Classification of Stuttered Speech Behaviors of Filipinos Who Stutter Using Machine Learning Algorithms

**Project Team Members**

Juan Diego V. Huet

Aaron Andrei D. Ambas

Kyle Rafael F. Sulabo

**Abstract**

The research aims to classify stuttered speech behaviors of Filipinos who stutter using machine learning algorithms. Preliminary research shows a method used by other researchers that classifies stuttered speech by obtaining Mel Frequency Cepstral Coefficients (MFCC’s) and using these as data to train machine learning classifiers. The researchers have employed this method but using a dataset built from audio recordings of Filipinos who stutter. Four Filipinos who stutter are employed as respondents. Stuttered instances are isolated from continuous speech by manually segmenting them. Thirteen MFCC’s are then extracted from each frame of each audio clip containing segmented stuttered speech. The MFCC’s for all frames of each audio clip are then used as data to be fed to the different machine learning algorithms employed namely, k-Nearest Neighbor, C4.5, Naïve Bayes, and Multilayer Perceptron. The algorithm that produced the most accurate model is the Multilayer Perceptron with an accuracy of 80%. Future researchers are recommended to employ more respondents when building a stuttered speech dataset to obtain a higher amount of instances used in this study. Also, more machine learning algorithms as well as more configurations can be used.

**Table of Contents**

**[CHAPTER I: INTRODUCTION………………………………………………………………1](#intro)**

[Background of the Study……………………………………………………………...1](#background)

[Statement of the Problem……………………………………………………………..2](#statement)

[Objectives……………………………………………………………………………....2](#objectives)

[Significance of the Study……………………………………...………………………2](#significance)

[Scope and Limitations…………………………………………………………………3](#scope)

[**CHAPTER II: Review of Related Literature………………………………………………5**](#rrl)

[Theoretical Background………………………………………………...…………….6](#theoretical)

[Related Studies………………………………………………………………..……..10](#related_studies)

[**CHAPTER III: Research Design and Methodology……………………………………13**](#methodology)

[Research Design……………………………………………………………………..13](#research_design)

[Data Gathering………………………………………………………………...……..13](#data_gathering)

[Data Processing………………………………………………………………...……14](#data_processing)

[**CHAPTER IV: Results and Discussion………………………………………………….17**](#results_discussion)

[k-Nearest Neighbor…………………………………………………………………..17](#k_nn_results)

[C4.5……………………………………………………………………………………18](#c45_results)

[Naïve Bayes…………………………………………………………………………..19](#nb_results)

[Multilayer Perceptron………………………………………………………………...19](#mlp_results)

[**CHAPTER V: Conclusions and Recommendations…………………………………..21**](#conclusions)

[**Bibliography…………………………………………………………………………………22**](#bibliography)

**[Appendix A…………………………………………………………………………………..25](#appendixA)**

[**Appendix B…………………………………………………………………………………..28**](#appendixB)

**CHAPTER I**

**INTRODUCTION**

**Background of the Study**

Stuttering is “…a speech disorder in which the flow of speech is disrupted by involuntary repetitions and prolongations of sounds, syllables, words or phrases as well as involuntary silent pauses or blocks in which the person who stutters is unable to produce sounds” (World Health Organization, 2010). The primary behaviors manifested by stuttering are repetition, prolongation of sounds, and blocking (Northern Arizona University, n.d.). A repetition occurs whenever the person who stutters repeats a sound, syllable, or a one-syllable word more than once or twice; a prolongation occurs when a speech sound is held out but the mouth, lips, or tongue stops moving; blocks occur when the person stops the flow of sound or air in the lungs, throat, mouth, lips, or tongue (HomeSpeechHome.com, n.d.).

On effort to alleviate stuttering, a speech language pathologist speech diagnosis then speech therapy. Speech therapy is the process of correcting disfluency in speech. A speech diagnosis is needed before the actual speech therapy in order to accurately decide what treatment a patient should receive (Scott, 2008).

In one variant of diagnosis is, the speech pathologist holds a casual conversation with the patient. Behaviors that are manifested physically by the patient are then documented by the speech pathologist. Afterwards, tests are conducted to measure the severity of stuttering by conducting speech focused tests that will test the patient’s fluency. The patient will be asked to read a passage where the percentage of the total number of words stuttered over the total number of words in the passage will be recorded. The patient will also be asked to hold a conversation with the speech pathologist with the patient discussing ideas about a certain topic, this time the patient’s voice will be recorded for use as reference. The percentage of the total number of words stuttered over the total number of words used by the patient will also be recorded. The speech pathologist also takes note of the different types of disfluency observed while conducting the speech tests.

The traditional method of diagnosing speech is generally time consuming. Another issue with it is that different speech pathologists may make their own different judgements when diagnosing (Kully and Boerg, 1988). Due to this, many different researchers have proposed many different techniques involving digital signal processing and machine learning to recognize stuttering, or classify fluent speech from non-fluent speech. However, it is also important to note that speech pathologists also need to manually classify different types of stuttered speech aside from just recognizing and counting them. There have been techniques proposed as well for classification of stuttered speech behavior types. Many of these studies propose to achieve this task by obtaining parametric representations of stuttered speech audio such as Mel Frequency Cepstral Coefficients and Linear Predictive Coefficients Cepstra, and utilizing various machine learning algorithms to classify stuttered speech into different behaviors, similar to techniques used in speech recognition. The results of these studies are promising, and theoretically proves that the methods are effective in classifying stuttered speech.

However, most of the studies conducted that classifies stuttered speech used the same source for the dataset, specifically the University College London Archive of Stuttered Speech (UCLASS). Also, no studies involving stuttered speech recognition of Filipinos who stutter have been gathered. The researchers saw this as an opportunity to conduct a study involving classification of stuttered speech from a different dataset, specifically of Filipinos. Hence, this study will be conducted to assess the effectiveness of different machine learning algorithms in classifying stuttered speech behaviors of Filipinos who stutter obtained from the raw dataset using Mel Frequency Cepstral Coefficients as the features. Furthermore, the study aims to determine the most effective method through analyzing of the results of the experiments that will be conducted.

**Statement of the Problem**

* How can stuttered speech behaviors be classified with the use of different machine learning algorithms?
* How effective will the different algorithms be in classifying stuttered speech behaviors?
* Which of the different algorithms used in the experiments is the most effective?

**Objectives of the Study**

* To implement different machine learning classifiers to classify audio with isolated stuttered speech instances based on the extracted audio features.
* To measure the effectiveness of each of the algorithms used in the experiments.

