IOP Conf. Series: Materials Science and Engineering **260** (2017) 012042 doi:10.1088/1757-899X/260/1/012042

Speech to Text Translation for Malay Language

Rami Ali Al-khulaidi, Rini Akmeliawati

International Islamic University Malaysia, Department of Mechatronics Engineering Jalan Gombak, 53100 Kuala Lumpur, Selangor, Malaysia

teemomory@gmail.com , rakmelia@iium.edu.my

Abstract

The speech recognition system is a front end and a back-end process that receives an audio signal uttered by a speaker and converts it into a text transcription. The speech system can be used in several fields including: therapeutic technology, education, social robotics and computer entertainments. In most cases in control tasks, which is the purpose of proposing our system, wherein the speed of performance and response concern as the system should integrate with other controlling platforms such as in voiced controlled robots. Therefore, the need for flexible platforms, that can be easily edited to jibe with functionality of the surroundings, came to the scene; unlike other software programs that require recording audios and multiple training for every entry such as MATLAB and Phoenix. In this paper, a speech recognition system for Malay language is implemented using Microsoft Visual Studio C#. 90 (ninety) Malay phrases were tested by 10 (ten) speakers from both genders in different contexts. The result shows that the overall accuracy (calculated from Confusion Matrix) is satisfactory as it is 92.69%.

1. Introduction

Spoken language is a main way of communication between human beings. However, it has become normal in our daily life seeing individuals utter words to devices such as laptop computers or smart phones; it's even a necessity in many cases like when a person with a certain disability must instruct a certain machine or a computer application to perform a certain task for him or her. Speech-to- text analysis attracts more attentions in the last decades. Therefore, such technology can be encountered in many different contexts from the inevitable use of the speech recognition system, along with human robot or animated Avatar, as a mediator between normal and hearing-impaired individuals to the joyful use in computer gaming.

Thus, being a part in the rapid improvement of this technology, we implemented a speech recognition system for Malay Language using C# in Microsoft Visual Studio Platform. It can be easily integrated with other components for controlling tasks. Microsoft visual studio environment has introduced itself as a future promising environment for developers interested in this area. So, it is a momentous advantage the developers can start benefiting thereof. Especially as the Microsoft Visual Studio integrates with controlling platforms such as Arduino and raspberry pi using windows form application and serial communication.

Unlike previously implemented systems that were sophisticated and limited in terms of words, speed and accuracy, our system is adaptable for future addition and edition, and capable to tactfully integrate with other platforms; these features are the major reasons behind implementing the system in Microsoft Visual Studio.

This paper is organised as follows. Section 2 shows some related works. Section 3 presents the review of the system along with some parts of algorithms used. The testing data are presented in Section 4. Section 5 shows the evaluation of the results, and finally, Section 6 presents the conclusion and the future works.

2. Related Work

A huge amount of research has been done in the area of speech processing locally (in Malaysia) and worldwide. [1] developed an algorithm to synthesize Malay spoken language using Hidden Markov Model (HMM) which performed well with a reasonable accuracy. There are also some works worldwide for different languages such as English, Mandarin, Indian, Spanish, French and others.
[2] developed an automatic speech recognition (ASR) system for Malay speaking children. They took advantage of HMM available in HTK Toolkit to build the speech model for Malay language. The developing environment was Linux based. The system is relatively good since it can accurately recognize of up to 76% of test words. However, the ASR focused only on small group of children (six children) to record their voice and include them in the system.

Reference [3] created an application to test word pronunciation for pre-school children aged between 3 and 6 years old. The system focused especially on vowel letters. The interface was in MATLAB using Spectrum Delta (SPD) features and Logistic Regression classification model. Although the accuracy was up to 92.29%, the system only tested vowels letters.

As we can see from both works, [2] and [3], regardless of the accuracy of the systems, they are only focused on a very narrow area which is on the recognition of children pronunciation. Moreover, both systems targeted small groups input, six children voice and three to six-year-old children.

Malay Speech Therapy Assistance Tools (MSTAT) that is used to diagnose children for language disorder and to train children with stuttering problem was proposed in [4]. Voice patterns of the normal and stutter children are used to train the HMM model which is used to evaluate speech problem for stuttering children. The accuracy of the system was reasonable but the system itself is used for diagnostic purposes.

