An Approach to accept input in Text Editor through voice and its Analysis, designing, development and implementation using Speech Recognition

Farhan Ali Surahio[1], Awais Khan Jumani[2] , Sawan Talpur[3]

Shah Abdul Latif Univesity of Khairpur Mirs, Sindh(Pakistan)[1]
Shah Abdul Latif Univesity of Khairpur Mirs, Sindh(Pakistan)[2]
Shah Abdul Latif Univesity of Khairpur Mirs, Sindh(Pakistan)[3]

ABSTRACT
Speech Recognition is one of the most incredible Technology, and it use to operate commands in computer via voice. Many applications have been using Speech Recognition for different purpose and 'Text Editor through voice' is one of them. Traditionally 'Text Editor through voice is based on experiencing the praxis using 'Hidden Markov Model' and application was designed in Visual Basic 6.0 and application was controlled by 'Speech Recognition' (Speaker independent) which translates input before finding specific words, phrases and sentences stored in database using Speech Recognition Engine. After finding and matching recognized input from database it puts that in document area of text. This paper presents analysis, designing, development and implementation of same 'Text Editor through voice' and approach is based on experiencing the praxis using 'Hidden Markov Model' and application is designed in Visual Basic.Net framework. We have added some new phrases and special characters in to existing application and designed extended Language Models and Grammar in Speech Recognition Engine. We illustrate you list of extended phrases, words in tables with figures that are effectively implemented and executed in our developed application.

Keywords
Markov Model, Neural Network, Language Model & Grammar, Speech Recognition Engine, Dynamic Time Warping and Graphical User Interface (GUI).

1. Introduction

Since 1930, it is difficult for scientist and engineers to make a system which respond appropriate, while given commands operating via voice. In 1930s, Homer Dudley of Bell Laboratories proposed a system model for speech investigate and synthesis [5], the problem of automatic speech recognition has been approached progressively, from a simple instrument that responds to a small set of sounds to a complicated system that responds to painless spoken natural language and takes into description the varying information of the language in which the speech is produced. Based on major advances in statistical modeling of speech in the 1980s, automatic speech recognition system today find extensive application in farm duties that require a human-machine interface.

Most of the applications are developed to perform some tasks in different organizations such applications are given below;

- **Playing back simple information:** In many circumstances customers do not actually need or want to speak to a live operator. For instance, if they have a little time or they have only require basic information then speech recognition can be used to cut waiting times and provide customers with the information they want.
- **Call Steering:** Putting callers through to the right department. Waiting in a queue to get through to an operator or, worse still, finally being put through to the wrong operator can be very frustrating to your customer, resulting in dissatisfaction. By introducing speech recognition, you can allow callers to choose a 'self-service' route or alternatively 'say' what they want and be directed to the correct department or individual.
- **Speech-to-text processing:** These types of applications are effectively takes audio content and transcribes it into written words in word processor or other display destination.
- **Voice user interface:** These kinds of application use to operate via voice command device to make a call and these applications fall into two major categories such as
  - Voice activated dialing.
  - Routing of Calls
- **Verification / identification:** These types of applications allows device manufacturer to define key phrases to wake up the so that it works out of the box for any user.

Speech recognition is the transformation of verbal inputs known as words, phrases or sentences into content. It is also known as 'Speech to Text', 'Computer Speech

Manuscript received March 5, 2016
Manuscript revised March 20, 2016

Recognition' or 'Automatic Speech Recognition'. It is one kind of technology and was first introduced by AT&T Bell Laboratories in the year 1930s.
Some speech based programs are allows to users for dictation on window desktop applications. For instance users speak something via microphone, then these program types same spoken words, sentences, phrases on the activated application window.
The speech recognition process is performed by a software component known as Speech recognition engine.

The initial function of the speech recognition engine is to process spoken user input and translate it into text that an application can understand.
Figure # 1 illustrates that Speech recognition engine requires two kinds of files to recognize speeches, which are described below.