**Significance of the Study**

The findings of this research will benefit the following agencies:

*People who Stutter*

This project will benefit patients under speech therapy that will undergo speech diagnostics. Using a classifier model to classify repetitions and prolongations can make the process of differentiating repetitions from prolongations more objective. Having a more objective classification of stuttered speech behaviors can improve the accuracy of diagnoses.

*Speech Pathologists*

Speech Pathologists will be able to use this project to produce more accurate and more objective results when conducting a speech diagnosis. Better diagnostic results will lead to more appropriate therapies among patients. This can also be used alongside the traditional method of diagnosis to lessen ambiguity.

*Other Researchers*

This group of people will benefit the most from the findings of this study. It should be noted that the scope of the study is relatively small-scale, and requires the integration of other methods and research in order for it to be used in practical applications. Hence, other researchers will benefit from this study by being introduced to a potentially new method of stuttered speech classification. The findings in this study can also be integrated or supplemented by other researchers in their studies.

**Scope and Limitations**

The study aims to measure the effectiveness of different machine learning algorithms in classifying isolated stuttered speech instances. It is very important to note that this study only involves the classification of isolated stuttered speech. Explicitly, the research does not aim to detect stuttered speech from continuous speech. The expected input of the classifier models are audio features from segments of stuttered speech sounds, namely repetitions and prolongations. As stated before, two of the problems that traditional speech therapy has is that it is too time consuming and that different judges might make different counts when diagnosing stuttering. Although it is clear that the study will classify stuttered speech, it is important to note that it will not directly impact the time it takes to do so. The project does not aim to implement the results of the research into a working product/software, but rather aims to prove the proposition of whether or not certain algorithms can be used to classify stuttered speech behaviors, and determine which algorithm will be the most effective in doing so. Due to this, factors such as the time it takes for a supposed automated classifier to classify stuttered speech will not be measured against the time it takes for speech pathologists classify stuttered speech.

Also, the study will only be concerned with repetitions and prolongations as the stuttered speech behaviors to be classified. There are many behaviors that can be manifested by a person who stutters, however not all those behaviors can be manifested in voice recordings. Due to this the researchers can only classify between stuttered speech behaviors that can be recognized in audio, although the researchers have only selected repetitions and prolongations as behaviors to be classified to limit the scope of the study. It should also be noted that due to the rarity of stuttering among people, the researchers were only able to employ four people who stutter as the respondents. Due to this, the amount of instances that will be inputted to the machine learning classifiers will be relatively few as compared to the average amount of instances used by the other said researches conducted. As a possible result, the produced classifiers in this study might produce at lower accuracy as compared to other researches.

**CHAPTER II**

**REVIEW OF RELATED LITERATURE AND STUDIES**

This chapter discusses essential and relevant theories and concepts that are heavily involved in this study, as well as other related research that the authors have found relevant and/or led to the formulation of this study. The chapter will also discuss how these relevant theories/concepts and research are used or involved in the study.

**Behaviors of Stuttering**

Stuttering behaviors are mainly characterized into primary and secondary. The primary behaviors are:

* Repetitions of sounds, syllables and words
* Prolongation of single sounds
* Blocks of airflow when speaking

Secondary behaviors include:

* Hesitations
* Interjections of sounds, syllables of words (ahh, uhm)
* Word revision, word changes
* Unnecessary motor movements

(Northern Arizona University, n.d.)

There are many behaviors manifested by a person who stutters. However, only a few of those behaviors can be manifested in a voice recording. Hence, the researchers only considered classifying repetitions from prolongations manifested in voice recordings. Repetitions occur when the person who stutters repeats a sound, syllable, or a word with a single syllable. An example could be:

"*W- W- W-* Where are you going?"

In this example, the person is having a hard time moving from the “w” in “where” to the remaining sounds of the word. The person completes the word on the fourth attempt.

In this study, the researchers will be concerned on the parts where a person will have a difficulty pronouncing. In this example, the “w-w-w-” part of the phrase/sentence will be considered.

Prolongations occur when a person who stutters prolongs pronouncing a single syllable without pronouncing another syllable. An example for this could be:

"*SSSS* ave me a seat."

The person is having a hard time moving from the “s” in “save” to the remaining part of the word (American Speech-Language-Hearing Association, n.d.). Likewise, the researchers will be concerned on the parts where the person will have a difficulty pronouncing the syllable. In this example, the “ssss” part of the sentence will be considered.

**Theoretical Background**

**Mel Frequency Cepstral Coefficients**

Introduced by Davis and Mermelstein in the 1980’s, MFCC represents the envelope of the short time power spectrum manifested by the shape of the vocal tract of humans. MFCCs give accurate representations of phonemes being produced (practicalcryptography.com, n.d.). MFCC’s are widely used in automatic speech and speaker recognition, and is the most cutting-edge ever since it was proposed. Before MFCCs were introduced, Linear Predictive Coefficients (LPC) and Linear Predictive Coefficients Cepstra (LPCC) were used were used in speech recognition.

In this study, MFCC’s will be extracted from segmented stuttered speech audio signals. These features will be used as the dataset that will be fed to the model. Basically, the MLP model will base its classification by using the extracted MFCC’s as the previous information.

To get the MFCC’s of a signal, the following steps have to be done:

1. Frame the signal into multiple frames of length 20-40 milliseconds(ms) with a corresponding frame step. 25 milliseconds is the standard. Assuming that the signal is sampled at 16khz, the frame length of the 25ms signal is 400 frames (0.025\*16000 = 400). The frame step allows some overlap to the frames. Assuming that the frame step is 10ms (160 samples), the first 400 sample frame starts at sample 0, the next 400 sample frame starts at sample 160 and so on until the end of the speech file is reached.

The next steps will be applied to every single frame obtained from Step 1. One set of MFCC’s will be obtained from each frame.

1. Take the Discrete Fourier Transform (DFT) of each frame by performing:



Where:

* is the Fourier transform of and is the signal
* is the total number of frames

1. Convert the resulting spectrum from Step 2 into Mel scale using the following formula:



Where:

* is frequency in hertz

1. Take the log of the resulting spectrum from Step 3.
2. Take the Discrete Cosine Transform of the log spectrum from Step 4 by performing:



Where:

* is the frame number
* is the number of cepstrum coefficients

The result of the process are MFCC’s representing each frame of the input speech signal.

