots, x_9$ and column for the label to train the RNN, and the timestamp.

3.3.3 RNN Data Processing Program

The RNN program takes the cleaned training data from the preprocessing program, trains the RNN, then runs the RNN with another set of cleaned data to test the results. The first step in this process is to separate out a validation set from the original training data. In this study, one hundred rows were taken from the training set to use as a validation set. The RNN is configured using the sequential model, where each layer has exactly one input tensor and exactly one output tensor. The RNN is also configured to monitor the value loss of the RNN and stop the algorithm early when the value loss stops improving. The RNN is created with two layers, the LSTM layer, with 10 LSTM units, the input of which contains 10 data items, and a time distributed dense layer with a sigmoid function to ensure the output is either a one or zero to signify motion or stationary. After the RNN has been configured, it is trained with the training data set and validation set, then run against a separate test set to check the algorithms accuracy.

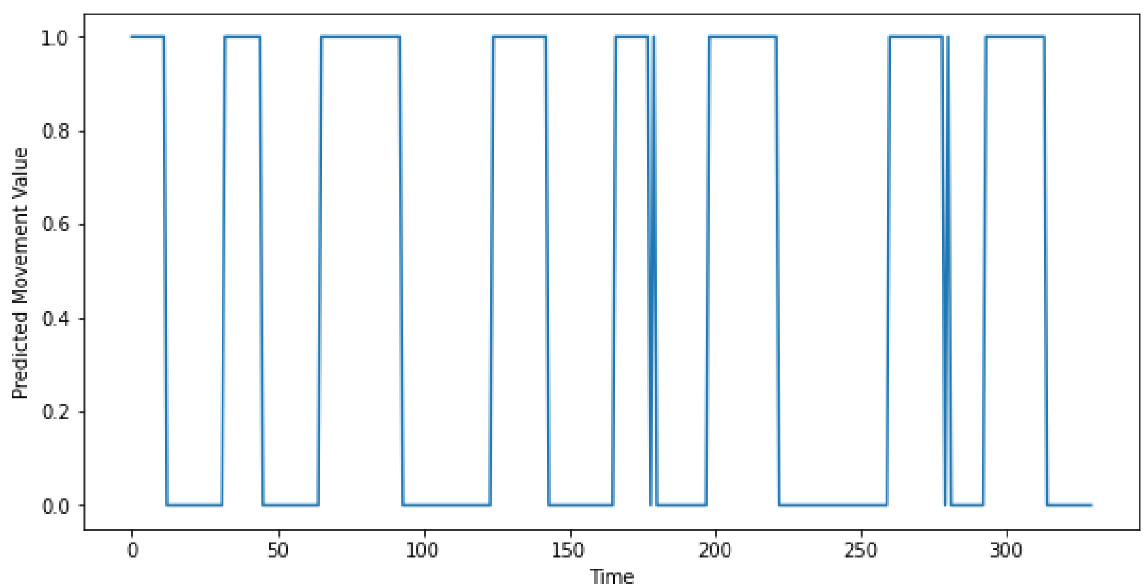

Figure 12. Example Output Prediction Set from RNN

3.3.4 Collecting Time Data for Analysis

After running the RNN algorithm and obtaining a prediction set, the associated timestamp data is used to collect movement information. The difference between the first and last timestamp in every run of values is collected and marked as stationary or moving based on the predicted value of that run. From this information, the total time spent for the entire session is collected by adding all the collected instances of time together. The total amount of time spent moving, total number of movement bouts, and longest bout of movement are also able to be extrapolated from the collected data.

3.4 Experimental Study: Gathering Data from a Smart Watch

In addition to collecting movement data from an android phone mounted to a powered wheelchair, a study was also done on gathering movement data from a smart watch worn by a wheelchair user. Rather than collect data of the wheelchairs motion, this allows the collection of and individuals arm movements. If motion can be tracked this way, it would allow manual wheelchair users to track movement using the pushing motions involved in manual wheelchair movement. It would also allow both manual and powered wheelchair users to track movement for exercises to gather data about when exercises are performed, and how long each exercise is performed.