In [5], fusion techniques and adapting filtering is used for Malay language recognition. They used Mel-Frequency Cepstral Coefficients (MFCC) vectors to provide an estimate of the vocal tract filter. Dynamic Time Warping (DTW) and HMM are the two recognition algorithms used DTW is used to detect the nearest recorded voice. Meanwhile, HMM is used to emit a new feature vector for each frame according to an emission probability density function associated with that state. They also use RLS noise cancellation, end point detecting, framing, normalization, filtering. The system is cumbersome since it included many processing and filtering techniques. Therefore, the complexity of the system makes it unsuitable to be applied for additional purposes such as translation because it will be so slow and it may interfere with other functions that improve real time translation or any other kind of performance such as voiced controlled machinery.

All the stated works are about Malay speech recognition and synthesizing. Some of them were focused on small groups of population including children whether in small group [2] and [3] or in general but restricted to patient children only [4]. The others that involved the speech of all age groups for Malay people in general like [5]. Unfortunately, they were quite intricate which make them difficult to be applied in other functions such as controlling other devices. For example, when we want to vocally instruct a robot or at least ordering a certain machine to perform a certain task.

3. System Overview

A speech recognizer is a system that receives an audio stream and converts it into a text transcription. The process of speech recognition is considered as front end and back end. The process of isolating segments of sound, speech, and converting the segments into a series of numeric values that characterize the vocal sounds in the signal is the front-end process. The back end, on the other hand, is a search engine which is specialized to take the output of front end and look out across three databases as in Figure 1:

- The acoustic model which represents the language acoustic sounds; this can be trained to recognize the characteristics of a certain user's speech patterns and acoustic environments.
- The lexicon includes a large number of the words in the language, and offers information on how to pronounce each word.
- The language model represents the means in which the words of the language are combined.

The quality of the speech recognizer is detected by how good the recognizer at sanitizing its search, removing the poor matches, and choosing the more similar matches. This fully depends on the quality of the language and acoustic models of the recognizer and the algorithms effectiveness, for the processing of input sound and the searching across its models.

3.1 Grammar

The grammar is the most substantial part of the speech recognition system as it limits the number of recognized words and phrases. The speech recognizer often needs to process certain utterances with specific semantic meaning to the intended application rather than using the general language model, such as every day spoken language. This will have several benefits to the system such as, increasing the accuracy of the system, ensuring that all recognized sets are meaningful to the intended application as well as specifying the semantic values inherent in the detected text. This grammar can be authoring programmatically in the Microsoft Speech Platform SDK.