**Machine Learning**

Machine Learning is a type of artificial intelligence that provides computers with the ability to learn without being explicitly programmed (Samuel, 1959). It involves development of computer programs that can teach themselves to grow and adapt to change when new data is exposed. Machine Learning systems look through data to look for patterns and detect those patterns in the data to and adjust actions accordingly. A machine learning model refers to the model artifact created by training (Amazon Web Services Documentation, Amazon Machine Learning Guide, Training Machine Learning Models).

Machine Learning algorithms are usually classified into two: Supervised Learning and Unsupervised Learning. Supervised learning uses what it has learned in the past and applies it to new data. Some examples of algorithms that belong to Supervised Learning are Support Vector Machines, linear regression, naive Bayes, Neural Networks, etc. Unsupervised Learning on the other hand describes hidden patterns from unlabeled data. Some examples are clustering through k-means, mixture models, or hierarchical clustering; anomaly detection; and Neural Networks.

In this study, different supervised learning classification algorithms will be used in the sense that the model will be trained with a dataset with labelled instances. Each instance will be labelled as either from the class repetition or prolongation.

**Naïve Bayes**

Predictions can be made using the Bayes Theorem:

Where:

* P(h|d) is the probability of hypothesis h given the data d
* P(d|h) is the probability of data d given that the hypothesis h was true
* P(h) is the probability of hypothesis h being true (regardless of the data)
* P(d) is the probability of the data (regardless of the hypothesis)

**C4.5 (Decision Tree)**

The general algorithm for building decision trees is:

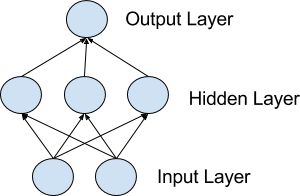
1. Check for the above base cases.
2. For each attribute *a*, find the normalized information gain ratio from splitting on *a*.
3. Let *a\_best* be the attribute with the highest normalized information gain.
4. Create a decision *node* that splits on *a\_best*.
5. Recur on the sublists obtained by splitting on *a\_best*, and add those nodes as children of *node*.

**k-Nearest Neighbor**

When used in classification, k-NN can calculate the output as the class with the highest frequency from the k-most similar instances. The algorithm classifies an unknown instance by getting *k* instances with the closest attributes with the unknown instance. k-NN is simple as the model representation is just the entire training dataset.

**Multilayer Perceptron**

Multilayer Perceptrons are neural networks; a perceptron is a single neuron model that was a precursor to larger neural networks. The building block of these neural networks are artificial neurons where each neuron is a computational unit where it receives weighted input values and produce an output value using an activation function. The weighted input values are produced by multiplying the input from a previous neuron to a weight. Weights are often initialized as small random values, usually from size 0 to 0.3.



*Figure I: Example model of a Neural Network*

The neurons are arranged to form a network of neurons, producing an input layer, hidden layer(s), and an output layer. Each neuron from the input layer represents the number of attributes of each instance, and the output layer represents the number of classes. An instance may be classified as either of the classes by an activation function to output probabilities in the output layer, then the output with the highest probability can be produce a classification. When training a neural network, the weights are updated by comparing the output to the expected output and the error is calculated. This error is then passed through back the network and each of the weights are updated according to the amount of how they contributed to the error. Once the network has been trained, it can be used to make predictions.

**Related Studies**

There have been numerous studies conducted that proposed to classify or identify stuttered speech with the use of machine learning, psychology, linguistics, and digital signal processing. The following is a discussion of related research that the researchers have found relevant.

In 1995, Howell and Sackin classified stuttered speech into repetitions and prolongations with the use of an artificial neural network (ANN). They extracted 39 acoustic parameters, 20 vector based on autocorrelation function plus spectral coefficient based on a 19 channel vocoder. Envelope of speech waveform was obtained by filtering the signal using a 10hz lowpass filter. The best hit/miss rate was 0.82 for prolongations and 0.77 for repetitions. Further research was done by Howell, Sackin, and Glenn in 1997. In this research, they explored classification of stuttered words from fluent words, as well as classify stuttered words into prolongations and repetitions using ANN. For the dataset, they employed 12 children who speak stuttered English. The speech samples can be obtained from the University College London Archive of Stuttered Speech (UCLASS). For this study, speech were manually segmented into individual words. For the attributes, the researchers used whole word and part word duration; whole word, first part, and second part fragmentation; whole word, first part, and second part spectral measure; and part word energy. These attributes are then input to the networks. The classifier yielded 95% accuracy for fluent words and 78% for disfluent words. From the disfluent words, a 58% accuracy was achieved for prolongations and 43% accuracy for repetitions.

Geetha et. al (2000) researched on using ANN to classify children who stutter from children with normal non-fluency by using medical data as input. 51 children were employed as respondents; the medical data of 25 children were used to train the ANN, and 26 were used for testing. The attributes that are used as input are age, sex, type of disfluency, frequency of disfluency, duration, physical concomitant, rate of speech, historical, attitudinal and behavioral scores, and family history. A 92% accuracy of predicting normal nonfluency and stuttering was achieved.

Szczurowska et al. (2006) experimented with using Konohen and Multilayer Perceptron networks in classifying fluent and disfluent speech. Recordings were taken eight stuttering Polish speakers and segmented disfluent 4-second-long fragments containing disfluency. Speech of fluent speakers containing the same fragments were also recorded. All utterances were analysed by FFT 512 with the use of a 21 digital 1/3-octave ﬁlters of centre frequencies between 100 and 10000 Hz and an A-weighting ﬁlter. FFT time resolution was 23 ms, which transformed every four-second sample into 21 vectors consisting of 171 time points. These were then inputted to different types of MLP networks for training. A best accuracy of 76.67% was achieved.

A study by Ravikumar et al. (2008) explored the possibility of automatic detection of repetitions in read speech. This study proposed automatic detection of repeated syllables using four major steps they defined: segmentation, feature extraction, score matching, and decision logic. During the data gathering part, the researchers employed ten people who stutter with a mean age of 25 as the respondents. A standard English Passage of 150 words was selected as the reading passage to be read by the respondents and these speeches were recorded at a sampling rate of 16000 samples per second. The collected audio samples are first manually segmented by the researchers into syllables. After segmentation, the segmented speech syllables are subject to feature extraction; 12 Mel Frequency Cepstral Coefficients(MFCC) were extracted from the segmented speech syllables. Score matching is then done using the Dynamic Time Warping Algorithm. The angle between the segmented speech syllables were computed by using DTW on the MFCC’s. These values were given to the decision logic to identify whether the syllables were repeated or not. The perceptron algorithm was used as the decision logic. This proposed approach by the researchers achieved an accuracy of 83%.