3.4.1 Android Studio

For this study, two applications were created. A smart watch application to collect movement data, and a partner app for the android phone the watch connects to the watch and saves the output locally. The smart watch app is built using the same structure as the application for the previous study. The SensorManager starts collecting sensor data when the user presses a button on the watch face and stops when the user presses the button again, the screens and buttons displayed to the user are shown in Figure 13.

Figure 13. Smart Watch Application Screens

The collected sensor information is sent to the phone application where it is saved locally so the data can be easily accessed by connected the phone to a computer and moving the files.

3.4.2 Differences in Accelerometer Data

For this study, the movement data was collected while an individual in a powered wheelchair performed various exercises designed to prevent the formation of pressure ulcers. The accelerometer data across each axis varied depending on the exercise and did not appear similar to the raw data collected from the previous study. However, after transforming the raw acceleration data into a set of jerk data, the differences in the data were not as visible. The periods of movement varied but there were still clear patterns where motion was detected and where the device was stationary.

3.4.3 Analysis of Data with the RNN

The way that this data is interpreted based on the RNN algorithms prediction will be different but the RNN from the previous study was still able to analyze this data set. For this study, periods where the device is stationary indicate that an exercise is being performed, whereas periods of movement indicate the transition from one exercise to another. The users performing the exercises were also required to start and stop recording data by pressing a button on the watch which resulted in movement at the beginning and end of the data set that was unrelated to the exercises.

Chapter IV

Results and Discussion

4.1 Results from Experimental Studies

4.1.1 Accuracy of the RNN on Powered Wheelchair Movement

The study attempted to train the data set with five different sets of movement data collected from a smart phone that was in a phone holder mounted to the arm of the chair. Each time the RNN was trained with a new data set it was tested against the other data sets and the accuracy of the RNN was recorded. The lowest accuracy determined during this study was 96% and the highest accuracy recorded was 99%.

File ID	Number of data items in file	RNN Accuracy
1	4158	99%
2	4378	97%
3	4074	98%
4	4433	97%
5	6656	96%