1. **Language Model or Grammar**
2. **Acoustic Model**

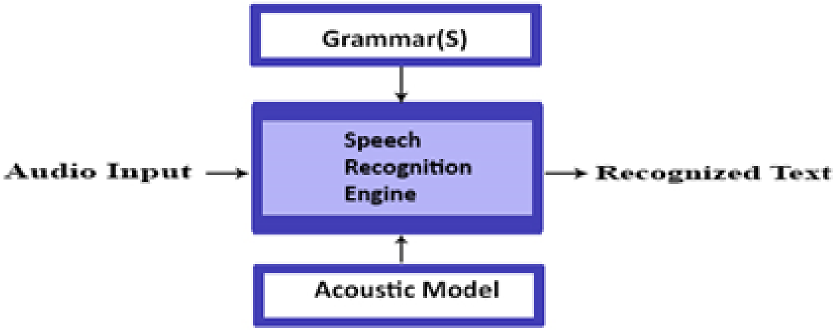


Figure 1: Speech Recognition Engine Component

1- Language Model or Grammar: A Language Model is a file containing the probabilities of sequence of words. A Grammar is a much smaller file containing set of predefined combination of words. Language Models are used for 'Dictation' applications, whereas Grammar are used as desktop 'Command and Control' applications.

2- Acoustic Model: Contains a statistical representation of the distinct sounds that make up each word in the language Model or Grammar. Each distinct sound corresponds to a phoneme. Speech Recognition Engine uses software that is called Decoder, which get the sounds spoken by a user and finds the acoustic Model for the same sounds, when a match is completed, the Decoder determines the phoneme corresponding to the sound. It keeps track of the matching phonemes until it reaches a pause in the users' speech. It then searches the Language Model or Grammar file for the same series of phonemes. If a match is made it returns the text of the corresponding word or phrase to the calling program.

2. Algorithms and Models

2.1. Dynamic Time Warping:

The Dynamic Time Warping (DTW) is an algorithm, it was introduced in 1960s [10]. It is an essential and ages algorithm was used in speech recognition System known as Dynamic Time Warping algorithm [7] [12] [14], it is used to measure the resemblances of objects/ sequences in the form of speed or time. For instance similarity would be detected in running pattern where in film one person was running slowly and other person was running fast. This algorithm can be applied to any data; even data is graphics, video or audio. It analyzes data by turning into a linear representation.
This algorithm is used in many areas: Computer Animation, Computer vision, data mining [13], online signature matching, signal processing [9], gesture recognition and speech recognition [2].

2.2. Hidden Markov Model

It is modern general purpose algorithm. It is widely used in speech recognition systems because of that statistical models are used by this algorithm, which creates output in the form of series of quantities or symbols. It is based on statistical models that output a series of symbols or quantities [3].

2.3 Neural Networks

Neural Networks were created in the late 1980s. These were emerging and an attractive acoustic modeling approaches used in Automatic Speech Recognition (ASR). From the era the algorithms have been used in different speech based systems such as phoneme categorization [8]. These algorithms are attractive recognition models for speech recognition because they formulate no assumptions as compares to Hidden Markov Models regarding feature statistical properties. This algorithm is used as preprocessing i.e; dimensionality reduction [6] and feature transformation for Hidden Markov Model based recognition [15], they have proposed four Language Models / Grammar which was implemented in Text Editor through voice. First was used for 'Command & Control' purpose and three were used for 'Dictation' purpose. Figure # 2 has shown the implemented of language model in speech recognition engine.