Another study by Mahesha and Vinod (2012) proposed classification of disfluent speech using k-Nearest Neighbors algorithm and Support Vector Machine. The researchers defined disfluent speech as speech containing repetitions, prolongations, interjections, and pauses. For the data gathering, audio samples used in the study was obtained from the University College London’s Archive of Stuttered Speech (UCLASS). Samples are taken from standard reading of 25 different speakers with age between 10 to 20 years. After obtaining the samples, the researchers segmented disfluent speech manually. Another pool of fluent speech is produced by 20 fluent speakers with a mean age group of 25 using the same reading passages used in the UCLASS database. Fluent and disfluent speech were segmented manually from these speech samples; overall, a speech corpus that contains 50 fluent and 50 disfluent speech segments is created. 40 fluent and 40 disfluent speech segments will be used for training the models, while 10 fluent and 10 disfluent speech segments will be used to test the models. Next, the MFCC’s are obtained from the fluent and disfluent speech samples and used as the data to train two models with two machine learning algorithms (K-NN classifier and Support Vector Machine). The produced model using k-nn classifier achieved an accuracy of 86.67% in classifying disfluent speech and 93.34% in classifying fluent speech while the produced model using support vector machine algorithm achieved an accuracy of 90% in classifying disfluent speech and 96.67% in classifying fluent speech.

Hariharan et al. (2012) proposed classification of speech disfluencies with MFCC and LPCC features. The study used speech samples from UCLASS, speech samples from 39 people were used for this study; it includes one sample each from two female speakers and 37 male speakers ranged between 11 years two months and 20 years and one month. The speech samples contain speeches of reading passages specifically “One More Week to Easter” and “Arthur the Rat”; each of the two passages contains more than 300 words. The two types of disfluencies considered by the researchers were repetitions and prolongations and were identified and segmented manually by listening to the recorded speech signals. The segmented speech samples were downsampled from 44.1khz to 16khz and pre-emphasized with a high pass filter. After this the MFCC’s and LPCC’s were extracted from the segmented speech samples with different frame lengths and overlaps. The study experimented with 10, 20, 30, 40, and 50ms frame lengths and no overlap, 33.33%, 50%, and 75% overlap. The study also experimented with different high pass filters (alpha=0.91 to alpha=0.99). After feature extraction, k-Nearest Neighbor and Linear Discriminant Analysis classifiers were used. The best results achieved for MFCC and LPCC features are: 94.51% for LPCC (frame length=30ms, window overlap=75%, alpha=0.98) and 92.55% for MFCC (frame length=20ms, overlap=50%, alpha=0.9375).

The two studies conducted by Mahesha and Vinod, and Hariharan et al. give great significance to this research as their works prove that MFCC’s are effective parametric representations of stuttered speech in machine learning. This study will use MFCC’s as the input features to the different machine learning algorithms, similar to the methods used by the said studies.

Based on the literature review, no known research has been yet conducted that utilizes Filipino stuttered speech as the dataset. Most of the researches conducted that proposes to classify stuttered speech behaviors have used the UCLASS database as their training and testing dataset. The researchers aim to conduct further research that involves stuttered speech behavior classification but in a local context.

**CHAPTER III**

**RESEARCH DESIGN AND METHODOLOGY**

In this chapter, the nature of the research design used and the series of methods used in conducting the research experiments will be comprehensively discussed.

**Research Design**

This study is mainly a quantitative-experimental and comparative type of research. The study is quantitative in nature since the experiments conducted heavily involve the use of a quantitative type of dataset, specifically the MFCC’s of the isolated stuttered speech signals. MFCC’s serve as the parametric representation of the audio signals that will be used as the dataset for training the machine learning algorithms. Furthermore, the effectiveness of each of the algorithms used in the experiments will be quantified by their accuracy, as well as Precision, Recall, and F-measure. The study is an experimental type of research because all the conclusions that will be drawn will come from the results of the experiments conducted. In addition, the study is also a comparative type of research as the effectiveness of each of the algorithms used will be compared, and the most effective will be noted.

**Data Gathering**

In order for one to train a classifier model, one must input a training dataset to the algorithm where each instance of dataset is labelled with the class where they belong. In this case, two classes will be used: repetitions and prolongations.To build this dataset, the researchers employed four male Filipinos who stutter as their respondents. Due to the rarity of stuttering, the researchers were only able to find a limited number of Filipinos who stutter who were willing to act as respondents. The respondents were taken from followers of the Philippine Stuttering Association; three passages, specifically Grandfather Passage, The Rainbow Passage, and Limpy were asked to be read while having their voice recorded. The voice recordings produced were given to the researchers for further analysis.

**Data Processing**

This part of the methodology will be split into different steps for ease of comprehension. This part will consist of three general steps: Audio Segmentation, Feature Extraction and Preprocessing, and Model Training and Testing. Audio Segmentation will describe the isolation of stuttered speech from continuous speech. Feature Extraction and Preprocessing will discuss how the parametric features from audio clips are extracted, and Machine Training and Testing will discuss how machine learning classifier algorithms are used to distinguish repetitions and prolongations.

Segmentation

All the voice recordings obtained from the respondents are first converted to 16khz using the software, FFmpeg. FFmpeg is a software containing libraries that record, convert, and stream audio and video. After conversion, the voice recordings were manually analyzed by the researchers and manually segmented the parts where each of prolongations and repetitions occurred into many audio clips using Audacity. Audacity is an open source audio software for audio editing. This produced 55 audio clips with 33 of them containing prolongations and 22 of them containing repetitions. The produced audio clips will be subject to feature extraction.

Feature Extraction and Preprocessing

13 Mel Frequency Cepstral Coefficients will be extracted from each frame of each segmented stuttered audio clips. The researchers used *jAudio* to automatically extract the 13 MFCC’s for each frame in each audio signal. jAudio is a framework for audio feature extraction. It meets the needs of Music Information Retrieval(MIR) researchers by providing a library of analysis algorithms that are suitable for a wide array of MIR tasks. The researchers used a 512 frames window size (32ms) and a 256 frame window overlap. The result is each audio clip will produce a 13 x N matrix of values with N as the total number of frames an audio signal has. These values will be fed to the different machine learning algorithms. However, the the segmented audio clips are of different lengths, and this will produce matrices of different sizes.