Table 1. Accuracy obtained on tested data sets

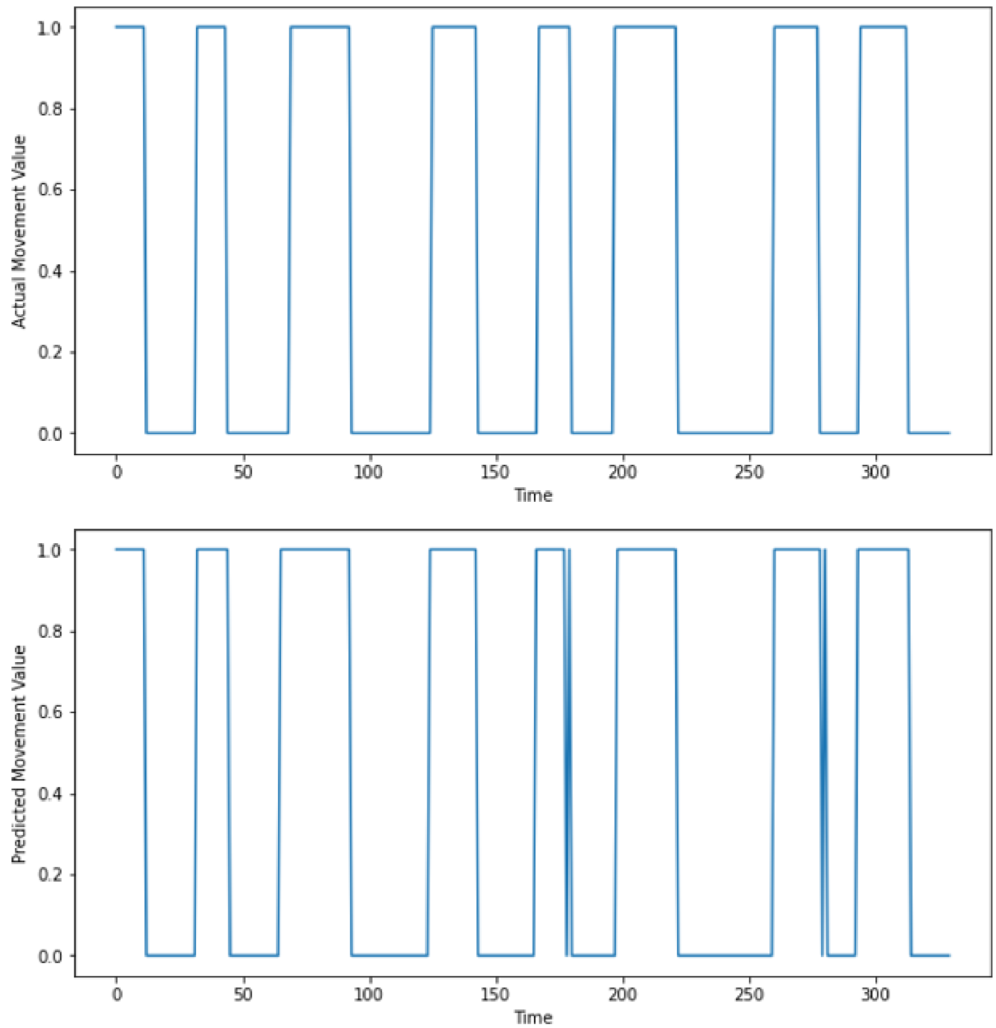

Figure 14. Example of actual movement data versus predicted movement data

As shown in Figure 14, the area where the RNN failed to accurately predict the movement value most often occurred at the beginning or end of a bout of movement. This is explained by the accelerometer data being close to or at the threshold value at the beginning and end of a movement, as the accelerations begin to increase or decrease. The loss and validation loss were also recorded and graphed for each test set.

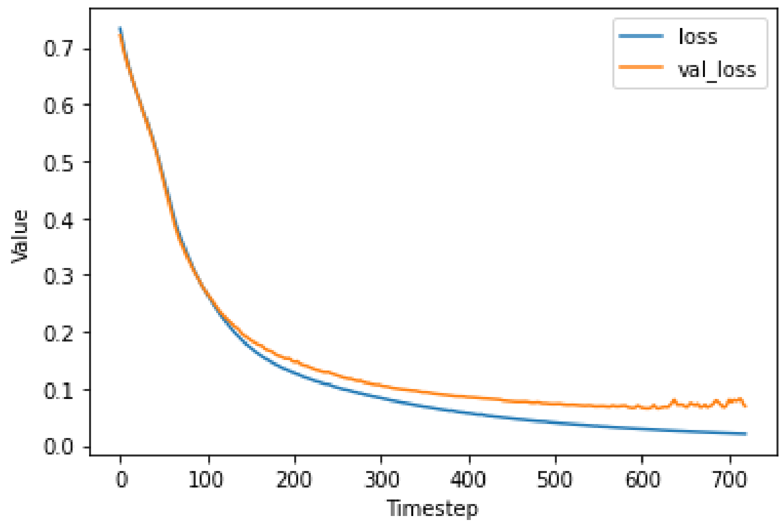

Figure 15. Example of loss and validation loss from RNN

The loss value is associated with the loss in the test data and the validation loss value is associated with the loss in the validation set. The validation loss value is used to trigger an early stop in the RNN when it no longer decreases, resulting in the final loss value.

4.1.2 Accuracy of the RNN on Smart Watch Movement

The accuracy of the RNN when used to analyze the data from the smart watch was less precise, with an average accuracy of 87%.

File ID	Number of data items in file	RNN Accuracy
1	2559	85%
2	2859	84%
3	1469	90%
4	2365	90%
5	2070	89%

Table 1. Accuracy obtained on tested data sets

The movements for the smart watch were more difficult to accurately model, as during a bout of exercise, and individual is likely to experience some level of movement in the hands and wrist due to the strain of the exercise. There was also some movement captured at the beginning and end of each sensor collection session sue to the need for the user to press the start and stop button on the watch. This was not an issue for the phone as it was mounted but pressing the button on the watch usually resulted in some amount of movement being captured. While the results from the watch are less accurate, they are still useful, as the errors would not greatly affect the amount of exercise data recorded and would only affect the determination for the start and end of each exercise.

