2021

AP® CollegeBoard

AP® Research Academic Paper

Sample Student Responses and Scoring Commentary

Inside:

Sample A

- ☑ Scoring Guideline
- ☑ Student Samples
- ☑ Scoring Commentary

The Response…				
Score of 1 **Report on Existing Knowledge**	**Score of 2** **Report on Existing Knowledge with Simplistic Use of a Research Method**	**Score of 3** **Ineffectual Argument for a New Understanding**	**Score of 4** **Well-Supported, Articulate Argument Conveying a New Understanding**	**Score of 5** **Rich Analysis of a New Understanding Addressing a Gap in the Research Base**
Presents an overly broad topic of inquiry.	Presents a topic of inquiry with narrowing scope or focus, that is NOT carried through either in the method or in the overall line of reasoning.	Carries the focus or scope of a topic of inquiry through the method **AND** overall line of reasoning, even though the focus or scope might still be narrowing.	Focuses a topic of inquiry with clear and narrow parameters, which are addressed through the method and the conclusion.	Focuses a topic of inquiry with clear and narrow parameters, which are addressed through the method and the conclusion.
Situates a topic of inquiry within a single perspective derived from scholarly works **OR** through a variety of perspectives derived from mostly non-scholarly works.	Situates a topic of inquiry within a single perspective derived from scholarly works **OR** through a variety of perspectives derived from mostly non-scholarly works.	Situates a topic of inquiry within relevant scholarly works of varying perspectives, although connections to some works may be unclear.	Explicitly connects a topic of inquiry to relevant scholarly works of varying perspectives **AND** logically explains how the topic of inquiry addresses a gap.	Explicitly connects a topic of inquiry to relevant scholarly works of varying perspectives **AND** logically explains how the topic of inquiry addresses a gap.
Describes a search and report process.	Describes a nonreplicable research method **OR** provides an oversimplified description of a method, with questionable alignment to the purpose of the inquiry.	Describes a reasonably replicable research method, with questionable alignment to the purpose of the inquiry.	Logically defends the alignment of a detailed, replicable research method to the purpose of the inquiry.	Logically defends the alignment of a detailed, replicable research method to the purpose of the inquiry.
Summarizes or reports existing knowledge in the field of understanding pertaining to the topic of inquiry.	Summarizes or reports existing knowledge in the field of understanding pertaining to the topic of inquiry.	Conveys a new understanding or conclusion, with an underdeveloped line of reasoning **OR** insufficient evidence.	Supports a new understanding or conclusion through a logically organized line of reasoning **AND** sufficient evidence. The limitations and/or implications, if present, of the new understanding or conclusion are oversimplified.	Justifies a new understanding or conclusion through a logical progression of inquiry choices, sufficient evidence, explanation of the limitations of the conclusion, and an explanation of the implications to the community of practice.
Generally communicates the student's ideas, although errors in grammar, discipline-specific style, and organization distract or confuse the reader.	Generally communicates the student's ideas, although errors in grammar, discipline-specific style, and organization distract or confuse the reader.	Competently communicates the student's ideas, although there may be some errors in grammar, discipline-specific style, and organization.	Competently communicates the student's ideas, although there may be some errors in grammar, discipline-specific style, and organization.	Enhances the communication of the student's ideas through organization, use of design elements, conventions of grammar, style, mechanics, and word precision, with few to no errors.
Cites **AND/OR** attributes sources (in bibliography/ works cited and/or in-text), with multiple errors and/or an inconsistent use of a discipline-specific style.	Cites **AND/OR** attributes sources (in bibliography/ works cited and/or in-text), with multiple errors and/or an inconsistent use of a discipline-specific style.	Cites **AND** attributes sources, using a discipline-specific style (in both bibliography/works cited **AND** in-text), with few errors or inconsistencies.	Cites **AND** attributes sources, with a consistent use of an appropriate discipline-specific style (in both bibliography/works cited **AND** in-text), with few to no errors.	Cites **AND** attributes sources, with a consistent use of an appropriate discipline-specific style (in both bibliography/works cited **AND** in-text), with few to no errors.

Academic Paper

Overview

This performance task was intended to assess students' ability to conduct scholarly and responsible research and articulate an evidence-based argument that clearly communicates the conclusion, solution, or answer to their stated research question. More specifically, this performance task was intended to assess students' ability to:

- Generate a focused research question that is situated within or connected to a larger scholarly context or community;
- Explore relationships between and among multiple works representing multiple perspectives within the scholarly literature related to the topic of inquiry;
  Articulate what approach, method, or process they have chosen to use to address their research question, why they have chosen that approach to answering their question, and how they employed it;
- Develop and present their own argument, conclusion, or new understanding while acknowledging its limitations and discussing implications;
- Support their conclusion through the compilation, use, and synthesis of relevant and significant evidence generated by their research;
- Use organizational and design elements to effectively convey the paper's message;
- Consistently and accurately cite, attribute, and integrate the knowledge and work of others, while distinguishing between the student's voice and that of others;
- Generate a paper in which word choice and syntax enhance communication by adhering to established conventions of grammar, usage, and mechanics.

The Classification of EMG Signals using Machine Learning for the Construction of a Silent Speech Interface

AP Research
17 January 2020

Discipline: Biomedical Engineering

Research Question:
Which type of Machine Learning Algorithm (Convolutional Neural Networks or Pattern Recognition) is most accurate at classifying surface ElectroMyoGraph (sEMG) signals from the submental triangle (area under the chin) to develop a Silent Speech Interface?

Approach: To Explain and Create
Design: Experimental
Method: Quantitative True Experimental & Engineering Method
Data Source: Primary

Method Design Statement:
This study uses a quantitative experimental research design to explain which Machine Learning model produces the best accuracy for the classification/interpretation of surface Electromyograph signals for use in a Silent Speech Interface.

Keywords:
Silent Speech Interface, sEMG, Machine Learning, Convolutional Neural Network, Pattern Recognition

Word Count: 4998

Documentation Style: APA

Abstract

With 7.5 million people suffering from speech impediments, it is imperative that accurate speech aids are developed. Conditions such as stroke, ALS, and cerebral palsy leave their patients unable to speak and force them to use cumbersome and inefficient devices such as eye/cheek trackers. In this study, a speech aid known as a Silent Speech Interface (SSI) was created. This device could be used by patients with speech disorders to communicate letters in the English alphabet voicelessly, merely by articulating words or sentences in the mouth without producing any sounds. This device captures and records the subtle neurological activation of the muscles in the internal speech system from the surface of the skin. In simpler terms, the SSI records electrical EMG signals from the speech system. These EMG signals are then classified into speech in real-time using a trained Machine Learning model. This device could effectively determine what was communicated with 80.1% accuracy. Through the usage of this device, it was found that the SVM algorithm was the most effective ML model for the classification of EMG signals from the throat. These findings fill the lack of research on optimal ML models for use in an SSI. Overall, this study involves the creation of a device that measures biomedical signals and translates them into speech using the SVM model with high accuracy. This study's findings could improve the accuracy of future SSIs by showing which algorithms are most accurate for use in an SSI.

Introduction

Multiple Sclerosis (MS) is a progressive neurodegenerative disease characterized by lesions in the nervous system that affects nearly 2.3 million people worldwide. As the disease progresses, MS creates communication problems between the brain and body. Two major impairments that come with MS are speech disorders known as dysphonia and dysarthria. These speech disorders are common, affecting about 50% of MS patients (Brown, 2000). Dysphonia affects speech muscles, which can lead to patients being inaudible (Beukelman and Garrett, 1988). Other diseases such as Motor Neuron Diseases (MNDs) cause patients' speech to become unclear, taking away a patients' ability to speak. MND patients are forced to use eye/cheek tracking speech aids which make the user perform specific muscle movements to select letters/words the user wants to communicate. These trivial/cumbersome devices prove to be an extremely slow and fatiguing solution for communication. In this study, these systems which use eye/cheek tracking to develop a Speech Interface will be referred to as Conventional Speech Interfaces (CSI). Although CSI technology allows patients to communicate, it is far from optimal due to the slow rate of communication and high inaccuracy (Kapur, 2019). Newer speech aids use technology that doesn't involve traditional eye/cheek tracking.