To solve this issue of matrices with variable lengths, the researchers used *zero padded normalization* to convert the matrices to uniform sizes. Zero padding is done by inserting zeroes to fill empty frames for matrices that has features less than the number of inputs expected by the classifier. In this study, the researchers assigned the length of the largest produced matrix as the number of inputs expected by the classifier, in this case 80. The researchers used zero padding as the normalization technique as zero padding is previously known as an effective technique (Salam et. al, 2009). After normalization, 55 matrices each of size 13 x 80 is produced. Each matrix is then converted to a single vector. This is done by flattening each matrix into a vector of size 1040 x 1. Each of the 13 MFCC’s per frame is combined in a single vector row (13 MFCC’s \* 80 frames = 1040 values). This produces 55 vectors of size 1040 x 1. Each of the vector represent each instance that will be fed to the different machine learning algorithms using Weka.

Model Training and Testing

*Table I: Dataset used in the experiments*

|  |  |
| --- | --- |
| Instances: | 55 |
| Attributes: | 1040 |
| Classes: | 2 |

*Table II: Class Distribution of the Dataset in number of instances*

|  |  |
| --- | --- |
| Repetitions | Prolongations |
| 22 | 33 |

The researchers used Weka to simulate the different classifier algorithms that will be benchmarked in the study. Weka is a collection of machine learning algorithms written in Java and developed at the University of Waikato, New Zealand. It contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. Four classifier algorithms have been employed in the study, **Naive Bayes**, **C4.5** (“J48” in Weka), **Multilayer Perceptron**, and **k-Nearest Neighbor**. For Naive Bayes and C4.5, the default settings in Weka were used. For k-Nearest Neighbor, the number of neighbors used is 1. For the Multilayer Perceptron Classifier, the learning rate used is 0.1 and the momentum rate is 0.9; the pair of values is considered as standard for MLP by Rumelhart (Rumelhart, 1986).

To test the effectiveness of the algorithms used, 10-fold cross validation will be performed instead of splitting the dataset into training and testing instances. 10-fold cross validation is performed by producing 10 models of the same algorithm where for each model, 90% will be used for training and 10% will be used for testing. For each of those 10 models, a different set of training instances will be used such that all the instances will act as part of the training data for one of the ten models. The results of the 10 models will be averaged.

To measure the effectiveness of each of the different algorithms, different measures will be taken into account. First of all, the accuracy or percentage of the correctly classified instances will be considered. Aside from accuracy, the Precision, Recall, and F-measure of each class of each model, as well as the weighted measures for the whole model. Precision is defined as the fraction of instances correctly classified as positive out of all the elements the algorithm classified as positive in a certain class, while Recall is the fraction of instances correctly classified as positive by the algorithm out of all the positive instances in a certain class. F-measure is a combined metric derived from Precision and Recall. The formulas for the four measures are:

where True Positives are instances correctly classified as a given class, True Negatives are instances that are correctly classified as not belonging in a certain class, False Positives are instances falsely classified as a given class, and False Negatives are instances falsely classified as not belonging to the class it belongs. The optimal values for these measures are 100% (or 1), it being the maximum value attainable.

**CHAPTER IV**

**RESULTS AND DISCUSSION**

This chapter will discuss the results of the experiments conducted, and the interpretations of results. The dataset gathered is used to train the four classifiers using the different machine learning algorithms used. Below are the results gathered for each classifier.

**k-Nearest Neighbor -** with k=1

*Table III: Classifier Results for the k-NN Classifier*

|  |  |  |  |
| --- | --- | --- | --- |
| **Correctly Classified Instances (accuracy): 69.0909 %** | | | |
| *Class/Measure* | Precision | Recall | F-Measure |
| Repetition | 0.692 | 0.409 | 0.514 |
| Prolongation | 0.690 | 0.879 | 0.773 |
| **Weighted Average** | **0.691** | **0.691** | **0.670** |

*Table IV: Confusion Matrix for the k-NN Classifier*

|  |  |  |
| --- | --- | --- |
|  | Classified as Repetition | Classified as Prolongation |
| Actual Repetition | **9** | 13 |
| Actual Prolongation | 4 | **29** |

The k-Nearest Neighbor Classifier classified 38 instances correctly out of the 55 instances, yielding an accuracy of 69.0909%. The weighted average of Precision is 0.691, 0.691 for Recall, and 0.670 for F-measure. Out of the 22 repetitions, the classifier was able to correctly classify 9 of them; out of the 33 prolongations, the classifier was able to correctly classify 29 of them.

**C4.5 -** Default Settings

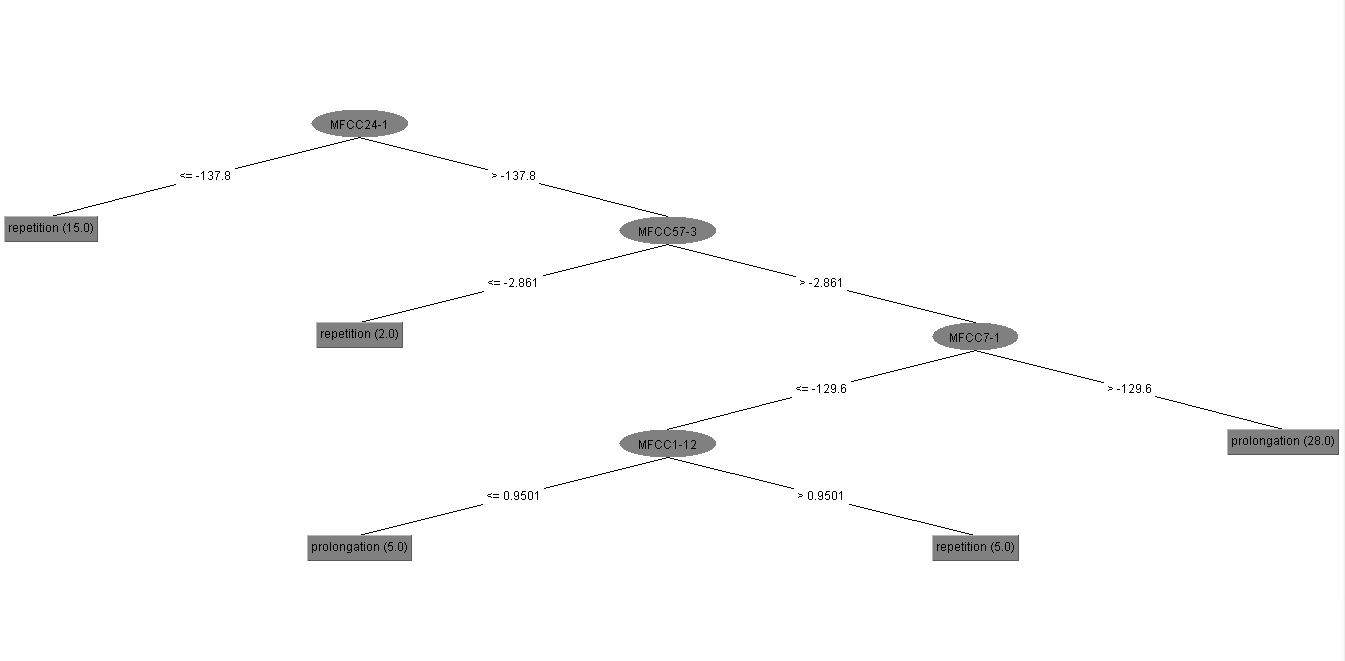
*Table V: Classifier Results for the C4.5 Classifier*