4.2 Analysis of Results

The experimental studies proved that it is possible for a sufficiently trained AI to analyze the movements of wheelchair users. The accuracy of the results obtained from the experiment is comparable to the accuracy of extra equipment attached to the wheelchairs, without the need for additional cost and installation. The information can be used by an individual to assess their own lifestyle or be sent to a healthcare professional to help assess the individual's overall health.

While the accuracy of the smart watch application study was not as high as the smartphone application study, there was still useful information gathered from the results. These results could be used to ensure that an individual is performing their exercises for the appropriate amount of time, and in conjunction with the first application create an even better assessment of overall health and activity.

5 Conclusion and Future Research Direction

In this study, we presented an approach aiming to quantitatively assess a wheelchair user's mobility. Our approach only uses a wheelchair user's own smartphone/smartwatch to collect wheelchair maneuvering data, thus excluding the cost of purchasing and maintaining additional data sensors. We have developed approaches for sensor data processing and analysis. Particularly, we have developed a recurrent neural network (ANN), which could accurately classify wheelchair maneuvering data and provide meaningful analysis results on a wheelchair user's mobility.

In the next step of research, we will extend our research scope to manual wheelchair users, which account for the majority of the wheelchair population.

4.3.1 Application of Current Research to Manual Wheelchair Users

The smart watch application that was created could be used with manual wheelchair users to collect movement data without needing to have the phone mounted to the chair in any way. The movement data could be generated from the movements of a wheelchair user pushing the wheels, since the transformation of accelerations into individual jerk values is less concerned with the type of movement than the presence of movement. There would also still be the physical movement of the chair to add motion, so that even if the users hands are stalled at the beginning or end of a wheel push, there would still be accelerations from the chair captured by the sensors.

4.3.1 Continuation of Smart Watch Application

The smart watch application could be extended to help ensure individuals correctly perform their exercises. A new RNN could be trained to recognize the distinct movements involved in each exercise to better analyze which individual exercises are performed and for how long. Each exercise is associated with acceleration in a different direction and with a different range of movements. If this acceleration data is captured without being converted to jerk data and an RNN is trained for each exercise it would be possible to correctly predict which exercise if being performed given a set of accelerations.

References

AHRQ. (2012). Pressure Ulcer Risk Assessment and Prevention: A Comparative Effectiveness Review, 1–18.

AndroidDev. (2020). SensorEvent: Android Developers. https://developer.android.com/reference/android/hardware/SensorEvent.

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks, 5(2), 157–166. https://doi.org/10.1109/72.279181

Conger, S. A., & Bassett, D. R. (2011). A Compendium of Energy Costs of Physical Activities for Individuals Who use Manual Wheelchairs. Adapted Physical Activity Quarterly, 28(4), 310–325. https://doi.org/10.1123/apaq.28.4.310

Froehlich-Grobe, K., & Lollar, D. (2011). Obesity and Disability. American Journal of Preventive Medicine, 41(5), 541–545. https://doi.org/10.1016/j.amepre.2011.07.015

Fu, J., Hao, W., White, T., Yan, Y., Jones, M., & Jan, Y.-K. (2013). Capturing and Analyzing Wheelchair Maneuvering Patterns with Mobile Cloud Computing. In 35th Annual International Conference of the IEEE EMBS, 3 - 7 July, 2013 (pp. 2419–2422). Osaka; IEEE.

Fu, J., Li, F., Ong, M., Cook, T., Qian, G., & Zhao, Y. D. (2018). A Novel Approach for Assessing Power Wheelchair Users' Mobility by Using Curve Fitting. Lecture Notes in Computer Science Digital Human Modeling. Applications in Health, Safety, Ergonomics, and Risk Management, 158–168. https://doi.org/10.1007/978-3-319-91397-1_14

health.gov. (2014). Disability and Health. https://www.healthypeople.gov/2020/topics-objectives/topic/disability-and-health?topicid=9.

Hiremath, S. V., Intille, S. S., Kelleher, A., Cooper, R. A., & Ding, D. (2015). Detection of physical activities using a physical activity monitor system for wheelchair users. Medical Engineering & Physics, 37(1), 68–76. https://doi.org/10.1016/j.medengphy.2014.10.009

James, W. D., Elston, D. M., Treat, J., Rosenbach, M., & Neuhaus, I. M. (2020). Andrews' diseases of the skin: clinical dermatology. Elsevier.