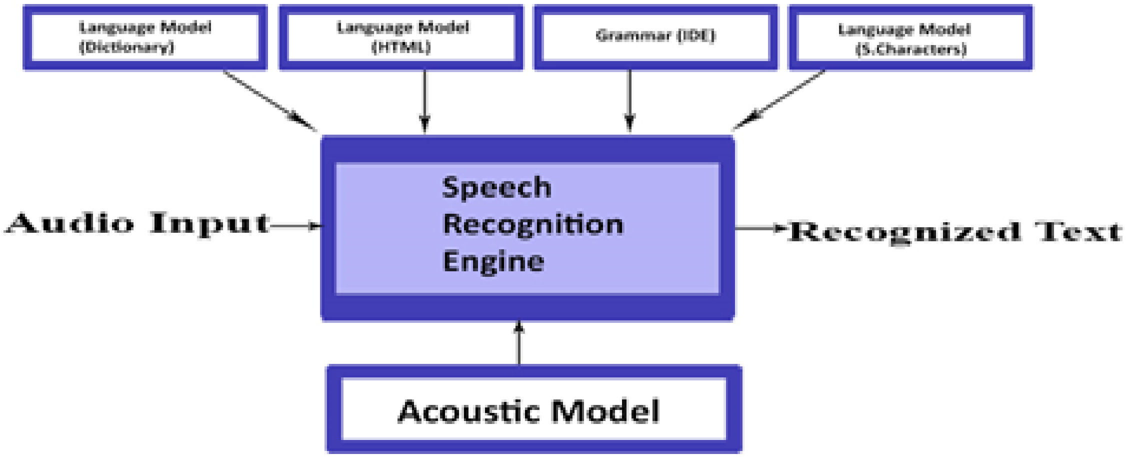

Language Model (Dictionary)
Language Model (HTML)
Grammar (IDE)
Language Model (S.Characters)
Audio Input
Speech Recognition Engine
Recognized Text
Acoustic Model

Figure 2: Implemented of Language Models & Grammar

They have proposed three models for dictation and used 33 phrases in HTML model, 34 Grammar for command control
Purpose, 38 special characters, numbers for Language Model dictation purpose.

3. Proposed Work

The research is determined on the five language models / grammars, which are implemented in Text Editor through voice. Those models / grammars are;

1) Dictionary
2) HTML (Hypertext Markup Language)
3) PHP (Hypertext Preprocessor)
4) IDE (Integrated Development Environment)
5) Special Character(S. Characters)

From these four language models / grammars, one is used as 'Command & Control' purpose other four used for 'Dictation' purpose. Their classification is given in figure # 3.

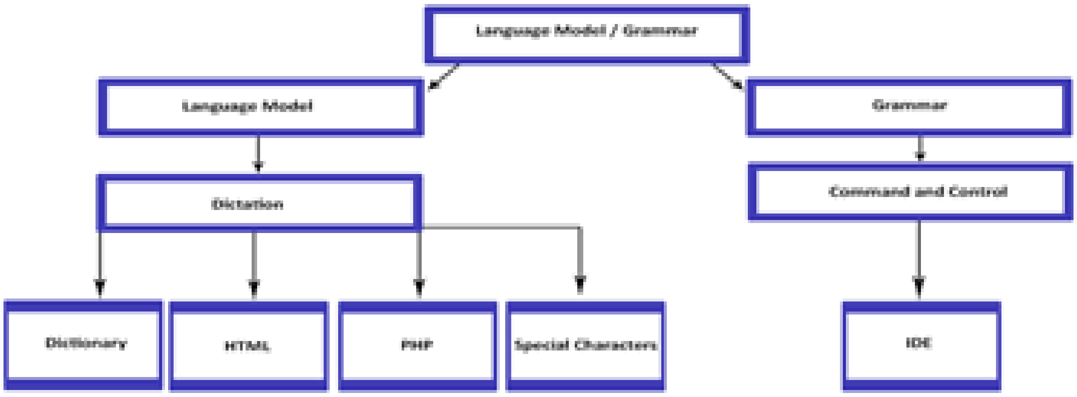


Figure 3: Classification of Language Model / Grammar

4. Achievement of Programmed Language Models & Grammar

As discussed earlier in introduction section that Speech Recognition Engine requires two kinds of files to recognize inputs. First is the Language/Grammar model and second is acoustic model. So we have created four langue models and one grammar in the Figure # 4 we have shown the implementation of language models and grammar model in speech recognition engine.

5. Application Pictures and Results

Figure # 5 GUI (Graphical User interface) of our designed application. In the left side of application we have give five MIC icons which perform functions in order to use and analyze language models grammar.

5.1. Dictionary

This language model use for dictation purpose where a user can insert and use word, phrases and sentences in current document 12000 words are stored in dictionary database. Figure #6 illustrates identifying some words and letters in current document area which are added by speaking using MIC.

