International Journal of Engineering Research and General Science Volume 4, Issue 2, March-April, 2016
ISSN 2091-2730

An Intelligent System Control Using Speech Recognition

Girish. S, Anooplal. K. S

Abstract— Speech recognition technology has refashion the way people with disabilities to use computers. This raising technology accelerates more opportunities for medical prescriptions, education, employment etc. Speech recognition is an alternative to typing on a keyboard. Just talk to the computer and our words appear on the screen. The software has been developed to administer a fast approach of writing onto a computer and can help people with a variety of disabilities. It is useful for people with physically challenged who often find typing difficult, painful or impossible. Voice recognition software further help those who with spelling adversity, including dyslexic users, because recognized words are always correctly spelled.

Speech recognition technology helps people with disabilities to interact with computers more easily and support doctors to make prescription easily. Using this speech recognition we create a system which acts according to the voice commands given by the user. People with motor limitations, which cannot use a standard keyboard and mouse, can use their voices to navigate the computer and create documents. The technology is further convenient to people with learning disabilities that experience dilemma with spelling and writing. Some individuals with speech impairments may practice speech recognition as a therapeutic tool to progress vocal quality. Speech recognition technology has great potential to provide people with disabilities greater access to computers and a world of opportunities.

Keywords — Speech recognition; Intelligent System.

INTRODUCTION

Research in automatic speech recognition by machine has been done for almost five decades. It is worthwhile to briefly review some research highlights. The earliest attempts to devise systems for automatic speech recognition by machine were made in the 1950s, when various researchers tried to exploit the fundamental ideas of acoustic-phonetics. In 1952, at Bell Laboratories, Davis, Biddulph and Balashek built a system for isolated digit recognition for a single speaker [1]. The system relied heavily on measuring spectral resonances during the vowel region of each digit. In an independent effort at RCA Laboratories in 1956, Olson and Belar tried to recognize 10 distinct syllables of a single talker, as embodied in 10 monosyllabic words [2].

The system again relied on spectral measurements (as provided by an analog filter bank) primarily during vowel regions. In 1959, at University College in England, Fry and Denes tried to build a phoneme recognizer to recognize four vowels and nine constants [3]. They used a spectrum analyzer and pattern matcher to make the recognition decision. A novel aspect of this research was the use of statistical information about allowable sequences of phonemes in English (a rudimentary form of language syntax) to improve overall phoneme accuracy for words consisting of two or more phonemes. Another effort of note in this period was the vowel recognizer of Forgie and Forgie, constructed at MIT Lincon Laboratories in 1959, in which 10 vowels embedded in a/b/-vowel-/t/ format were recognized in speaker independent manner [4]. Again a filter bank analyzer was used to provide spectral information, and a time varying estimate of the vocal tract resonances was made to decide which vowel was spoken.

In the 1960s several fundamental ideas in speech recognition surfaced and were published. However, the decade started with several Japanese laboratories entering the recognition arena and building special purpose hardware as part of their systems. One early Japanese system, described by Suzuki and Nakata of the Radio Research Lab in Tokyo [5], was a hardware vowel recognizer. An elaborate filter bank spectrum analyzer was used along with logic that connected the outputs of each channel of the spectrum analyzer (in a weighted manner) to a vowel-decision circuit, and a majority decision logic scheme was used to choose the spoken vowel. Another hardware effort in Japan was the work of Sakai and Doshita of Kyoto University in 1962, who built a hardware phoneme recognizer [6]. A hardware speech segmented was used along with a zero crossing analysis of different regions of the spoken input to provide the recognition output. A third Japanese effort was the digit recognizer hardware of Nagata and coworkers at NEC Laboratories in 1963 [7]. This effort was perhaps most notable as the initial attempt at speech recognition at NEC and led to a long and highly productive research program.