Hartigan, John A., (1975). Clustering Algorithms (99th. ed.). John Wiley & Sons, Inc., USA.

Hastie T., Tibshirani R., Friedman J. (2009) Overview of Supervised Learning. In: The Elements of Statistical Learning. Springer Series in Statistics. Springer, New York, NY. https://doi.org/10.1007/978-0-387-84858-7_2

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84–90. https://doi.org/10.1145/3065386

Mathworks. (2014). Accelerometer. https://www.mathworks.com/help/supportpkg/android/ref/accelerometer.html.

Mitchell, T. M. (1997). Machine Learning. McGraw-Hill.

Mündermann, L., Corazza, S., & Andriacchi, T. P. (2006). The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications. Journal of NeuroEngineering and Rehabilitation, 3(1), 6. https://doi.org/10.1186/1743-0003-3-6

Nooijen, C. F. J., Stam, H. J., Bergen, M. P., Bongers-Janssen, H. M. H., Valent, L., Langeveld, S. van, … Berg-Emons, R. J. G. van den. (2015, December 11). A behavioural intervention increases physical activity in people with subacute spinal cord injury: a randomised trial. https://www.sciencedirect.com/science/article/pii/S1836955315001290.

Olah, C. (2015, August 37). Understanding LSTM Networks. Understanding LSTM Networks -- colah's blog. http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

Saxena, S. (2017, October 27). Artificial Neuron Networks (Basics): Introduction to Neural Networks. Medium. https://becominghuman.ai/artificial-neuron-networks-basics-introduction-to-neural-networks-3082f1dcca8c.

Seguin, R., Buchner, D. M., Liu, J., Allison, M., Manini, T., Wang, C.-Y., … Lacroix, A. Z. (2014). Sedentary Behavior and Mortality in Older Women. American Journal of Preventive Medicine, 46(2), 122–135. https://doi.org/10.1016/j.amepre.2013.10.021

Shuger, S. L., Barry, V. W., Sui, X., Mcclain, A., Hand, G. A., Wilcox, S., … Blair, S. N. (2011). Electronic feedback in a diet- and physical activity-based lifestyle intervention for weight loss: a randomized controlled trial. International Journal of Behavioral Nutrition and Physical Activity, 8(1), 41. https://doi.org/10.1186/1479-5868-8-41

Sutton, R. S., Bach, F., & Barto, A. G. (2018). Reinforcement Learning: An Introduction (2nd ed.). MIT Press Ltd.

van Diggelen, F., & Enge, P. (2015). The World's first GPS MOOC and Worldwide Laboratory using Smartphones. Proceedings of the 28th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2015), 361–369.

Viswarupan, N. (2017, October 12). K-Means Data Clustering. Medium. https://towardsdatascience.com/k-means-data-clustering-bce3335d2203.

Warms, C. A., & Belza, B. L. (2004). Actigraphy as a Measure of Physical Activity for Wheelchair Users With Spinal Cord Injury. Nursing Research, 53(2), 136–143. https://doi.org/10.1097/00006199-200403000-00010

Wikimedia Foundation. (2020, November 10). Long short-term memory. Wikipedia. https://en.wikipedia.org/wiki/Long_short-term_memory.

Wise, J. M., & Hongu, N. (2009). Pedometer, Accelerometer, and Mobile Technology for Promoting Physical Activity. The University of Arizona College of Agriculture and Life Sciences. https://extension.arizona.edu/sites/extension.arizona.edu/files/pubs/az1491-2014.pdf.

Wolpert, D. H., & Macready, W. G. (1995). No Free Lunch Theorems for Search. The Sante Fe Institute.

Woodford, C. (2020, June 17). How do pedometers work? Explain that Stuff. https://www.explainthatstuff.com/how-pedometers-work.html.

Woude, L. V. D., Veeger, H., Dallmeijer, A., Janssen, T., & Rozendaal, L. (2001). Biomechanics and physiology in active manual wheelchair propulsion. Medical Engineering & Physics, 23(10), 713–733. https://doi.org/10.1016/s1350-4533(01)00083-2