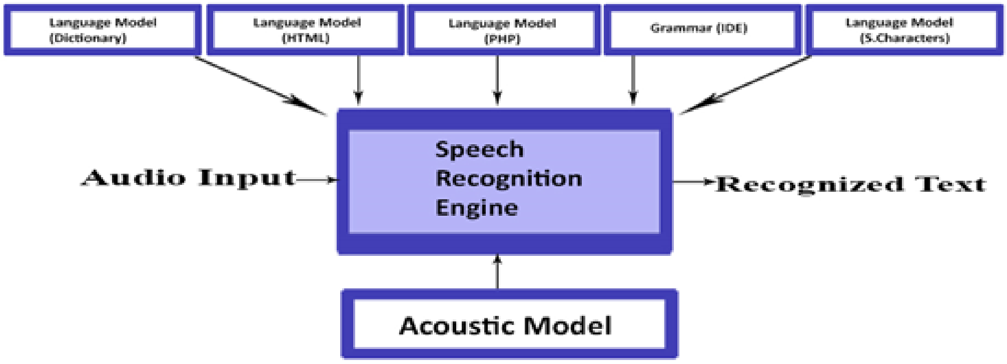

Language Model (Dictionary)
Language Model (HTML)
Language Model (PHP)
Grammar (IDE)
Language Model (S.Characters)
Speech Recognition Engine
Audio Input
Recognized Text
Acoustic Model

Figure 4: (Implementation of proposed Language Models & Grammar)

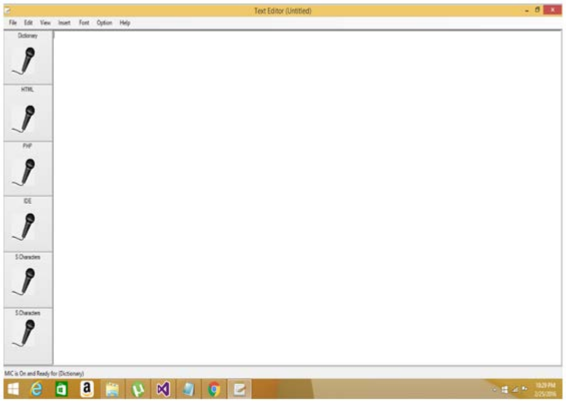
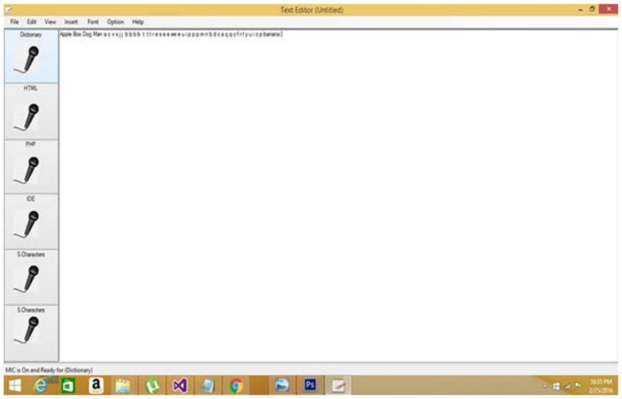


Figure 5: (Text Editor through Voice) Active Editor Window



Apple Box Dog Man a c v x j j b b b b t t t r e s e e e ee e u i p p p m n b d z a q q c f r t y u i o p banana

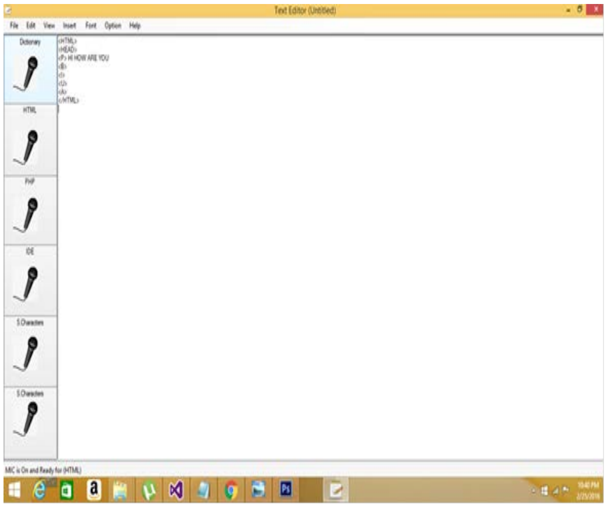
Figure 6: (Current Active Editor Window) using MIC (Dictionary is Functioning

5.2 HTML

This Language Model used to create web script based with extended Phrases on dictation. Words and Phrases for their relating HTML Tags are given in table no: 1 and Figure # 7 illustrates created web script speaking their phrases using MIC.