|  |  |  |  |
| --- | --- | --- | --- |
| **Correctly Classified Instances (accuracy): 78.1818 %** | | | |
| *Class/Measure* | Precision | Recall | F-Measure |
| Repetition | 0.727 | 0.727 | 0.727 |
| Prolongation | 0.818 | 0.818 | 0.818 |
| **Weighted Average** | **0.782** | **0.782** | **0.782** |

*Table VI: Confusion Matrix for the C4.5 Classifier*

|  |  |  |
| --- | --- | --- |
|  | Classified as Repetition | Classified as Prolongation |
| Actual Repetition | **16** | 6 |
| Actual Prolongation | 6 | **27** |

The C4.5 classifier seems to have relatively better results as compared to the Naïve Bayes classifier. The C4.5 classifier was able to classify 43 instances correctly out of 55 instances. The weighted average for Precision, Recall, and F-measure is 0.782. The classifier was able to classify 16 repetitions correctly out of 22, and 27 prolongations correctly out of 33.



*Figure II: Visualization of the C4.5 tree produced*

Based on the visualization of the tree, the algorithm notices key differences between repetitions and prolongations in the dataset from the attributes: first coefficient of frame 24, third coefficient of frame 57, first coefficient of frame 7, and twelfth coefficient of frame 1.

**Naive Bayes -** Default Settings (Gaussian Distribution)

*Table VII: Classifier Results for the k-NN Classifier*

|  |  |  |  |
| --- | --- | --- | --- |
| **Correctly Classified Instances (accuracy): 72.7273 %** | | | |
| *Class/Measure* | Precision | Recall | F-Measure |
| Repetition | 0.733 | 0.500 | 0.595 |
| Prolongation | 0.725 | 0.879 | 0.795 |
| **Weighted Average** | **0.728** | **0.727** | **0.715** |

*Table VIII: Confusion Matrix for the k-NN Classifier*

|  |  |  |
| --- | --- | --- |
|  | Classified as Repetition | Classified as Prolongation |
| Actual Repetition | **11** | 11 |
| Actual Prolongation | 4 | **29** |

The Naive Bayes classifier successfully classified 40 instances out of the 55, yielding an accuracy of 72.7273%. The weighted average of precision is 0.728, 0.727 for recall, and 0.715 for f-measure. The classifier successfully classified 11 out of the 22 repetitions correctly, and 29 of the 33 prolongations correctly.

**Multilayer Perceptron -** Learning Rate = 0.1, Momentum Rate 0.9

*Table IX: Classifier Results for the MLP Classifier*

|  |  |  |  |
| --- | --- | --- | --- |
| **Correctly Classified Instances (accuracy): 80%** | | | |
| *Class/Measure* | Precision | Recall | F-Measure |
| Repetition | 0.789 | 0.682 | 0.732 |
| Prolongation | 0.806 | 0.879 | 0.841 |
| **Weighted Average** | **0.799** | **0.800** | **0.797** |

*Table X: Confusion Matrix for the MLP Classifier*

|  |  |  |
| --- | --- | --- |
|  | Classified as Repetition | Classified as Prolongation |
| Actual Repetition | **15** | 7 |
| Actual Prolongation | 4 | **29** |

The Multilayer Perceptron Classifier performed slightly more effective as compared to the C4.5 classifiers by successfully classifying 44 instances out of the 55, achieving an accuracy of 80%. The weighted average of precision is 0.799, 0.800 for recall, and 0.797 for f-measure. Out of 22 repetitions, the classifier was able to correctly classify 15 repetitions, and 29 prolongations out of 33.

*Table XI: Summary of accuracy of the different classifiers*

|  |  |  |
| --- | --- | --- |
| *Classifier/Measure* | Accuracy | Weighted F-measure |
| k-Nearest Neighbor | 69.0909 % | 0.670 |
| C4.5 | 78.1818 % | 0.782 |
| Naive Bayes | 72.7273 % | 0.715 |
| Multilayer Perceptron | **80%** | 0.797 |

Based on the results gathered from the experiments conducted, the Multilayer Perceptron performed the best out of the four algorithms used with an accuracy of 80%. While the MLP classifier was chosen to be the most effective, it is also evident that the performance of the other three classifiers are satisfactory. However, when compared to other said similar researches, the classifiers produced in this study are relatively less effective in terms of accuracy. The lower than expected performance of the classifiers in this study may be attributed to the low number of instances used as compared to past researches.

**CHAPTER V**

**CONCLUSIONS AND RECOMMENDATIONS**

This study aims to classify stuttered speech behaviors of Filipinos using machine learning algorithms. The researchers used a method based on previous techniques used by other researchers to classify stuttered speech behaviors to apply on the dataset that the researchers built. This was done by extracting the Mel Frequency Cepstral Coefficients of each frame of each audio clip containing stuttered speech. The MFCC’s served as parametric representations that were used as input data to the different machine learning classifier algorithms. The researchers made use of four different machine learning algorithms namely, k-Nearest Neighbor (k=1), Naive Bayes, C4.5, and Multilayer Perceptron (Learning Rate = 0.1, Momentum Rate = 0.9). The classifiers achieved accuracies of 69.0909 %, 78.1818 %, 72.7273 %, and 80%, respectively. Out of the four classifiers produced, the Multilayer Perceptron classifier proved to be the most effective.