In the 1960s three key research projects were initiated that have had major implications on the research and development of speech recognition for the past 20 years. The first of these projects was the efforts of Martin and his colleagues at RCA Laboratories, beginning in the late 1960s, to develop realistic solutions to the problems associated with non-uniformity of time scales in speech

events. Martin developed a set of elementary Time-normalization methods, based on the ability to reliably detect speech starts and ends that significantly reduced the variability of the recognition scores [8]. Martin ultimately developed the method and founded one of the first companies, Threshold Technology, which built, marketed, and sold speech-recognition products, At about the same time in the Soviet Union, Vintsyuk proposed the use of dynamic programming methods for time aligning a pair of speech utterances[9]. Although the essence of the concepts of dynamic time warping, as well as rudimentary versions of the algorithms for connected word recognition, were embodied in Vintsyuk's work, it was largely unknown in the West and did not come to light until the early 1980s; this was long after the more formal methods were proposed and implemented by others. A final achievement of note in the 1960s was the pioneering research of Reddy in the field of continuous speech recognition by dynamic tracking of phonemes [10], Reddy's research eventually spawned a long and highly successful speech-recognition research program at Carnegie Mellon University (to which Reddy moved in the late 1960s) which, to this day, remains a world leader in continuous-speech-recognition systems.

In the 1970s speech-recognition research achieved a number of significant milestones. First the area of isolated word or discrete utterance recognition become a viable and usable technology based on fundamental studies by Velichko and Zagoruyko in Russia [11], Sakoe and Chiba in Japan [12], and Itakura in the United States [13]. The Russian studies helped advance the use of pattern-recognition ideas in speech recognition. The Japanese research showed how dynamic programming methods could be successfully applied; and Itakura's research showed how the ideas of linear predictive coding (LPC), which had already been successfully used in low-bit-rate speech coding, could be extended to speech recognition systems through the use of an appropriate distance measure based on LPC spectral parameters. Another milestone of the 1970s was the beginning of a longstanding, highly successful, group effort in large vocabulary speech recognition at IBM in which researchers studied three distinct tasks over a period of almost two decades, namely the New Raleigh language [14] for simple database queries, the laser patent text language [15] for transcribing laser patents, and the office correspondence task, called Tangora [16], for dictation of simple memos. Finally, at AT&T Bell Labs, reasearchers began a series of experiments aimed at making speech-recognition systems that were truly speaker independent [17]. To achieve this goal a wide range of sophisticated clustering algorithms were used to determine the number of distinct patterns required to represent all variations of different words across a wide user population. This research has been refined over a decade so that the techniques for creating speaker-independent patterns are now well understood and widely used.

Just as isolated word recognition was a key focus of research in the 1970s, the problem of connected word recognition was a focus of research in the 1980s. Here the goal was to create a robust system capable of recognizing a fluently spoken string of words (e.g. Digits) based on matching a concatenated pattern of individual words. A wide variety of connected word-recognition algorithms were formulated and implemented, including the two-level dynamic programming approach of Sakoe at Nippon Electric Corporation (NEC) [18], the one-pass method of Bridle and Brown at Joint Speech Research Unit (JSRU) in England [19], the level building approach of Myers and Rabiners at Bell Labs [20], and the frame synchronous level building approach of Lee and Rabiner at Bell Labs [21]. Each of these "optimal" matching procedures had its own implementation advantages, which were exploited for a wide range of tasks Speech research in the 1980s was characterized by a shift in technology from template-based approaches to statistical modeling methods-especially the hidden Markov model approach [22, 23]. Although the methodology of Hidden Markov Modeling (HMM) was well known and understood in a few laboratories (primarily IBM, Institute for Defense Analyses (IDA), and Dragon Systems), it was not until widespread publications of the methods and theory of HMMs, in the mid-1980, that the technique became widely applied in virtually every speech-recognition research laboratory in the world.