Table No 1: (List of extended Phrases and HTML Tags)

Phrases	Opening Tags	Phrases	Closing Tags
HTML	<HTML>	Close HTML	</HTML>
HEAD	<HEAD>	Close HEAD	</HEAD>
TITLE	<TITLE>	Close TITLE	</TITLE>
Body	<Body>	Close Body	</Body>
Image	<Image>	---	---
Anchor	<A>	Close A	</A>
B	<B>	Close B	</B>
I	<I>	Close I	</I>
U	<U>	Close U	</U>
Center	<Center>	Close Center	</Center>
Font	<Font>	Close Font	</Font>
HR	<HR>	Close HR	</HR>
BR	<BR>	Close BR	</BR>
P	<P>	Close P	</P>
Table	<Table>	Close Table	</Table>
TH	<TH>	Close TH	</TH>
TR	<TR>	Close TR	</TR>
TD	<TD>	Close TD	</TD>
H1	<H1>	Close H1	</H1>
H2	<H2>	Close H2	</H2>
H3	<H3>	Close H3	</H3>
H4	<H4>	Close H4	</H4>
H5	<H5>	Close H5	</H5>
H6	<H6>	Close H6	</H6>
Sub	_{	Close Sub	}
Sup	^{	Close Sup	}
Marquee	<Marquee>	Close Marquee	</Marquee>
Frame	<Frame>	Close Frame	</Frame>
Frameset	<Frameset>	Close Frameset	</Frameset>
Form	<Form>	Close Form	</Form>
Input	<Input>	---	---
Select	<Select>	Close Select	</Select>
Option	<Option>	---	---
Text Area	<Textarea>	Close Text Area	</Textarea>



```
<HTML>
<HEAD>
<P> HI HOW ARE YOU
<B>
<I>
<U>
<A>
</HTML>
```

MIC is On and Ready for (HTML)

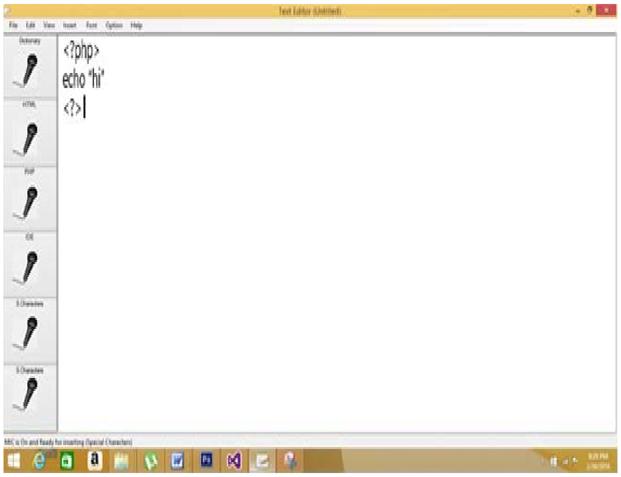
Figure 7: (Testing HTML functions using MIC)

5.3. PHP

This Language Model is used to create a simple web testing page based on dictation. Words and Phrases for their relating PHP tags are given in table no: 2 and Figure # 8 illustrates simple web script using MIC.

Table No 2: (List of Phrases and PHP Tags)

Phrases	Opening Tags	Phrases	Closing Tags
PHP	<?PHP>	Close PHP	<?>
ECHO	ECHO	---	---



```
<?php>
echo "hi"
<?>
```

MIC is On and Ready for inserting Special Characters

Figure 8: (Testing PHP functions using MIC)

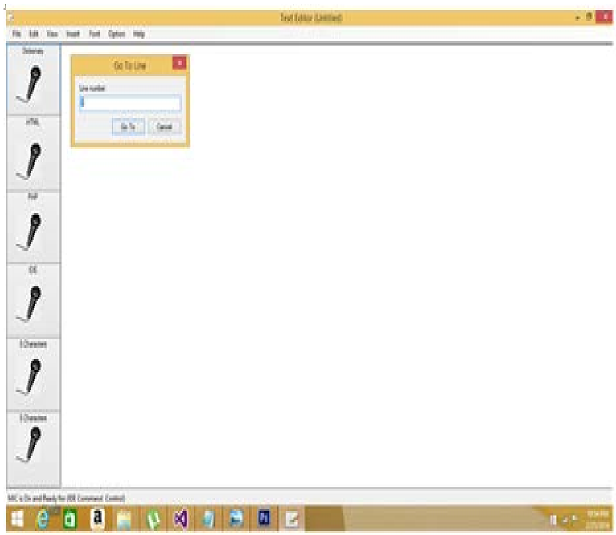
5.4. IDE

This grammar based on command and control purpose phrases and their description are given in table no: 3 Figure no: 9 illustrates go to function is called by speaking using MIC.