I. Silent Speech Interfaces and the Electromyograph (EMG) Signal

Silent speech refers to the act of minimally or internally articulating words without producing sounds (Kapur, 2019). Producing silent speech is less fatiguing than regular speech or using CSIs. Although silent speech is inaudible, it produces signals that can be recorded and classified into words using Machine Learning. These signals are ElectroMyoGraph (EMG) signals which are created by subtle muscle contractions. During silent speech, speech muscles (cheek, lips, etc.) contract, producing EMG signals in certain patterns. When the same words are spoken, the same muscle contractions occur to produce specific EMG signal patterns. Thus, if the EMG signals can be recorded, it is possible to translate the signal patterns to determine the speech that was silently spoken. This allows for the development of a new type of speech interface known as Silent Speech Interfaces (SSI).

SSIs are a more effective speech aid compared to CSIs. However, an SSI's accuracy is highly dependent on the computer algorithms that are used to translate the EMG signals into speech (Kapur, 2019).

The most common method to record EMG signals involves placing electrodes on the skin to detect muscle contractions (Kapur. 2019). EMG signals recorded in this manner are also known as surface electromyograph (sEMG) signals as they are recorded from the skin's surface. EMG signals are recorded in this manner due to its non-invasive nature and easy implementation (Kapur, 2019).

II. Artificial Intelligence for Construction of Automated Silent Speech Interface

SSIs make use of Machine Learning to identify sEMG/EMG patterns to translate silent speech into language (Denby, 2011). Machine Learning (ML) is a subset of Artificial Intelligence (AI) and is defined as "the field of study that gives computers the ability to learn without being explicitly programmed" as said by Dr. Arthur Samuel, who originally coined the term (1959).

ML is a broad field and can be subdivided into 2 main categories: supervised learning and unsupervised learning. The construction of an SSI requires supervised learning, in which, as Dr. Ayodele of the University of Portsmouth in the UK states:

> The algorithm generates a function that maps inputs to desired outputs. One

> standard formulation of the supervised learning task is the classification problem: the learner is required to learn (to approximate the behavior of) a function which maps a vector into one of several classes by looking at several input-output examples of the function. (Ayodele, 2010)

Using supervised learning, it is possible to “teach” and develop an ML algorithm that can translate sEMG/EMG signals into the letters/words that were silently spoken.

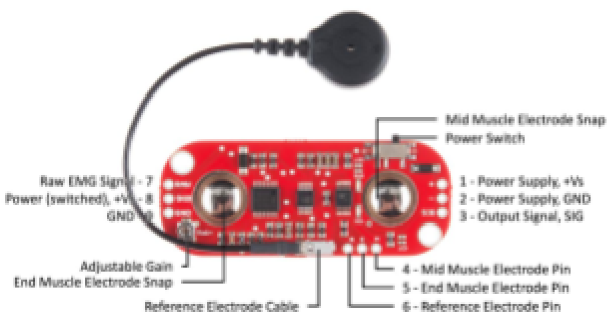
In this study, sEMG signals will be translated into one of the five vowels. Thus ML classification algorithms were used to classify sEMG signals into the letters/words that were silently spoken.

In the case of EMG signal classification, only several supervised ML algorithms could be used for the construction of an SSI, because only a few ML algorithms are capable of analyzing and processing signals (series of numbers). This narrowed the ML models that could be used and set the scope of this study (Further explained in *V. Types of Algorithms*). Access to ML Development Software has allowed several researchers to classify EMG signals. This facilitated the past development of SSIs.

III. Previous Findings

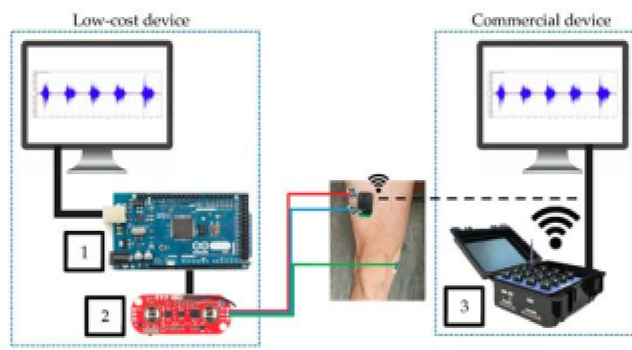
EMGs are typically recorded through an electromyograph. These high-end machines are large/expensive and thus an inconvenient solution to monitor EMG signals for the development of an SSI. A reliable alternative to the electromyograph is the Myoware muscle sensor shown in Figure 2 (Char, 2018). A study conducted by Kareem et al. used the Myoware to record EMG signals (2017). By comparing the sEMG signals recorded by the Myoware sensor to those recorded by the Electromyograph, Kareem et al. found that the Myoware can be used in ML applications due to its high accuracy (2017). Furthermore, it was determined that sEMG signals recorded from the Myoware and the electromyograph have the same patterns (Figure 3). Therefore Kareem et al. identified that classification of sEMG signals is possible using the Myoware due to its high accuracy. This study was important as it justified the use of a Myoware sensor (used in this study) as an alternative to the electromyograph used in other studies.

Figure 2: MyoWare Muscle Sensor



This figure shows the Myoware muscle sensor which was used to record EMG signals

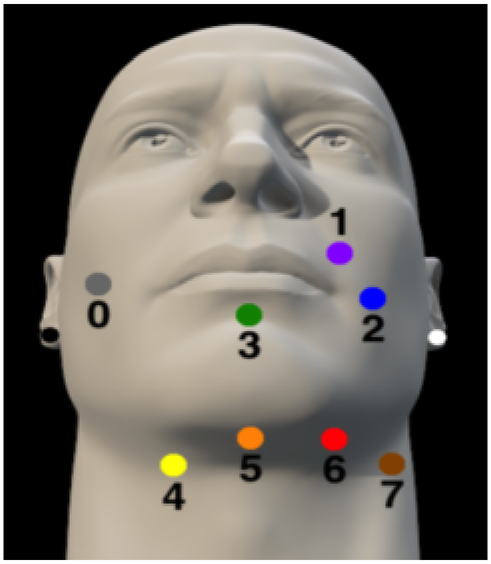
Figure 3: EMG Data Similarity



This figure shows that EMG signals collected from the Myoware and electromyograph are nearly identical

A landmark paper, titled *"Non-Invasive Silent Speech Recognition in Multiple Sclerosis with Dysphonia,"* by Kapur et al. is one of the most advanced research on the implementation of an SSI. The SSI created, recorded EMG signals from a multitude of locations from the face/throat as shown in Figure 4. These signals were used to train, validate, and test the Convolutional Neural Network (CNN) -ML algorithm- that was used to build the SSI. Although the use of the model was never justified, the CNN model yielded high accuracy of 79%. The SSI developed improved the speed/accuracy of communication compared to CSIs (Kapur, 2019). This study was crucial as it laid the foundation for the methodology in this study as it was the only study that identified steps to create an SSI.

Figure 4: Picture of Electrode Locations



This figure shows an image of electrode placements used in Kapur's study

Another study by Shultz et al. developed an SSI using the EMG-PIT corpus, a database of EMG recordings from the speech system. Using a Gaussian Mixture Model (GMM), the developed SSI managed to achieve a low error rate of 10%. Similar to Kapur et al.'s study, Shultz's study lacks justification for the ML algorithm used. This lack of justification could suggest that other ML algorithms would perform better at classifying sEMG signals in an SSI.

IV. Introduction to the Research Gap

"Machine Learning Algorithms for Characterization of EMG Signals" by Karlik was fundamental to understanding the gap in the field of knowledge. This study compared several ML algorithms to classify EMG signals for use in arm prostheses. Various ML algorithms -Fuzzy Systems, Probabilistic, and Swarm intelligence- were all used to classify EMG signals. After applying ML to the EMG data, classification accuracies were compared to other studies

conducted. Karlik used a CNN model to achieve a 98% classification accuracy. Using these results, Karlik identifies that CNN algorithms are the most accurate ML algorithm for classifying EMG data for arm prosthesis.

Despite previous research involving EMG signals and SSIs, no study has identified the most effective ML algorithm to classify sEMG signals for use in an SSI. sEMG signals collected from the speech muscles have different/subtle patterns due to the weak signals produced and the variations in regular speech. Because of this distinction between EMG signals produced from the speech system and the human arm, there still is a lack of understanding of the optimal ML model to use for sEMG translation/classification in an SSI.