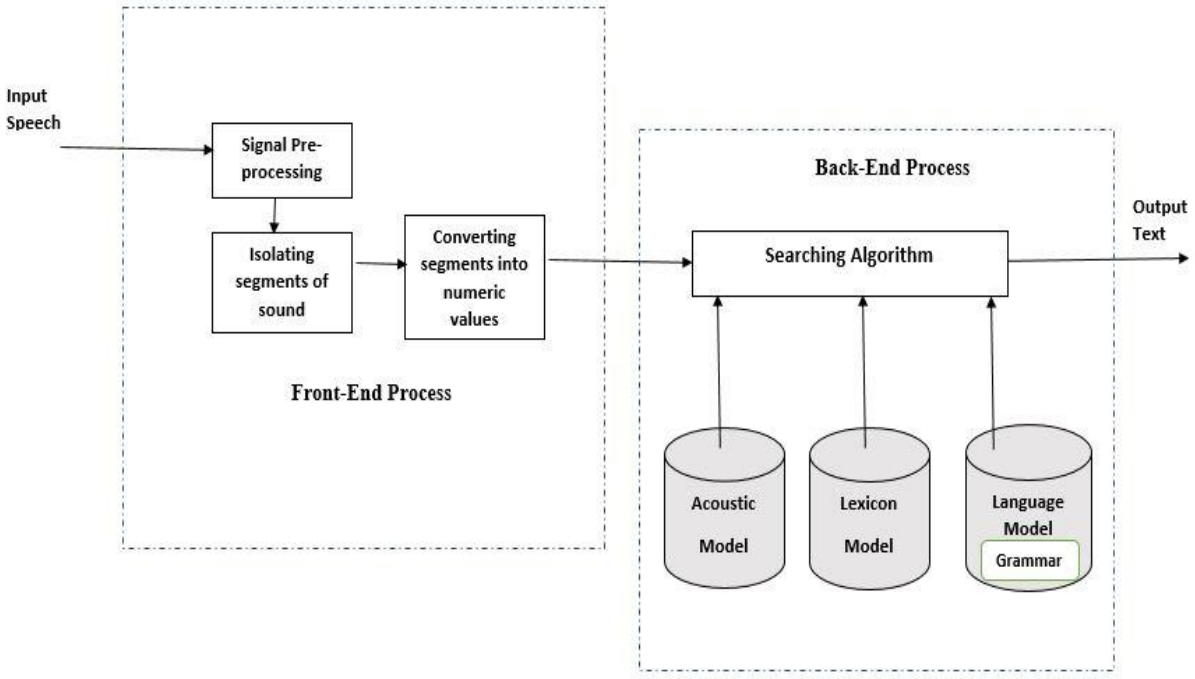

Figure 1. The Flowchart of Speech System

3.2 System Design

The process of implementing speech recognition to a Windows Forms application starts by launching MS Visual Studio and creating a new C# Windows Forms application with the proposed name. Before creating any program with speech recognition system in C#, "System.speech" library has to be added by "using" namespace statement at the beginning of the code, and following by an instance of the "SpeechRecognitionEngine".

The grammar also must be added into the "SpeechRecognitionEngine". Otherwise, the speech recognizer will not be able to recognize the targeted phrases. This could be the counterpart of the pre-recorded database in other software such as, MATLAB. But, here only texts of the exact wording of intended phrases should be inserted instead. The words and phrases included in the Grammar List are

presented in Table 1. The method "RecognizeAsync" was used, to load grammars asynchronous, as well as event handler which is used to display the recognized word or item. The execution of the program starts by initializing the grammar instance along with grammar builder object. Then, the grammar will be loaded into the speech recognizer following by event handler. Next, the Graphical User Interface will show up. However, the recognizing will not commence unless the "Start" pushbutton is clicked. Once the pushbutton is pressed, the system will be waiting for an utterance to analyse and decide if it's included in the grammar list. Consequently, the method "sRecognize_SpeechRecognized" will be invoked if the speech is recognized. Then, accordingly the result property of event handler will display the recognized phrase if it is included in grammar list. Otherwise, a "Say that again" message will appear telling us to speak again. The program will continue as long as the quit button is not clicked. The flowchart of the main program is shown in Figure 2 below.