Table No 3: (IDE control list of Phrases)

List of Phrases	Description of Phrase
New	To Open new document
Open	To Open saved document
Save	To Save Document
Save As	To Save document with new name
Print	To Print document
Exit	To Exit Text Editor
Delete	To Delete selected text
Cut	To Cut selected text
Copy	To Copy selected text
Paste	To place cut or copied text
Find	To Search text from document

Replace	To Replace document
Go To	Go To required line number
Select All	To Select All Text
Time	To Insert time in document
Tool Bar	To Call tool bar function
Status Bar	To Call status bar function
Standard Buttons	To Call standard buttons function
Date and Time	To Insert date and time in document
Bold	To change the format of text as Bold
Italic	To change the format of text as Italic
Underline	To change the format of text as underline
Font	To Call font function
Color	To Call color function
Dictionary	To call Dictionary function
HTML	To call HTML function
IDE	To call IDE function
Special Characters	To call special character function
Database	To call database wizard function
De Activate	To Off MIC
ital Characters	To call capital character
Small Characters	To call small character function
About Me	To know about Application Developer
About Project	To know about Project Description
Contents	Help and Index



MIC is On and Ready for IDE Command Control

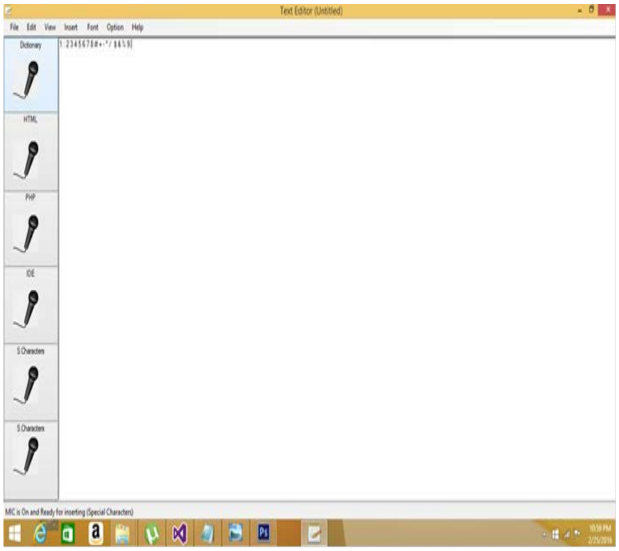
Figure 9: (IDE GO TO Function is selected using MIC)

5.5. Special Characters

This Language Model provides users to insert special characters and numbers in to current active document for dictation purpose. Phrases and descriptions are given in table no: 4 and Figure # 10 illustrates special characters and numbers in current document using MIC.

Table No 4: (Special Characters and description)

List of Phrases	**Description**
Less than	To insert (<) sign in document
Greater than	To insert (>) sign in document
Dot	To insert (.) sign in document
Comma	To insert (,) sign in document
Colon	To insert (;) sign in document
Semi colon	To insert (:) sign in document
Single quote	To insert (‘) sign in document
Double quote	To insert (“) sign in document
Question mark	To insert (?) sign in document
Steric	To insert (*) sign in document
And	To insert (&) sign in document
Percent	To insert (%) sign in document
Slash	To insert (/) sign in document
Back slash	To insert (\) sign in document
Hash	To insert (#) sign in document
Dollar	To insert ($) sign in document
Dash	To insert (-) sign in document
Underscore	To insert (_) sign in document
Exclamation	To insert (!) sign in document
Addition	To insert (+) sign in document
Subtraction	To insert (-) sign in document
Multiplication	To insert(*) sign in document
Division	To insert(/) sign in document
Zero	To Insert (0) sign in document
One	To insert (1) sign in document
Two	To insert (2) sign in document
Three	To insert (3) sign in document
Four	To insert (4) sign in document
Five	To insert (5) sign in document
Six	To insert (6) sign in document
Seven	To insert (7) sign in document
Eight	To insert (8) sign in document
Nine	To insert (9) sign in document
Back	To call the function of (back space) key
Insert	To call the function of(insert) key
Delete	To call the function of(delete) key
Home	To call the function of (home) key
End	To call the function of (end) key
Page up	To call the function of (page up) key
Page down	To call the function of (page down) key



1:2345678#+-*/ $&%9

MIC is On and Ready for inserting (Special Characters)

Figure 10: (Special Character Functioning using MIC)

6. Conclusion

It is studied that implemented traditional 'Text Editor through voice' have four models and one Grammar and it was based on experiencing the praxis using 'Hidden Markov Model' and application was developed in Visual Basic. They have used 33 phrases in HTML model, 34 Grammar for command control purpose, 38 special characters, and numbers for Language Model dictation purpose.