Furthermore, other researchers who have developed SSIs, such as Kapur and Shultz, don't compare various ML models or provide justification as to why a certain ML model was used (Stated in further detail previously: *III. Previous Findings*). This further establishes the gap in research that the optimal ML model for use in an SSI hasn't been identified.

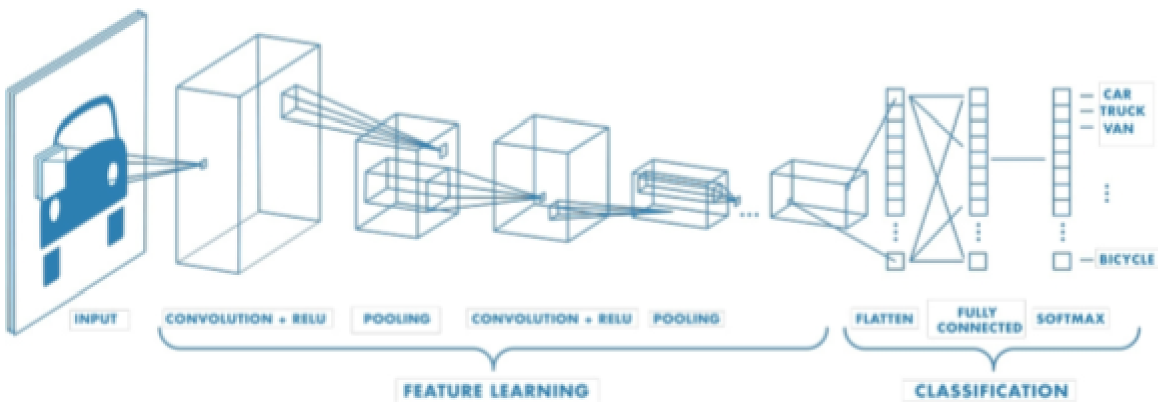
V. Types of Algorithms

Throughout the literature on EMG classification, various ML algorithms have been used (Char, 2018). Kapur's study involved developing a Convolutional Neural Network (2018) to classify EMG signals whereas in Shultz's study a form of Pattern Recognition was used (2009). In the field of electroencephalography(EEG) signal classification, CNNs and pattern recognition algorithms have also been effectively used to classify these biomedical signals (Zia ur Rehman et al., 2018). Due to the common use of these ML Algorithms to classify biomedical signals, these algorithms were explored in this study.

A. Convolutional Neural Network

The Convolutional Neural Network (CNN) gained popularity as it was an effective method to recognize objects in images. As this method gained traction in the ML field, it has been optimized by numerous researchers allowing for the development of effective CNNs to classify images (Alaskar, 2018).

Figure 5: Convolutional Neural Network



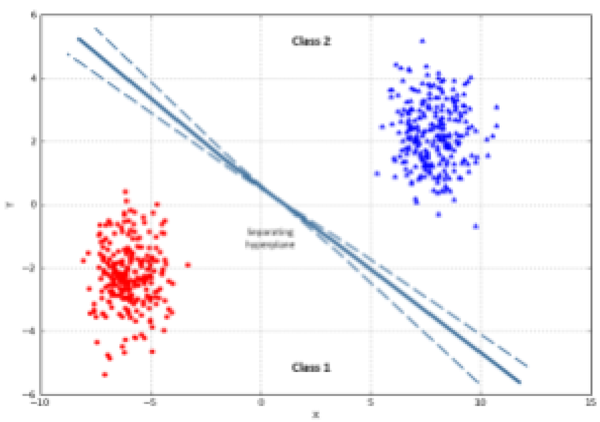
This figure shows the different layers of CNN and highlights the Feature Learning and Classification layers (Alaskar, 2018)

As shown in Figure 5, CNNs function by processing an input image with a series of filters known as the feature learning layers (MathWorks, *Googlenet* n.d.) which allow the ML model to identify specific “features” of images. Once the algorithm identifies image features, the classification layers match the input images with correct output, in this case being predicted speech (letters/words).

When developing a CNN it is possible to reuse CNN models built by previous researchers and repurpose them for the applications of a new study. This process of repurposing previously developed CNNs is known as transfer learning and involves keeping the same feature learning layers of an existing CNN and replacing just the classification layers for a specific application (Bonaccorso, 2017). Computer scientists have developed numerous CNN algorithms such as AlexNet (Krizhevsky et al., 2017) and VGGNet-16 (Muhammad et al., 2018) for various applications. However, studies show that GoogleNet (Tang et al., 2017) is the most accurate CNN for image recognition (Mohanty et al., 2016). Thus GoogleNet was chosen to be implemented in this study.

B. Pattern Recognition

Figure 6: Pattern Recognition Algorithms



This figure shows the division of classes (red/blue points) using pattern recognition

Pattern recognition (PR) is standardly used for biomedical signal classification (Bishop et al., 1970) and was thus deployed in this study which aims to compare PR against the CNN (Research Goals: *Section VI. Research Goals & Research Gap*). PR is known for its simple construction and deployment. PR algorithms attempt to develop a division between multiple classes which are represented in red/blue in Figure 6. PR algorithms can determine which class is associated with an EMG signal by plotting a point based on the given inputs and identifying the point's location relative to the division.

There are multiple ways to create a division using PR. In the scope of this study, 7 PR Algorithms were developed/deployed as only 7 PR models could classify EMG signals. These PR Algorithms were used to classify sEMG signals into speech.

VI. Research Goals & Research Gap

SSIs are superior to other vocal aids as shown by Kapur's research where it's found that SSIs enable accurate communication (2019). In Kapur's research, a CNN was used to classify sEMG signals in an SSI (2019). Although this SSI was accurate at EMG to speech translation, no reasoning was provided to justify the use of a CNN. Similarly, in a study conducted by Schultz et al., a Gaussian Mixture (GM) model is used to classify sEMG signals into speech. Although the GM model was accurate, this study also fails to justify the use of GM Models for the classification of sEMG signals.

Only Karlik's study generated/compared algorithms to determine the most accurate ML algorithm for EMG classification in arm prosthesis. This study identifies the CNN as the most accurate to classify EMG signals for arm prosthesis, however, EMG signals recorded from speech muscles will have different structures/patterns (Eremenko et al., n.d.). Therefore Karlik's study cannot truly identify the most accurate ML algorithm to classify EMG signals into speech. Thus there is no conclusive study to identify the best ML algorithm to classify sEMG signals into speech. This ultimately results in a gap of knowledge in the field of sEMG classification and SSI development. This study aimed to identify the most accurate ML algorithm for use in an SSI.

Overall, this research study attempts to fill this gap by comparing two different types of Machine Learning Classification Algorithms (Convolutional Neural Networks and Pattern

Recognition) to identify the most accurate algorithm for use in an SSI. Therefore, this project aims to construct two types of ML algorithms in order to gain insight into the question:

"Which type of Machine Learning Algorithm (Convolutional Neural Networks or Pattern Recognition) is most accurate at classifying surface ElectroMyoGraph (sEMG) signals from the submental triangle (area under the chin) to develop a Silent Speech Interface?" This study would help further improve the ability of SSIs to translate sEMG signals into speech, allowing for more accurate communication.

Engineering Goal: To Develop a Speech Interface using a Muscle Sensor that can both collect and classify sEMG signals from the submental triangle (area under the chin) with greater than 80% accuracy.

The engineering goal of achieving an 80% accuracy was developed as other studies on SSIs also strived to acquire an 80% accuracy. The engineering goal also involved creating an SSI using a low-cost muscle sensor (Myoware) as previous studies only used electromyographs (Karlik, 2018).

Methodology

To achieve the research/engineering goal an SSI that translates sEMG signals into speech using ML needed to be developed. Additionally, to develop the ML algorithms, an EMG dataset to train/test the ML algorithms had to be created. This dataset was created through the use of a developed Arduino-based EMG recorder. Once the EMG recorder is connected to a laptop running the ML models for EMG classification, the device will function as an SSI which can both record and translate sEMG signals generated from silent speech. To translate EMG recordings to speech, multiple ML algorithms were constructed.