It must be noted that the study utilized a dataset that only contained 55 instances, which is greatly smaller as compared to other studies such as Hariharan et. al, and Mahesha and Vinod. Hence, future researchers are advised to attempt to include more Filipinos who stutter as respondents in order to gain a greater number of instances for a dataset. Using a dataset with a higher amount of instances can produce more accurate classifiers. Future researchers are also recommended to attempt to use other parametric representations of audio aside from MFCC’s, as well as experiment with more machine learning algorithms. Finally, different settings for the different machine learning algorithms should also be experimented on.

**Bibliography**

American Speech-Language-Hearing Association. *Stuttering.* Retrieved on November 30, 2016, from <http://www.asha.org/public/speech/disorders/stuttering/>

Amazon Web Services. *Amazon Machine Learning Documentation*. Retrieved September 5, 2016, from <https://aws.amazon.com/documentation/machine-learning/>

Arizona Board of Regents. *Fluency Disorders - Communication Sciences and Disorders - Northern Arizona University*. Retrieved September 5, 2016, from <https://nau.edu/chhs/csd/clinic/fluency-disorders/>

Audacity Team (2016). Audacity(R): Free Audio Editor and Recorder. Version 2.1.2 [Software] retrieved from <http://audacity.sourceforge.net/>.

Brownlee, J. (2016). Crash Course On Multi-Layer Perceptron Neural Networks. *Machine Learning Mastery*. Retrieved from http://machinelearningmastery.com/neural-networks-crash-course/ on December 8, 2016.

Brownlee, J. (2016). K-Nearest Neighbors for Machine Learning. *Machine Learning Mastery*. Retrieved from http://machinelearningmastery.com/k-nearest-neighbors-for-machine-learning/ on December 8, 2016.

Brownlee, J. (2016). Naive Bayes Tutorial for Machine Learning. *Machine Learning Mastery*. Retrieved from <http://machinelearningmastery.com/naive-bayes-tutorial-for-machine-learning/> on 07 Dec. 2016.

Chee, L. S., Ai, O. C., Hariharan, M., & Yacob, S. (2010). Automatic detection of prolongations and repetitions using LPCC.

Davis, S. Mermelstein, P. (1980) Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. *In IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 28 No. 4, pp. 357-366

Deng, Li; Douglas O'Shaughnessy (2003). *Speech processing: a dynamic and optimization-oriented approach*. [Marcel Dekker](https://en.wikipedia.org/wiki/Marcel_Dekker). pp. 41–48. [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-8247-4040-8](https://en.wikipedia.org/wiki/Special:BookSources/0-8247-4040-8)

Eibe Frank, Mark A. Hall, and Ian H. Witten (2016). The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition, 2016.

FFmpeg Developers. (2016). ffmpeg tool (Version 20160808-ce2217b) [Software]. Available from <http://ffmpeg.org/>

Geetha, Y. V., Pratibha, K., Ashok, R., & Ravindra, S. K. (2000). Classiﬁcation of childhood disﬂuencies using neural networks. *Journal of Fluency Disorders*, 25(2), 99–117.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann P., Witten, I. (2009). The WEKA DataMining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1.

Hariharan, M., Yaacob, S., Chee, L. S., & Ai, O. C. (2012). Classiﬁcation of speech dysﬂuencies with MFCC and LPCC features. *Expert Systems with Applications*, *39*, 2157–2165.

HomeSpeechHome. *Stuttering, everything you need to know in simple terms*. Retrieved September 5, 2016, from <http://www.home-speech-home.com/stuttering.html>

Howell, P., & Sackin, S. (1995). Automatic recognition of repetitions and prolongations in stuttered speech. *In Proceedings of the ﬁrst World Congress on ﬂuency disorders.*

Howell, P., Sackin S., & Glenn, K. (1997a). Development of a two-stage procedure for the automatic recognition of disfluencies in the speech of children who stutter: I. Psychometric procedures appropriate for selection of training material for lexical disfluency classifiers. *Journal of Speech, Language, and Hearing Research*, 40(5), 1073.

Howell, P., Sackin, S., & Glenn, K. (1997b). Development of a two-stage procedure for the automatic recognition of disfluencies in the speech of children who stutter: II. ANN recognition of repetitions and prolongations with supplied word segment markers. *Journal of Speech, Language, and Hearing Research*, 40(5), 1085.

Kotsiantis S.B., Supervised Machine Learning: A Review of Classification Techniques, *Informatica* 31(2007) 249-268, 2007

Kully, D., & Boerg, E. (1988). An Investigation of Inter-clinic Agreement in the Identification of Fluent and Stuttered Syllables. *Journal of Fluency Disorders*, *13*, 309–318.

Mahesha, P., & Vinod, D. (2012). An Approach for Classification of Disfluent and Fluent Speech using K-NN and SVM. *International Journal of Computer Science, Engineering and Applications (IJCSEA),* 2(6), 23–32.

[McEnnis, Daniel, Ichiro Fujinaga, Cory McKay, Philippe DePalle. 2005. "JAudio: A feature extraction library". ISMIR.](http://ismir2005.net/proceedings/2003.pdf)

Nanopoulos, A., Alcock, R., & Manolopoulos, Y. (2001). Feature-based Classification of Time-series Data.

O'Shaughnessy, Douglas (1987). *Speech communication: human and machine.* Addison-Wesley. p. 150. [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [978-0-201-16520-3](https://en.wikipedia.org/wiki/Special:BookSources/978-0-201-16520-3).

Practical Cryptography. (2009). *Mel Frequency Cepstral Coefficient (MFCC) tutorial*. Retrieved September 5, 2016, from <http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/>

Ravikumar, K., Reddy, B., Rajagopal, R., & Nagaraj, H. (2008). Automatic Detection of Syllable Repetition in Read Speech for Objective Assessment of Stuttered Disfluencies. *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, *2*(10), 2142–2145.

Rumelhart. D.E., G.E. Hinton and R.J. William, *Learning representation by back-propagation errors*. Nature, 323: 533-536, 1986.

Salam, M., Mohamad, D., & Salleh, S. (2009). Temporal Speech Normalization Methods Comparison in Speech Recognition Using Neural Network. *International Conference of Soft Computing and Pattern Recognition,* 442-447.

Simon, P. (2013). [*Too Big to Ignore: The Business Case for Big Data*](https://books.google.com/books?id=Dn-Gdoh66sgC&pg=PA89#v=onepage&q&f=false). Wiley. p. 89. [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [978-1-118-63817-0](https://en.wikipedia.org/wiki/Special:BookSources/978-1-118-63817-0).

Szczurowska, I., Kuniszyk-Jozkowiak, W., & Smolka, E. (2006). The Application of Kohonen and Multilayer Perceptron Networks in the Speech Nonfluency Analysis. *Archives of Acoustics*, *31*(4), 205–210.