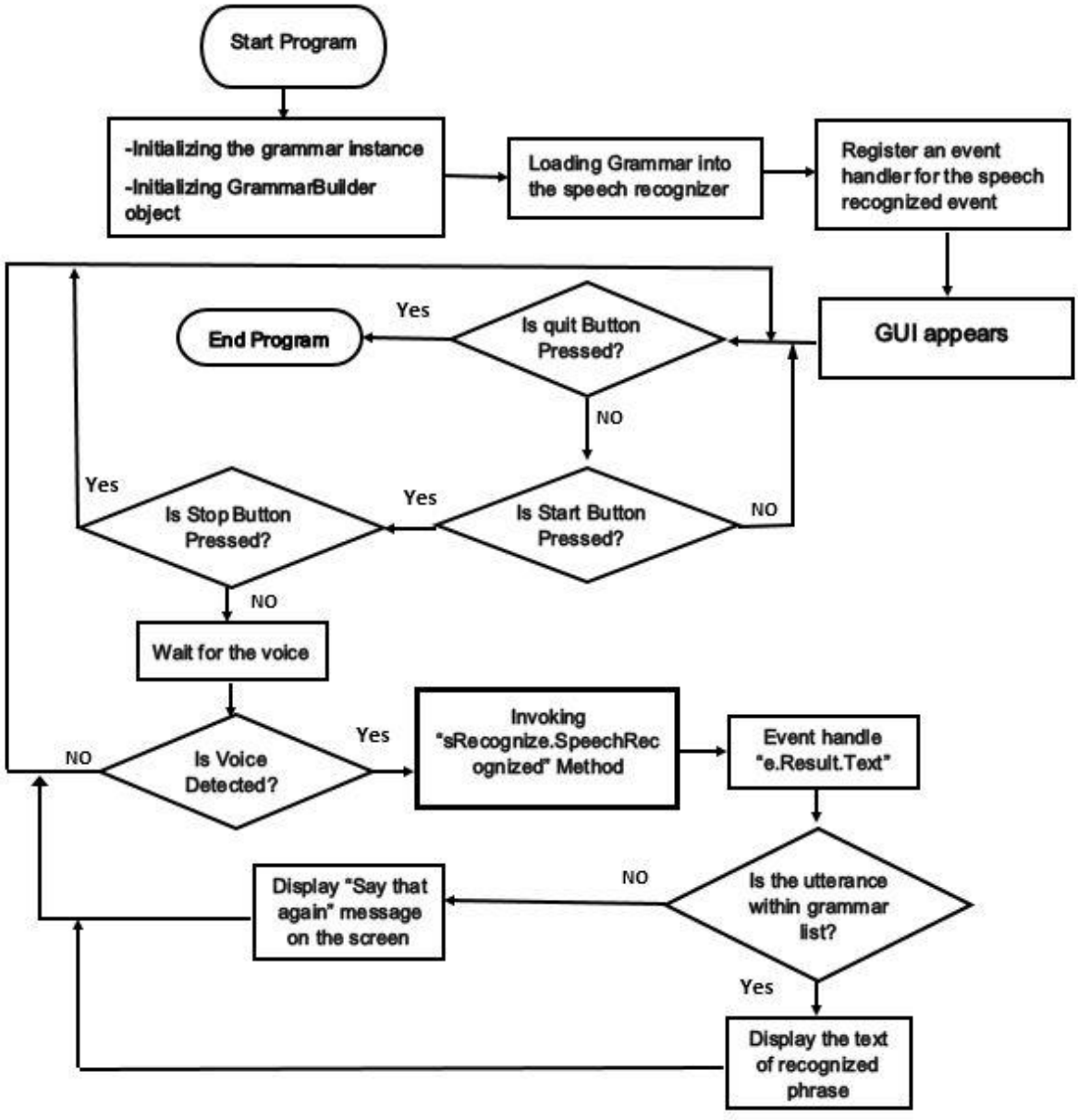

Figure 2. The Flowchart of The Main Program

3.3 User Interface

The design of user interface provides all the information about the recognized items and the control buttons of the system. We can see the data about the words such as the recognized words or phrases as it appears on the screen along with time of recognition and "say that again" message in case the word is not recognized. The control over the user interface includes the "start" button that tells the system to start recognizing the sound and the "stop" button that temporarily forces the system to stop. The user can also quit the interface window by either vocal command, saying "quit program" command, or by clicking on the quit button. Figure 3.