To compare these various methods of ML to answer the research question, a true quantitative experimental method was developed to evaluate the performance of different types of ML algorithms. Additionally, an engineering method was developed to evaluate the SSI created. This true quantitative experimental and engineering method both involved using the classification accuracies and F1 scores. Classification accuracy is a measure of how often the model is correct whereas F1 scores provide a more holistic view of the model taking both accuracy and precision into account (Eremenko et al., n.d.). These two parameters were used to answer the research question and determine if the engineering goal was met.

To identify the most accurate model through the true quantitative experimental method, the tested models' F1 scores were compared just as in Karlik's study. F1 scores range from 0 to 1, and high scores indicate that a model is both accurate and precise (Eremenko et al., n.d.). The models' F1 scores were compared, and the model with the highest score was identified as the best performing ML algorithm, answering the research question.

To determine if the created SSI met the engineering goal (80% accuracy), only the accuracy of the ML model with the highest F1 score was considered. This is because the SSI will use the best performing model (model with highest F1 score - identified in experimental method) to translate EMG signals. Thus the accuracy of the SSI is equal to the accuracy of the algorithm used in the SSI.

This method of comparing F1 scores is a common way of evaluating ML algorithms and was used in studies such as Karlik, who compared various ML algorithms for EMG arm prosthesis classification (n.d.). This study's results are valid as the use of standard algorithm evaluation parameters creates a standard method of comparison between tested algorithms. By using this method of analysis, it is possible to accurately identify which ML algorithm is best suited to translate EMG signals and if the SSI met the engineering goal (80% accuracy).

To evaluate models, the SSI had to be created. Creating an SSI involved the following procedures which were also carried out in Kapur's study:

1) Arduino-Based EMG Recorder Development - Creating a device to record EMG/sEMG signals
2) Database Preparation - Creating a dataset for Machine Learning using the created EMG recorder
3) PR Methods - Developing PR Algorithms to classify signals
4) CNN Methods - Developing a CNN to classify signals
5) EMG-based Silent Speech Interface Methods - Create an SSI

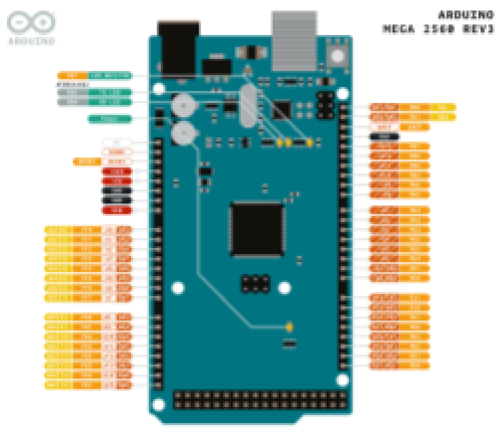
The above procedures are discussed in further detail below.

I. Arduino-Based EMG Recorder Development

The first step in the development of the EMG recorder was wiring components together. The EMG recorder would make use of a microcontroller (Arduino) that takes EMG recordings from a sensor and saves data to an SD card. The recorder has buttons to start recording EMG signals.

The Arduino - a small computer that can receive inputs from many sensors (Arduino, n.d.) - served to record EMG signals using the Myoware sensor. The Arduino Mega (Figure 7), was used due to its high sampling rate (Hartman, n.d.) which is crucial for ML applications as detailed EMG data can be collected (Eremenko et al., n.d.).

Figure 7: Arduino Mega Diagram

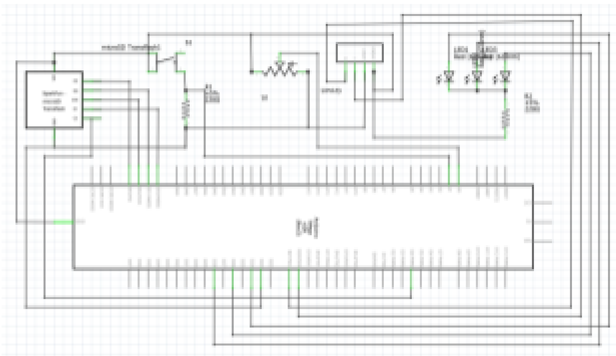


This figure shows a diagram of the Arduino Mega that was used to create the EMG recorder

For the Arduino to record EMG signals, the Myoware muscle sensor (Figure 2) was used to detect EMG signals. As discussed previously, the MyoWare is the best commercially available muscle sensor and has been used in other studies due to its reliability/accuracy (Hartman, n.d.). Although Kapur doesn't use the Myoware, Kareem justifies the use of this sensor as it produces accurate results (n.d.).

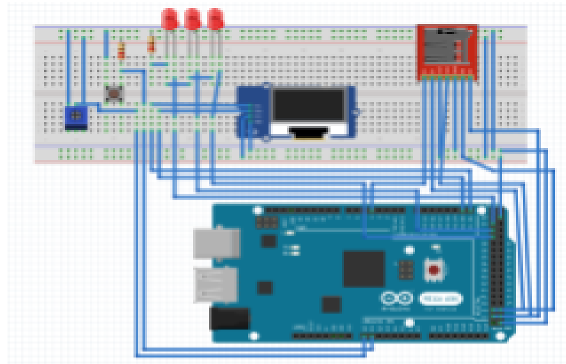
The Arduino Mega, Myoware, and button were all connected as shown in the schematic (Figure 8) and diagram (Figure 9) below. To ensure functionality, the device was connected this way according to data sheets provided by the Arduino company (n.d.).

Figure 8: Schematic of EMG Recorder



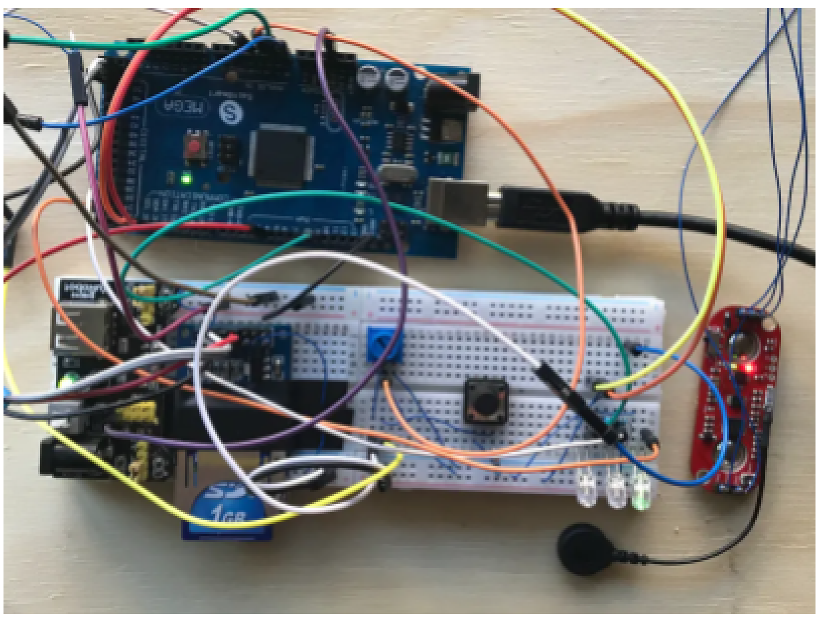
This figure shows the EMG recorder schematic

Figure 9: Diagram of EMG recorder



This figure shows the EMG recorder wiring diagram

Figure 10: Image of Developed EMG Recorder



This figure shows an image of the created EMG recorder with the red Myoware sensor to the right and the blue Arduino Mega at the top.

An image of the fully constructed EMG recorder is shown in Figure 10. This EMG recorder was programmed to perform various tasks. The device has 3 tasks:

1. Wait for the button to be pressed

2. Take user input on what letter is being silently spoken
3. Record and Save EMG values as quickly as possible

Tasks 1 and 2 (wait for the button to be pressed & take user input on what letter is being silently spoken) involve taking in user input. The code in Table I shows the commands executed to determine if the button has been pressed, whereas the code in Table II shows the commands executed to take in the input of what letter (A, E, I, O, U) is silently spoken. The code for the button allows the device to start recording EMG data only when the user is ready to speak. This was also done by Kapur and ensures that EMG data is only recorded when silent speech is produced. The code to determine what letter is being silently spoken is important as the device needs to associate each EMG recording with a specific letter.