**Appendix A**

**Reading Passages Used**

*Grandfather Passage*

You wish to know all about my grandfather. Well, he is nearly 93 years old, yet he still thinks as swiftly as ever. He dresses himself in an ancient, black frock coat, usually minus several buttons.

A long, flowing beard clings to his chin, giving those who observe him a pronounced feeling of the utmost respect. When he speaks his voice is just a bit cracked and quivers a trifle. Twice each day he plays skillfully and with zest upon a small organ.

Except in the winter when the snow or ice prevents, he slowly takes a short walk in the open air each day. We have often urged him to walk more and smoke less but he always answers, "Banana oil!" Grandfather likes to be modern in his language.

*Rainbow Passage*

When the sunlight strikes raindrops in the air, they act as a prism and form a rainbow. The rainbow is a division of white light into many beautiful colors. These take the shape of a long round arch, with its path high above, and its two ends apparently beyond the horizon. There is, according to legend, a boiling pot of gold at one end. People look, but no one ever finds it. When a man looks for something beyond his reach, his friends say he is looking for the pot of gold at the end of the rainbow. Throughout the centuries people have explained the rainbow in various ways. Some have accepted it as a miracle without physical explanation. To the Hebrews it was a token that there would be no more universal floods. The Greeks used to imagine that it was a sign from the gods to foretell war or heavy rain. The Norsemen considered the rainbow as a bridge over which the gods passed from earth to their home in the sky. Others have tried to explain the phenomenon physically. Aristotle thought that the rainbow was caused by reflection of the sun's rays by the rain. Since then physicists have found that it is not reflection, but refraction by the raindrops which causes the rainbows. Many complicated ideas about the rainbow have been formed. The difference in the rainbow depends considerably upon the size of the drops, and the width of the colored band increases as the size of the drops increases. The actual primary rainbow observed is said to be the effect of super-imposition of a number of bows. If the red of the second bow falls upon the green of the first, the result is to give a bow with an abnormally wide yellow band, since red and green light when mixed form yellow. This is a very common type of bow, one showing mainly red and yellow, with little or no green or blue.

*Limpy*

Limpy is a fuzzy, yellow, baby duck. He belongs to a fisherman. The fisherman lives in a little house by the bay. Every morning children go swimming in the bay. About 10:00, Limpy waddles out to the road to wait for the children. When he hears them coming he begins a loud, excited quacking.

The children always bring bread or corn for Limpy. He will nip at their fingers or peck at their bare toes until he is fed. Limpy never follows the children down to the shore. He likes to swim in his own little pond. It is much safer.

**Appendix B**

**Weka Classifier Results**

**C4.5**

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: jAudio

Instances: 55

Attributes: 1041

[list of attributes omitted]

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree

------------------

MFCC24-1 <= -137.8: repetition (15.0)

MFCC24-1 > -137.8

| MFCC57-3 <= -2.861: repetition (2.0)

| MFCC57-3 > -2.861

| | MFCC7-1 <= -129.6

| | | MFCC1-12 <= 0.9501: prolongation (5.0)

| | | MFCC1-12 > 0.9501: repetition (5.0)

| | MFCC7-1 > -129.6: prolongation (28.0)

Number of Leaves : 5

Size of the tree : 9

Time taken to build model: 0.08 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 43 78.1818 %

Incorrectly Classified Instances 12 21.8182 %

Kappa statistic 0.5455

Mean absolute error 0.2167

Root mean squared error 0.4541

Relative absolute error 45.0011 %

Root relative squared error 92.5067 %

Total Number of Instances 55

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.727 0.182 0.727 0.727 0.727 0.545 0.776 0.675 repetition

0.818 0.273 0.818 0.818 0.818 0.545 0.776 0.775 prolongation

Weighted Avg. 0.782 0.236 0.782 0.782 0.782 0.545 0.776 0.735

=== Confusion Matrix ===

a b <-- classified as

16 6 | a = repetition

6 27 | b = prolongation

**k-Nearest Neighbor**

=== Run information ===

Scheme: weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Relation: jAudio

Instances: 55

Attributes: 1041

[list of attributes omitted]

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

IB1 instance-based classifier

using 1 nearest neighbour(s) for classification

Time taken to build model: 0 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 38 69.0909 %

Incorrectly Classified Instances 17 30.9091 %

Kappa statistic 0.3089

Mean absolute error 0.3165

Root mean squared error 0.5454

Relative absolute error 65.7144 %

Root relative squared error 111.113 %

Total Number of Instances 55

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.409 0.121 0.692 0.409 0.514 0.332 0.654 0.557 repetition

0.879 0.591 0.690 0.879 0.773 0.332 0.654 0.687 prolongation

Weighted Avg. 0.691 0.403 0.691 0.691 0.670 0.332 0.654 0.635

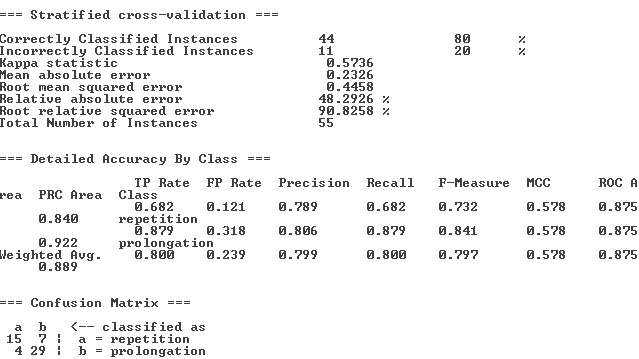
=== Confusion Matrix ===

a b <-- classified as

9 13 | a = repetition

4 29 | b = prolongation

**Multilayer Perceptron**

****

**Naïve Bayes**

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 40 72.7273 %

Incorrectly Classified Instances 15 27.2727 %

Kappa statistic 0.4

Mean absolute error 0.2727

Root mean squared error 0.5222

Relative absolute error 56.6255 %

Root relative squared error 106.3964 %

Total Number of Instances 55

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.500 0.121 0.733 0.500 0.595 0.417 0.762 0.666 repetition

0.879 0.500 0.725 0.879 0.795 0.417 0.677 0.708 prolongation

Weighted Avg. 0.727 0.348 0.728 0.727 0.715 0.417 0.711 0.691

=== Confusion Matrix ===

a b <-- classified as

11 11 | a = repetition

4 29 | b = prolongation