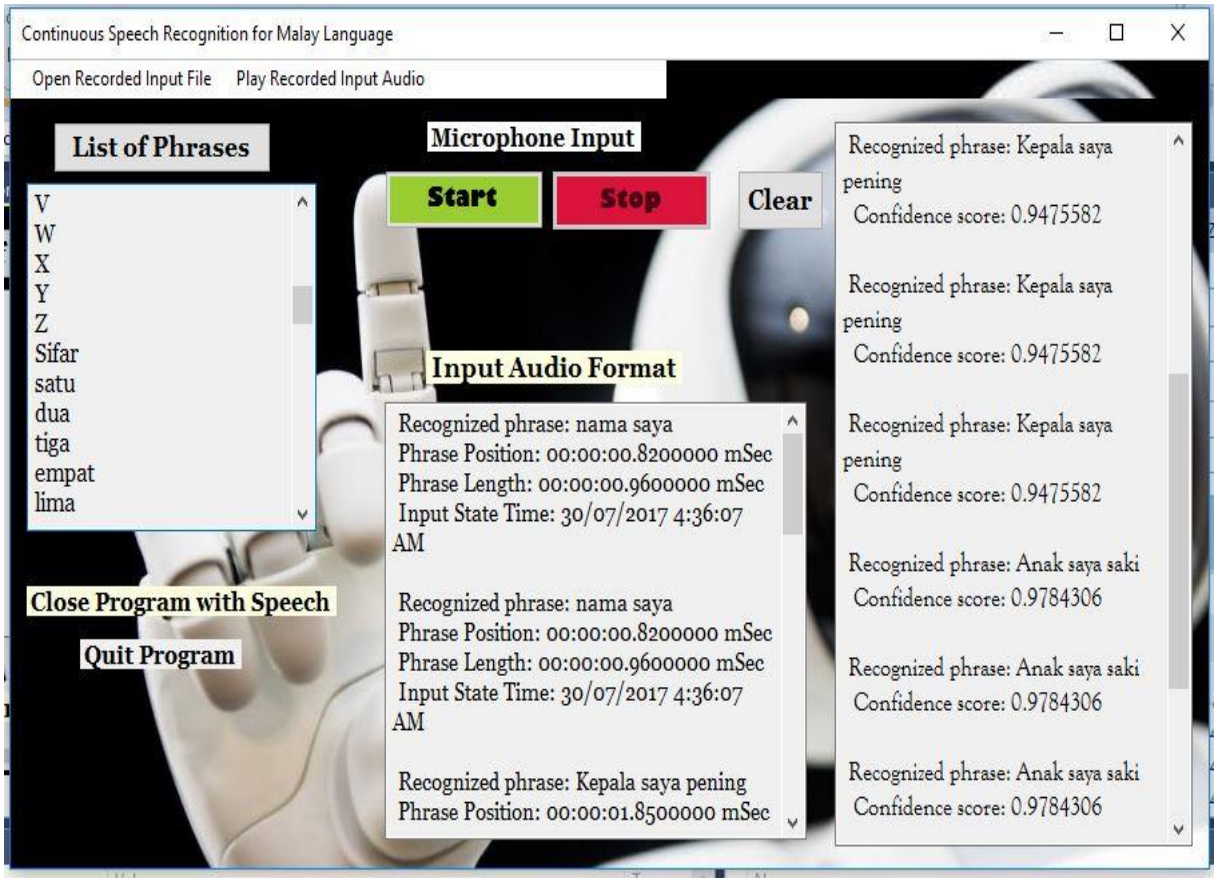

Figure 3. User Interface of Speech Recognition System

4. Testing of the System

Using the standard microphone of a quite old laptop (Dell Inspiron 1545), ninety (90) tested items (alphabets, single words and phrases) were tested by 10 male and female speakers. To ensure the accuracy of the system, we intended to ask each speaker to repeat each tested item five-times. Hence, the total of the data tested is 4500 which will be analysed in section 5. The criteria of choosing such 90 items lies on three crucial reasons: the recurrence of phrases, the diversity of contexts and the urgent situations.

Amongst all the commonly used phrases for Malay language, we carefully picked up the most recurrent phrases; phrases that are used in a daily life context such as salutations, or those used in classroom context such as giving permission for entering or leaving classrooms. In addition, some vital conversational phrases that are used between two or three individuals, considering the phrases that are frequently used as emergency calls or seeking help contexts.

Table 1 below shows the tested phrases in three different contexts: classroom, normal conversation and some phrases in restaurant. The reason for choosing the restaurant phrases is to test the system in chaotic environments and determine how accurate the system will be under such circumstances.

Table 1: The Tested Phrases

55 Class rooms Phrases	19 Conversational Phrases	6 Restaurant phrases
The 26 Malay alphabets, and 26 numbers (numbers from 0 to 14, and 20, 21, 22, 23, 30, 40, 50, 100, 200, 300, 1000. Three sentences also are included: (Sila Masuk Sila Dudduk and Tolong diam)	Nama saya, jumpa laga, saya datang dari, siapa nama anda, boleh cakap bahasa Melayu, saya sakit, pergi kemana, saya pergi ke hospital, Wang saya dicuri, jalan terus, badan saya ada luka, anak saya sakit, ada pencuri masuk rumah, dokumen saya hilang, kereta saya rusak, tolong saya, saya tak boleh bernafas, saya tak boleh berjalan.	Selamat pagi, selamat tengah hari, selamat petang, selamat malam, selamat tinggal, tumpang lalu.