TABLE I
CHECKING STATE OF BUTTON

```
void Button() {
  if (buttonState !=
!(digitalRead(buttonPin))) { // Button
has been pressed
    buttonState = !buttonState; //
Inverts signal
    displayEMG(); //Update Display
  }
}
```

TABLE II
RECORDING THE LETTER SPOKEN

```
void Potentiometer() {
  potVal = analogRead(potPin); // Reads
the "current" state of the Potentiometer
  if (abs(oldPotVal - potVal) >= 10) { //
If Potentiometer Value has changed
    oldPotVal = potVal; // Updating Old
Potentiometer Value
    index = map(potVal, 0, 1024, 0, 5);
// Mapping Values to index value
    Letter = letterList[index]; //
Updating Letter Variable

    displayEMG();
  }
}
```

To perform task 3 (record EMG data), the code shown in Table III is executed. This code both records the EMG signals and creates a dataset at the same time. The program does this simultaneously, just as Kapur's EMG recorder, as it optimizes the program's speed allowing many data points to be collected in a few seconds. The code in Table III is optimized to save 3000 comma-delimited EMG values in approximately 1.5 seconds.

TABLE III
RECORDING EMG DATA

```
void recordEMG() { // Recording EMG Optimized for speed
  for (int i = 0; i <= 3000; i++) {
     Data.print(String(analogRead(modEMGPin)) + ","); // print Raw EMG values
  }
}
```

After EMG data has been recorded, the EMG data has to be saved. This is done after recording data by using the "Data.close();" function which saves previously recorded EMG data on an SD card (Table IV).

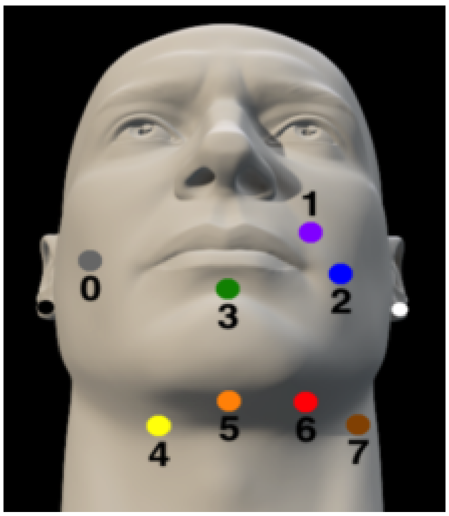
TABLE IV
SAVING DATA ON SD CARD

```
void loop() {
  if (buttonState == 1) { // If button is pressed
    Data = SD.open(Letter, FILE_WRITE); //Open SD card for Writing

    recordEMG(); // Void Loop for Recording EMG Signal

    Data.print(";");
    Data.close(); // Closing Data file
  }
```

II. Database Preparation

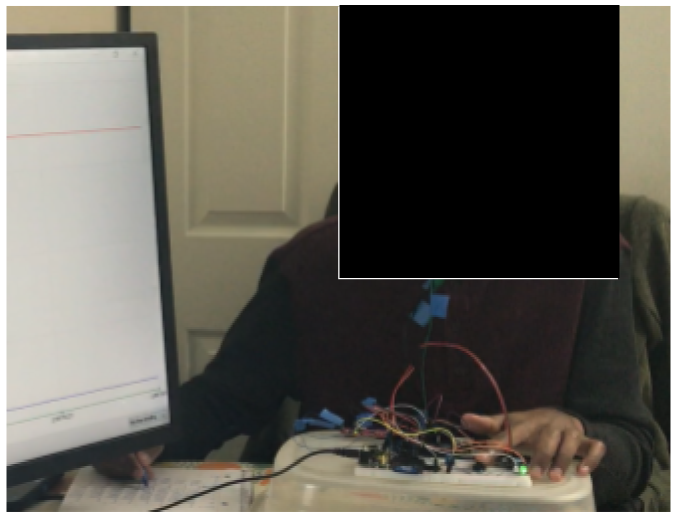
A database needed to be created to train the ML models. This data was created using the created EMG recorder device (Figure 10). Three electrodes were attached to the submental triangle, the area under the chin (Figure 11). A total of 1020 EMG recordings were taken, 170 for each of the 5 vowels and another 170 to establish a baseline of not speaking at all. Electrode placements were justified by Kapur's research where he identifies various areas on the throat to collect EMG signals (Figure 4). The muscle/area targeted in this study is marked with an orange dot that is labeled "5" in Figure 4. Kapur created a larger dataset, however, due to time constraints only 1020 signals were collected in this study.

Figure 4: Picture of Electrode Locations



This figure shows an image of electrode placements used in Kapur's study

Figure 11: Electrode Placements and Generating Data



This figure shows electrode placements used when collecting data. Electrodes are placed under the chin (submental triangle) which was also done in Kapur's study

Afterward, the EMG dataset was parsed into 2 sets: a training set and a testing set. 80% of the entire data set was stored in the training set, whereas 20% of the entire data set was stored in the testing set. The dataset was split in this manner to ensure that the magnitude of the training data was sufficient for the algorithm to maintain optimal accuracy (Mwebaze & Owomugisha, 2016).

III. Pattern Recognition Methods

Once imported into the MATLAB programming environment, the Classification Learner App - an ML tool used for developing ML algorithms (MathWorks, *Classification Learner App* n.d.) - was utilized to develop the Pattern Recognition Algorithms to classify sEMG signals (Eremenko et al., n.d.). The MATLAB Classification App allowed the easy implementation of different PR Algorithms. Only 7 types of Pattern Recognition algorithms were capable of translating EMG signals and they were all implemented: Support Vector Machine (SVM), Ensemble, K-Nearest Neighbors (K-NN), Decision Tree Classification, Naive Bayes (NB), Linear Discriminant, Quadratic Discriminant. As shown in Tables V and VI the only significant difference between the code for implementing an SVM model and KNN model lies in the function used to train the algorithm.

TABLE V
IMPORTING SVM CLASSIFIER

```
classificationSVM = fitcecoc(...
    predictors, ...
    response, ...
    'ClassNames', {'A'; 'E'; 'I'; 'O';
'U'; 'classification'});
```

TABLE VI
IMPORTING KNN CLASSIFIER

```
classificationKNN = fitcknn(...
    predictors, ...
    response, ...
    'ClassNames', {'A'; 'E'; 'I'; 'O';
'U'; 'classification'});
```

The EMG data and the corresponding vowel/letter were imported into the computer. Each algorithm was trained/tested on the same dataset to ensure the validity of the results. Due to time constraints and limited computing power, each algorithm was given only 10 iterations (opportunities) to learn from the data, ensuring that no ML model had an advantage over the other tested algorithms. After training each PR model, each algorithm was tested on the previously developed testing data to determine classification accuracy and F1 scores.

IV. Convolutional Neural Network Methods

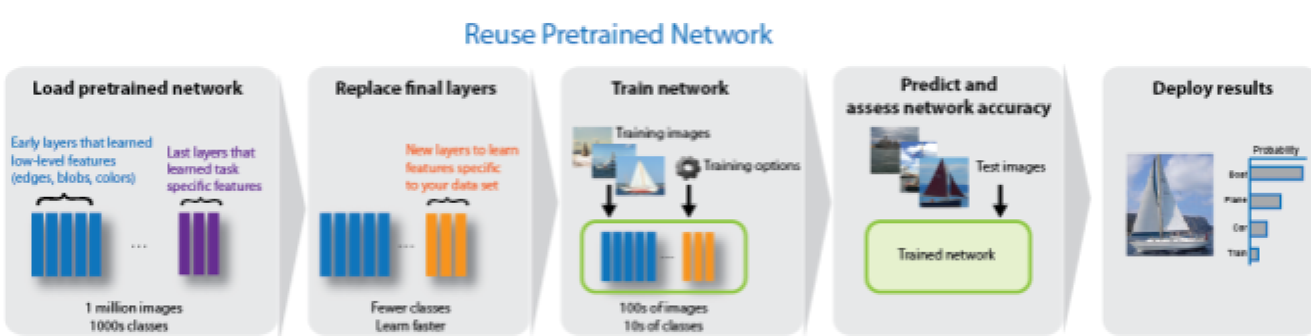
A method known as transfer learning was applied to create a CNN suited to analyze images of EMG signals. The algorithm was developed and run in MATLAB, using GoogleNet as a basis to create and structure the algorithm (MathWorks, *Googlenet* n.d.). GoogleNet - an open-source CNN - has been used in many research studies due to its high image recognition accuracy which surpasses other prebuilt CNN's used in other studies. The following function in Table VII was used to import the prebuilt CNN into the MATLAB workspace.