In our proposed work we have added anchor tag in HTML which enables users to link one page to another and also provided two major arithmetic functions multiplication and division in special characters and introduced one new model named PHP based on experiencing the praxis using 'Hidden Markov Model' and application was developed in Visual Basic.Net framework 'Text Editor through voice' via Speech Recognition Technology and it is working properly. This mini research has needed more improvements for successful commercial product. For instance:

- This application is not able to get input from other languages except English. There is need to develop same editor for Sindhi and Urdu Languages.
- This application needs to Environment having no noise.
- This application needs to accept variables and more functions for PHP.

References

[1] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, (1989) "Phoneme recognition using time-delay neural networks," IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 37, pp. 328-339.

[2] C. Myers, L. Rabiner, and A. Rosenberg, \Performance tradeo_s in dynamic time warping algorithms for isolated word recognition," Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing], IEEE Transactions on, vol. 28, no. 6, pp. 623{635,1980.

[3] Goel, V.; Byrne, W. J. (2000). "Minimum Bayesrisk automatic speech recognition". Computer Speech & Language 14 (2): 115– 135. doi:10.1006/csla.2000.0138. Retrieved 2011-03-28.

[4] H. Dudley, The Vocoder, Bell Labs Record, Vol. 17, pp. 122-126, 1939.

[5] H. Dudley, R. R. Riesz, and S. A. Watkins, A Synthetic Speaker, J. Franklin Institute, Vol. 227, pp. 739-764, 1939

[6] Hongbing Hu, Stephen A. Zahorian, (2010) "Dimensionality Reduction Methods for HMM Phonetic Recognition," ICASSP 2010, Dallas, TX

[7] Itakura, F. (1975). Minimum Prediction Residual Principle Applied to Speech Recognition. IEEE Trans. on Acoustics, Speech, and Signal Processing, 23(1):67-72, February 1975. Reprinted in Waibel and Lee (1990).

[8] J. Wu and C. Chan,(1993) "Isolated Word Recognition by Neural Network Models with Cross-Correlation Coefficients for Speech Dynamics," IEEE Trans. Pattern Anal. Mach. Intell., vol. 15, pp. 1174-1185.

[9] M. Muller, H. Mattes, and F. Kurth, \An e_cient multiscale approach to audio synchronization," pp. 192{197, 2006.

[10] R. Bellman and R. Kalaba, \On adaptive control processes,"Automatic Control, IRE Transactions on, vol. 4, no. 2, pp. 1{9,1959.

[11] S. A. Zahorian, A. M. Zimmer, and F. Meng, (2002) "Vowel Classification for Computer based Visual Feedback for Speech Training for the Hearing Impaired," in ICSLP 2002.

[12] Sakoe, H. and Chiba, S. (1978). Dynamic Programming Algorithm Optimization for Spoken Word Recognition. IEEE Trans. on Acoustics, Speech, and Signal Processing, 26(1):43-49, February 1978. Reprinted in Waibel and Lee (1990).

[13] V. Niennattrakul and C. A. Ratanamahatana,"On clustering multimedia time series data using k-means and dynamic time warping," in Multimedia and Ubiquitous Engineering, 2007. MUE '07. International Conference on, 2007, pp. 733{738.

[14] Vintsyuk, T. (1971). Element-Wise Recognition of Continuous Speech Composed of Words from a Specified Dictionary. Kibernetika 7:133-143, March-April 1971.

[15] Nadeem.A.K, Habibullah.U.A, A.Ghafoor.M, Mujeeb-U-Rehman.M and Kamran.T.P. "Speech Recognition in Context of predefined words, Phrases and Sentences stored in database and its analysis, designing, development and implementation in an Application". International Journal of Advance in Computer Science and Technology, vol.2, No 12, pp .256-266, December 2013.