Ten phrases are also used to check the confusability of the system. In other words, some additional phrases are not included in the algorithm, but added in the test to see if the system will mix them with the 80 included phrases. For example, if the speaker utters a word that is not included in the system and the system confuses it with an included phrase, this will be considered in determining the accuracy of the system.

5. Result and Discussion

The evaluation of data was performed using the confusion matrix which enabled us to get more detailed analysis. The parameters of confusion matrix are as follows:

- True Positive (TP) The correctly recognized words or phrases.
- True Negative (TN) The unconfused words or phrases.
- False Positive (FP) The confused words or phrases.
- False Negative (FN) The unrecognized (missing) words or phrases.

TP means that the words or phrases are predefined in the speech recognition systems and the system recognized them correctly whilst the speaker is saying them. For instance, if the speaker says" Saya sakit", and the recognized phrase is the same ("Saya sakit"), this will be True Positive (TP). However, if the system does not recognize the phrase, this will be False Negative (FN).

TN means that the words the speakers are saying are not predefined in the speech recognition system and the system is not confused with the predefined words or phrases. That is, if the speaker says "apa kabar", for example, the phrase will not be mixed up with other defined phrases. However, if the system mixes it up with a predefined phrase "Saya sakit", for example, this will be a False Positive (FP).

Hence, in order to achieve satisfactory results, the number of TP and TN has to be as high as possible as opposed to FP and FN which are better to be low. Because, the lower the numbers of FP and FN are, the more accurate the system is. Table 2 shows the overall tested data and table 3 shows the Confusion Matrix therefrom the accuracy of the system is calculated.

Table 2: The Overall Testing Data

Speaker	TP	FN	Sensitivity	TN	FP	Specificity
1	372	28	0.93	43	7	0.86
2	375	25	0.9375	43	7	0.86
3	379	21	0.9475	43	7	0.86
4	371	29	0.9275	39	11	0.78
5	377	23	0.9425	40	10	0.8
6	375	25	0.9375	43	7	0.86
7	365	35	0.9125	46	4	0.92
8	376	24	0.94	41	9	0.82
9	380	20	0.95	42	8	0.84
10	380	20	0.95	41	9	0.82
Total	**3750**	**250**	0.9375	421	79	0.842

Table 3: Confusion Matrix

	Sensitivity		**Specificity**
TP	**3750**	**FP**	**79**
FN	**250**	**TN**	**421**
	0.9375		0.842

By referring to the Confusion Matrix, the acccuracy of the system is determined as in the next formula.

$$\text{Accuracy} = \frac{\text{TP+TN}}{TP+FP+TN+FN}$$

$$\text{Accuracy} = \frac{3750+421}{3750+79+421+250} = 92.69\ \%$$

The Sensitivity and Specificity curve is shown in figure 4.