TABLE VII
GOOGLENET IMPORT

```
net = googlenet;
```

The GoogleNet algorithm was repurposed for this study to classify EMG signals into letters. This process of repurposing classification layers (Figure 12) from an existing model is known as transfer learning and is a common process used by Kapur and many other researchers. Transfer learning is beneficial as the CNN model developed using GoogleNet will likely have higher accuracy than other CNN algorithms (MathWorks, *Classification Learner App* n.d.).

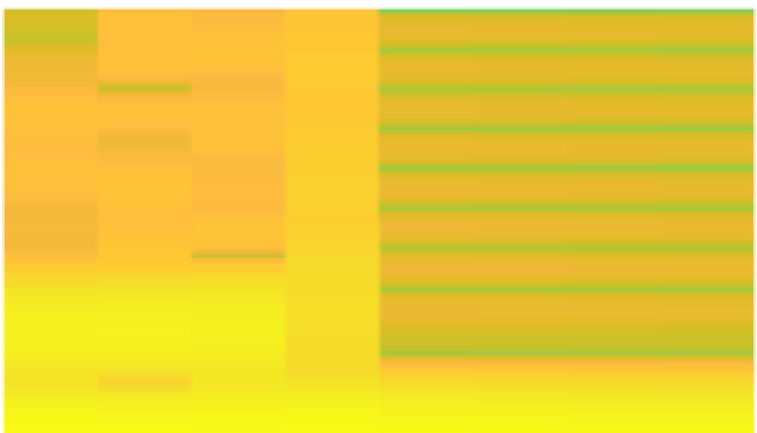
Figure 12: Transfer Learning Implementation



This figure shows the steps required to use transfer learning for an application

Because CNN's require image inputs for classification/training, the signal which was originally a series of numbers had to be converted into an image. The most effective way to convert a series of numbers into an image is by converting it to a spectrogram (Cohen, 2020). Spectrograms are a visual representation of signals and were also used in Kapur's study when developing an SSI. A spectrogram (Figure 13) for each of the 1020 EMG signals in the dataset was built using the function shown in Table VIII. These spectrograms were used to train/test the CNN just as Kapur's study did.

Figure 13: Spectrogram for the Vowel "A"



This figure shows the spectrogram for the vowel "A"

X-axis = time, Y-axis = frequency, green = higher amplitude

TABLE VIII
CREATING A SPECTROGRAM

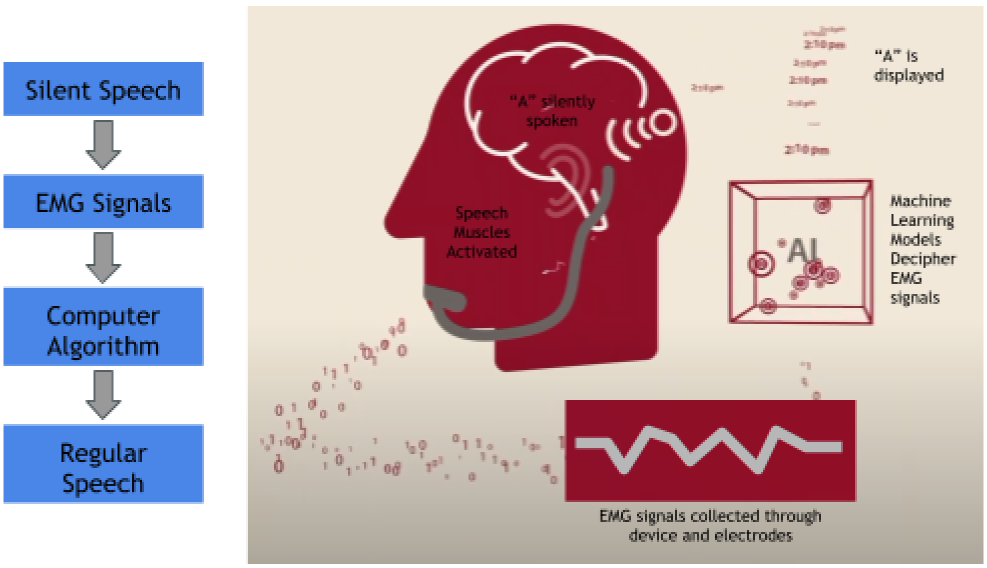
```
spectrogram(eval(signalName
), [], [], [], 'yaxis');
```

V. Creation of Silent Speech Interface

An SSI is a speech aid that records silent speech and uses an ML algorithm to translate the recorded EMG signals (Figure 14). Therefore the developed SSI has to be able to record EMG signals and then translate those signals using ML. This was accomplished by combining the previously built EMG recorder and a computer running ML algorithms.

Additionally, because the SSI's use the ML model to translate EMG signals into speech, the translation accuracy of the developed SSI is equal to the accuracy of the best performing ML model. Therefore, as explained before, the engineering goal for this project can be validated using the calculated accuracy of the best-performing ML model.

Figure 14: Silent Speech Interface Structure



This figure shows the general structure of an SSI that was used in this study as well as Kapur's study

Data Analysis & Results

As discussed in the methodology, the ML algorithms were tested using the same testing set. The classification accuracy and F1 scores were calculated for each tested model. F1 scores are commonly used by data scientists to compare ML models and determine which model is holistically better (Wood, 2019). The code for calculating F1 scores is shown in Table IX. The classification accuracies and F1 scores for the PR and CNN models are shown in Table X.

TABLE IX
CALCULATING F-SCORES

```
% Note: Variables were declared individually for each model
tp = 50; fp = 50; fn = 5;
precision = tp / (tp + fp); recall = tp / (tp + fn);
F1 = (2 * precision * recall) / (precision + recall);
```

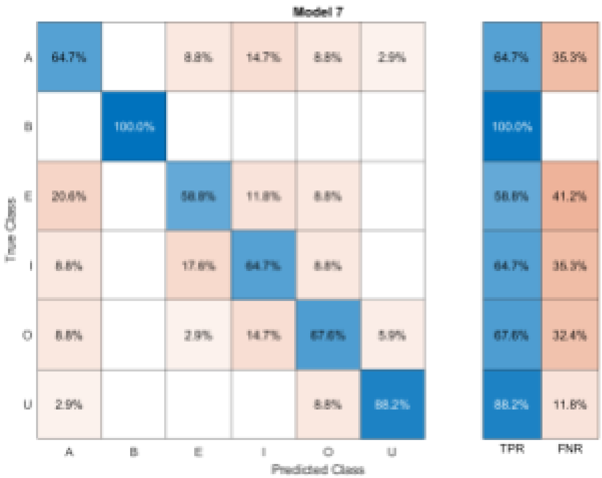
TABLE X
CLASSIFICATION ACCURACY AND F-SCORES

MODEL	CLASSIFICATION ACCURACY	F1 SCORES
CONVOLUTIONAL NEURAL NETWORK (CNN) - GOOGLENET	54.90%	0.60
SUPPORT VECTOR MACHINE (SVM) - GAUSSIAN	80.10%	0.81
ENSEMBLE - BAGGED TREES	74.60%	0.73
K-NEAREST NEIGHBORS (KNN) - WEIGHTED	66.70%	0.70
TREE - MEDIUM	59.80%	0.65
NAIVE BAYES - KERNEL	59.30%	0.65
QUADRATIC DISCRIMINANT	55.50%	0.54
LINEAR DISCRIMINANT	49.50%	0.48

As seen in Table X, the SVM model achieved the highest classification accuracy and F1 score. The accuracy values of the ML models tested ranged from 49.5% to 80.1% while the F1 scores ranged from 0.48 to 0.81. It can also be seen that the F1 scores closely correlated with the classification accuracy for each model and were often only ± 0.02 away from the classification accuracy.

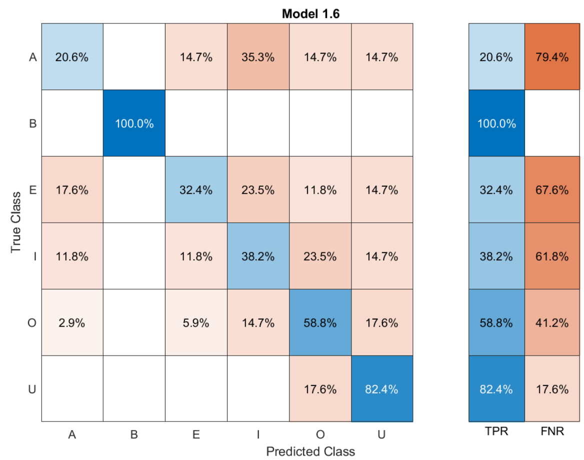
Confusion Matrices are a common way to depict the accuracies of ML models. The Confusion Matrices are shown for the SVM model (Figure 15), which had the highest F1-score/accuracy, and the CNN model (Figure 16) which was the only non-PR algorithm tested.

Figure 15: Confusion Matrix of Support Vector Machine



Confusion Matrix of the SVM model which has an 80.1% accuracy and F1 score of 0.81

Figure 16: Confusion Matrix of Convolutional Neural Network (CNN)



Confusion Matrix of the CNN which has a 54.9% accuracy and F1 score of 0.60

In Confusion Matrices (Figures 15 & 16), the rows denote what letter was silently spoken, whereas the column shows which letter was predicted by the trained ML algorithm. Therefore, all the correct predictions lie along the diagonal vector shaded in blue whereas incorrect predictions are shaded in orange.

Through analysis of the Confusion Matrix, it can be seen that both the SVM model (Figure 15) and CNN model (Figure 16) classified the signals for “not speaking” denoted by “B” (Blank) on the axes, with a 100% accuracy. The SVM model's largest error was due to misclassifying the EMG signals (for letter “E”) as the letter “A”. This accounted for 20.6% of incorrect predictions associated with the letter E. The CNN model’s largest error was due to the misclassifying the EMG signals (for letter “A”) as the letter “I”. This accounted for 35.3% of incorrect predictions associated with the letter “A”.

Both the SVM and CNN models performed poorly when classifying the letters “A” and “I”. This misclassification trend also occurred in the other tested PR models indicating that the EMG signals for “A” and “I” are hard to decipher. The SVM model had accuracies of 64.7% and 56.8% for the letters “A” and “I” respectively whereas the CNN model had accuracies of 20.6% and 32.4% for the same letters. These accuracy values for individual letters were the lowest and brought down the overall F1-score/accuracy.

Because these inaccurate predictions were produced by the ML models, the prediction accuracy can improve if the ML models are given more data to train(“learn”) from.

Discussion & Conclusion

I. Speech Impediments and Speech Aids

Speech Disorders are common among those with Motor Neuron Diseases (MNDs). MND patients are forced to use CSIs which are cumbersome and inaccurate speech aid systems. These systems make the user perform fatiguing muscle movements to select letters the user wants to communicate. These trivial devices prove to be an extremely slow and fatiguing solution for communication.

These issues prompted the development of an SSI which could be used by patients with speech disorders to communicate letters in the English alphabet voicelessly, merely by articulating words or sentences in the mouth without producing any sounds.

SSIs record EMG signals from speech muscles and translate these signals into speech with the use of ML. Existing research on SSIs makes use of high-end equipment (electromyograph) and also fails to justify the ML algorithms used to perform the EMG to speech translation. Furthermore, no study has identified the most accurate ML algorithm to use in an SSI. These gaps in research prompted this study which involved creating an accurate SSI and identifying the most accurate ML algorithm for use in an SSI.

II. Research Goals

Overall, this research study attempts to build an understanding to eliminate the existing gap in the field of knowledge by comparing two different types of ML Algorithms - CNNs and Pattern Recognition - to identify the most accurate ML algorithm for use in an SSI. Therefore, this project aims to construct two types of algorithms in order to determine which ML algorithm is most accurate for use in an SSI. An engineering goal was also developed, aiming to create an SSI with an 80% accuracy using low-cost muscle sensors as opposed to commonly used electromyographs.

III. New Understanding & Conclusions

An SSI was created (Figure 10), and 8 ML algorithms were tested. The calculated F1 scores of each model were used to determine the best performing model and the classification accuracy of the model with the highest F1 score was used to determine whether the engineering goal was achieved (Rationale for the usage of these metrics provided in Methods).

The classification accuracies and F-scores of the 8 tested models are listed in Table X. It was found that the highest F1-score (0.81) was achieved using the SVM model which is a type of PR algorithm. This means that the SVM Pattern Recognition model is the most accurate ML algorithm to use in an SSI for classifying EMG signals into speech, thus answering the research question. This is the first study to have ever identified the most accurate ML algorithm for classifying EMG signals in an SSI.

The classification accuracy for the SVM model, the best performing model, is 80.1%. This classification accuracy meets the engineering goal of 80% accuracy. This means that the created SSI, which used a Myoware muscle sensor, met the engineering goal and is the first study to have used a muscle sensor to implement an SSI rather than an electromyograph.

TABLE X
CLASSIFICATION ACCURACY AND F-SCORES

MODEL	CLASSIFICATION ACCURACY	F1 SCORES
CONVOLUTIONAL NEURAL NETWORK (CNN) - GOOGLENET	54.90%	0.60
SUPPORT VECTOR MACHINE (SVM) - GAUSSIAN	80.10%	0.81
ENSEMBLE - BAGGED TREES	74.60%	0.73
K-NEAREST NEIGHBORS (KNN) - WEIGHTED	66.70%	0.70
TREE - MEDIUM	59.80%	0.65
NAIVE BAYES - KERNEL	59.30%	0.65
QUADRATIC DISCRIMINANT	55.50%	0.54
LINEAR DISCRIMINANT	49.50%	0.48

IV. Explanation of Findings & Other Research

The SVM model likely outperformed the CNN model as it is a simpler model and can train on data quickly. On the other hand, the CNN, dealing with image inputs, trains slowly requiring more computing power. Because the number of iterations each model could train for was limited, the CNN wasn't able to fully train in the given amount of iterations. Therefore the SVM algorithm performed better.

The PR models tested had a wide range of accuracies. This can be attributed to the fact that some models, such as the Linear Discriminant, divide numerous classes ineffectively with simple methods. Although these algorithms perform well with small inputs, it wasn't useful to classify large EMG signals.

Karlik found in his study that the CNN is the most accurate algorithm for classifying EMG data for arm prosthesis. This study had different results because the nature of EMG signals from the arm and throat are different resulting in different optimal ML algorithms. Additionally, in this study, time and resource limitations could have impacted the performance of tested ML algorithms.

V. Limitations, Future Research & Implications

One limitation of this study is that it used a small dataset. Due to time constraints, the dataset developed to train/test the ML model contained only 1020 signals. With more training data, the accuracy of the ML algorithms would improve.

Another limitation is that this study cannot definitively identify the SVM algorithm as the most accurate ML model for the classification of EMG signals in an SSI. This is because only PR and CNNs were tested in the scope of this study. There are many types of ML algorithms, such as Artificial Neural Networks, that weren't tested in this study due to time constraints and complexity of models.

Future studies can address these limitations by evaluating/comparing more types of ML algorithms. Additionally, future studies could also develop datasets for a multitude of words from the English language. This would allow for the development of a more complete SSI that can truly be used in the real world.

This study's findings can help improve the accuracy of future SSIs by showing that SSIs can achieve better accuracy by using an SVM model. Additionally, the findings of this study can push researchers to develop SSIs without the use of electromyographs as this study was able to achieve good results using a low-cost muscle sensor.