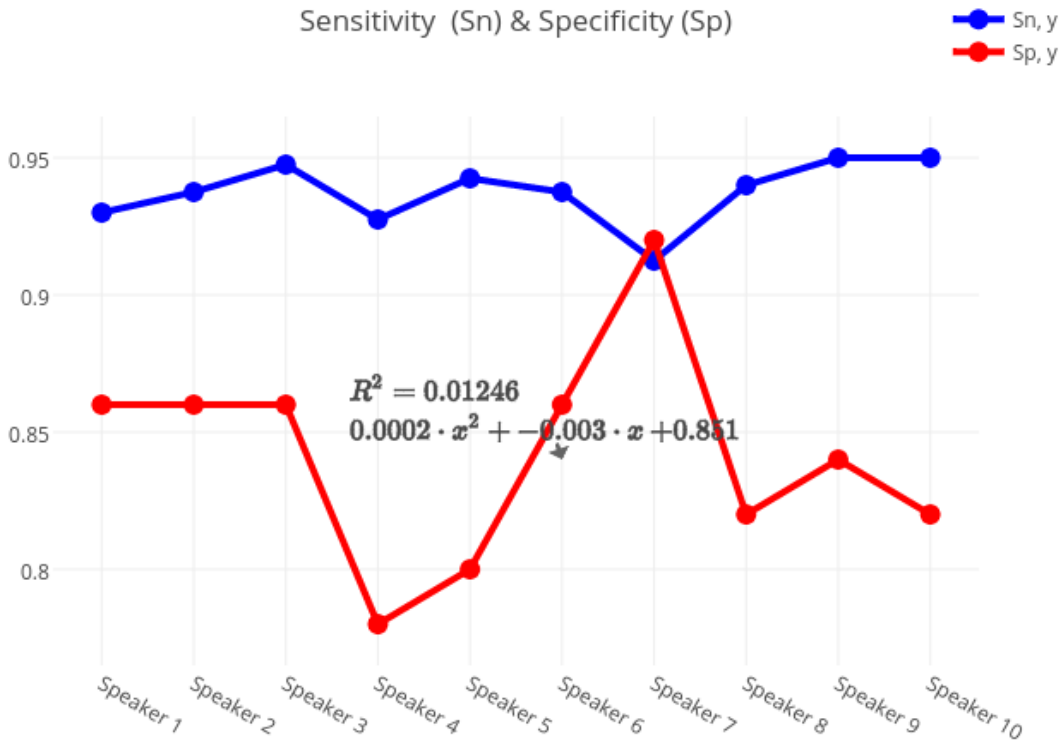

Figure 4. The Sensivity and Specivicity Curve

It can be percieved (from the result) that our system achieved a high accuracy in comparison to previously proposed ones. It can also be noted that the system can easily adapt to future edition as the phrases can be manually added into the algorithm without making recording audios and training them. That is, once a developer, or even a user with basic programming skills, desires to add to some phrases or to replace the existing ones, all he or she can do is writing the phrases literally into the code in the same wording as if he or she is writing in notepad. Therefore, this will facilitate the integration with other platforms such as Arduino and Raspberry Pi for control purposes; for example, when the users wish to control an automobile robot using vocal instructions, they can only add the corresponding word into the code such as, “forward”, “backward”, “left” or “right”. Then by the means of serial communication between Microsoft Visual Studio and Arduino the process will go smoothly. The adaptability of the speech system with control platforms is a significant property that enable us to choose Microsoft Visual Studio to be our platform for speech system.

6. Conclusion

In this work, a speech recognition system that translates the spoken Malay language into texts is presented. We used C# programming language and Microsoft Visual Studio as our platform which performs fast and accurate. The resulted accuracy is very authentic as we use the confusion matrix by adding extra 10 phrases in the test to check the confusability of the system. It is more satisfactory, though, as it is nearly 93%. The accuracy would have been higher than that, if a modern laptop computer, with a high-quality microphone, had been used.

An extension of the work in the future would be in applying the speech system in the control of external device such as a robot or any other automated machine; that will prove the adaptability of the system with other controlling platforms, such as Arduino, so that the features of the proposed speech system will be utilised.

References

[1] H. A. Maarif, R. Akmeliawati and Z. Z. Htike, "word classification for sign language synthesizer using hidden Markov model," in *The 5th International Conference on Information Information and Communication Technology for The Muslim World (ICT4M), 2014*, 2014.

[2] N. M. M. B. M. S. S. S. Feisal Dani Rahman, "Automatic Speech Recognition System for Malay," in *Third ICT International Student Project Conference (ICT-ISPC2014)* , 2014.

[3] M. Shahrul Azmi, "Malay Word Pronunciation Application for Pre-School Children using Vowel," 2015.

[4] T. T. S. a. S. H. S. Salleh, "Corpus-based Malay Text-to-Speech Synthesis System," in *2008 14th Asia-Pacific Conference on Communications* , Tokyo, 2007.

[5] S. S. A. H. K. I. a. A. N. S.A.R. Al-Haddad, "Robust Speech Recognition Using Fusion Techniques and Adaptive Filtering," *American Journal of Applied Sciences,* vol. 6, no. 2, pp. 290-295, 2009.