References

Alaskar, H. (2018, December). *Convolutional neural network application in biomedical signals.* Retrieved November 24, 2020, from https://pdfs.semanticscholar.org/fd36/d9c6abf405af2c12d1596162421d3f503591.pdf

Arduino. (n.d.). *Arduino Mega 2560 rev3.* Retrieved December 18, 2020, from https://store.arduino.cc/usa/mega-2560-r3

Arslan, Y. Z. (2006, July). *Observations on the characteristics of EMG signals recorded at different depths.* Retrieved November 24, 2020, from https://www.researchgate.net/figure/Penetration-depth-of-needle-electrode_fig1_241204405

Ayodele, T. (2010, February). *Types of machine learning algorithms.* Retrieved November 24, 2020, from https://www.researchgate.net/publication/221907660_Types_of_Machine_Learning_Algorithms

Beukelman, D., & Garrett, K. (1988, July 12). *Augmentative and alternative communication for adults with acquired severe communication disorders.* Retrieved November 24, 2020, from https://www.tandfonline.com/doi/abs/10.1080/07434618812331274687

Bishop, C., & Nasrabadi, N. (1970, January 01). *Pattern recognition and machine learning: Semantic scholar.* Retrieved November 25, 2020, from https://www.semanticscholar.org/paper/Pattern-Recognition-and-Machine-Learning-Bishop-Nasrabadi/668b1277fbece28c4841eeab1c97e4ebd0079700

Bonaccorso, G. (2017, July 24). *Machine learning algorithm.* Retrieved November 25, 2020, from https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781785889622

Brown, S. (2000, October 01). *Swallowing and speaking challenges for the MS patient.* Retrieved October 25, 2020, from https://meridian.allenpress.com/ijmsc/article/2/3/4/33107/Swallowing-and-Speaking-Challenges-for-the-MS

Char, D., Chung, T., Mckee, A., & Pai, A. (2018, June). *The human keyboard.* Retrieved August 27, 2020, from https://scholarcommons.scu.edu/cgi/viewcontent.cgi?article=1043&context=idp_senior

Cohen, M. (2020, November 30). *Signal processing problems, solved in MATLAB and in python [MOOC]. Udemy.* Retrieved August 27, 2020, from https://www.udemy.com/course/signal-processing/.

Denby, B., Schultz, T., Honda, K., Hueber, T., Gilbert, J., & Brumberg, J. (2011, August 20). *Silent speech interfaces.* Retrieved August 25, 2020, from https://hal.archives-ouvertes.fr/hal-00616227

Eremenko, K., & Ponteves, H., DataScience, S. (n.d.). *Machine learning A-Z: Hands-on python & R in data science [MOOC]. Udemy.* Retrieved December 17, 2020, from https://www.udemy.com/course/machinelearning/

Hartman, K. (n.d.). *Getting started with Myoware muscle sensor.* Retrieved December 18, 2020, from https://learn.adafruit.com/getting-started-with-myoware-muscle-sensor

Kapur, A., Kapur, S., & Maes, P. (2018, March 5). *AlterEgo: A personalized wearable silent speech interface.* Retrieved October 25, 2020, from https://www.media.mit.edu/publications/alterego-IUI/

Kapur, A., Sarawgi, U., Wadkins, E., & Wu, M. (2019). *Non-invasive silent speech recognition in multiple sclerosis with dysphonia.* Retrieved August 24, 2020, from https://ml4health.github.io/2019/pdf/49_ml4h_preprint.pdf

Kareem, F., Azeem, M., & Sameh, A. (2017, October). Classification of EMG signals of lower arm (forearm\hand) motion patterns used to control robot hand movement. Retrieved December 30, 2020, from https://www.worldresearchlibrary.org/up_proc/pdf/1134-151151520140-45.pdf.

Karlik, B. (2018, July). *Machine Learning Algorithms for Characterization of EMG Signals.* Retrieved November 24, 2020, from, https://www.researchgate.net/publication/271298672_Machine_Learning_Algorithms_for_Characterization_of_EMG_Signals

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017, May). *ImageNet classification with deep convolutional neural networks.* Retrieved November 24, 2020, from https://dl.acm.org/doi/10.1145/3065386

MathWorks. (n.d.). *Googlenet.* Retrieved December 18, 2020, from https://www.mathworks.com/help/deeplearning/ref/googlenet.html

MathWorks. (n.d.) *Classification learner app.* Retrieved December 18, 2020, from https://www.mathworks.com/help/stats/classification-learner-app.html

Mohanty, S., Hughes, D., & Salathé, M. (2016, September 22). *Using deep learning for image-based plant disease detection.* Retrieved November 25, 2020, from https://pubmed.ncbi.nlm.nih.gov/27713752/

Muhammad, U. (2018, August 18). *Pre-trained VGGNet architecture for remote-sensing image scene classification.* Retrieved November 24, 2020, from https://www.researchgate.net/publication/329315021_Pre-trained_VGGNet_Architecture_for_Remote-Sensing_Image_Scene_Classification

Mwebaze, E., & Owomugisha, G. (2016, December). *Machine learning for plant disease incidence and severity measurements from leaf images.* Retrieved December 17, 2020, from https://www.researchgate.net/publication/313451669_Machine_Learning_for_Plant_Disease_Incidence_and_Severity_Measurements_from_Leaf_Images

Samuel, A. L. (1959, July). *Some studies in machine learning using the game of checkers.* Retrieved November 25, 2020, from https://ieeexplore.ieee.org/document/5392560

Schultz, T., & Wand, M. (2009, December 4). *Modeling coarticulation in EMG-based continuous speech recognition.* Retrieved August 24, 2020, from https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.212.6271&rep=rep1&type=pdf

Scikit-learn. (n.d.). *Scikit-learn machine learning in Python.* Retrieved December 17, 2020, from https://scikit-learn.org/stable/

Tang, P., Wang, H., & Kwong, S. (2017, February 15). *G-MS2F: GoogLeNet based multi-stage feature fusion of deep CNN for scene recognition.* Retrieved November 25, 2020, from https://scholars.cityu.edu.hk/en/publications/gms2f(97b29fed-c9f3-4838-a817-c90a321c3fc7).html

Wood, T. (2019, May 17). *F-score.* Retrieved December 17, 2020, from https://deepai.org/machine-learning-glossary-and-terms/f-score.

Zia ur Rehman, M., Gilani, S. O., Waris, M. A., & Niazi, I. K. (2018, July). *Stacked sparse autoencoders for EMG-based classification of hand motions: A comparative multi-day* *analysis between surface and intramuscular EMG.* Retrieved November 24, 2020, from, https://www.researchgate.net/publication/326292263_Stacked_Sparse_Autoencoders_for_EMG-Based_Classification_of_Hand_Motions_A_Comparative_Multi_Day_Analyses_between_Surface_and_Intramuscular_EMG

Academic Paper

Note: Student samples are quoted verbatim and may contain spelling and grammatical errors.

Sample: A
Score: 5

This paper earned a score of 5 because it provides clear and narrowing parameters on the topic between pages 9 and 13. The paper presents a clear research question and engineering goal on page 13, specifically stating, "To Develop a Speech Interface using a Muscle Sensor that can both collect and classify sEMG signals from the submental triangle (area under the chin) with greater than 80% accuracy." Further, the gap is signaled on page 9 and then explicitly stated and defended in language that is accessible for a non-discipline audience on pages 12-13. In addition, schematics and figures helped to enhance the communication. For example, the paper incorporates elements of the algorithm to illustrate the steps and parts of the method (See pages 18, 19, 21, 22, and 25). These also help to support where the evidence is generated and enhances the communication. Justification of the new understanding is found on page 30, where the paper ties back to the scholarly conversation. Specifically on page 30, the paper states, "Karlik found in his study that the CNN is the most accurate algorithm for classifying EMG data for arm prosthesis. This study had different results because the nature of EMG signals from the arm and throat are different resulting in different optimal ML algorithms." Further, the paper gives strong future research implications stating, on page 31, that "future studies could also develop datasets for a multitude of words from the English language. This would allow for the development of a more complete SSI that can truly be used in the real world." While some image captions do not have a source citation, credit is given to the source in the narrative. For example, on page 20, the paper states, "… by Kapur's research where he identifies various areas on the throat to collect EMG signals (Figure 4)." This paper did not earn a 4 because there is an explanation of the limitations on the conclusion on page 30, stating that "this study cannot definitively identify the SVM algorithm as the most accurate ML model for the classification of EMG signals in an SSI. This is because only PR and CNNs were tested in the scope of this study. There are many types of ML algorithms, such as Artificial Neural Networks, that weren't tested in this study due to time constraints and complexity of the models." Further, communication was enhanced by careful word choice and a series of visuals and tables that helped explain the research process.