Another "new" technology that was reintroduced in the late 1980s was the idea of applying neural networks to problems in speech recognition. Neural networks were first introduced in the 1950s, but they did not prove useful initially because they had many practical problems. In the 1980s, however, a deeper understanding of the strengths and limitations of the technology was obtained, as well as the relationships of the technology to classical signal classification methods. Several new ways of implementing systems were also proposed [24, 25]. Finally, the 1980s was a decade in which a major impetus was given to large vocabulary, continuous-speech-recognition systems by the Defense Advanced Research Projects Agency (DARPA) community, which sponsored a large research program aimed at achieving high word accuracy for 1000-words, continuous-speech-recognition, and database management task. Major research contributions resulted from efforts at CMU (Notably the well-known SPHINX system) [26], BBN with the BYBLOS system [27], Lincoln Labs [28], SRI [29], MIT [30], and AT&T Bell Labs. The DARPA program has continued into the 1990s, with emphasis shifting to natural language front ends to the recognizer, and the task shifting to retrieval of air travel information. At the same time, speech-recognition technology has been increasingly used within telephone networks to automate as well as enhance operator services.

SPEECH RECOGNITION

The primary objective of this paper is to allow the user to interact with the computer through voice commands and let the user to control computer functions and dictate text by voice. For example, a person can scroll down the web page with a voice command, such as "Scroll down" control application functions, also performing various operations such as opening programs, web browser, notepad, search, documents, run, help, undo, etc. and also perform cut, copy and paste operations. Now doctors have to write prescription in capital letters so speech recognition supports doctors for medical prescription easily. More and more people with special needs are considering speech recognition as an alternate method for computer access because speech recognition products are more affordable and user-friendly than ever before and these applications needs less initial training than their predecessors and typically offers much improved accuracy. The system operations are to be performed according to the voice commands given by the user and it is created as an agent. This agent gets the voice commands from the user and performs the system functions. I.e. the agent acts as an interface between the user and the system. The commands which will be given by the user are separately stored in an xml file. This file helps the new user to understand the commands that he can execute in the system.

Speech recognition software has to be trained to recognize and understand the user's voice. The user has to train the software by reading a selection of paragraphs to the computer. By analyzing the voice, the computer creates a unique voice file for that particular user. For successful voice recognition, an appropriate training of the system is necessary. For this purpose we have used the software tool SDK 5.1 which converts voice to text and also add a vocabulary to the system to match or correct spellings while reading. A speech-enabled application and an SR engine (speech recognition engine) do not directly communicate with each other-all communications are done by using SAPI (Speech Application Programming Interface). Controlling audio input, from a microphone, or from other sources, converting audio data in to a valid engine format, loading grammar files, resolving grammar imports and grammar editing, compiling standard SAPI XML grammar format, conversion of custom grammar formats, parsing semantic tags in results, sharing of recognition across multiple applications, marshaling between engine and applications are done by SAPI. SAPI also manage other aspects like feedback the results and other information to the application, storing audio and serializing results for later analysis. SAPI ensures that applications do not cause errors, prevents applications from calling the engine with invalid parameters and further dealing with crashing of applications. The SR engine Performs recognition, uses SAPI grammar interfaces, loads dictation, generates, recognition and other events to provide information to the application.

FUNCTIONAL OVERVIEW

Most speech recognition software arrives bundled with noise canceling microphone. Placing an appropriate and consistent positioning of the microphone is indispensable for good recognition. It is also helpful to adjust the audio levels before each session to accommodate for everyday fluctuation in environmental cacophony and divergence in microphone positioning. The sound card is a critical factor of a speech recognition system. Recognition problems may be the result of low quality sound card performance or incompatibility between the soundcard and the voice recognition software. Generally speech recognition programs contain a utility program that assesses the quality of the soundcard. If the computer's soundcard is deficient, the intended user will need to get a vendor-approved soundcard. Voice recognition software vendors commonly provide a permitted list of soundcard's with their technical specifications. Laptop computer noise may interfere with quality sound processing using a USB microphone, which bypasses the internal soundcard, may resolve this issue.

Speak in a consistent level emphasis, speaking too vehemently or too tenderly makes it laborious for the computer to recognize what you've said. Practice a consistent rate, without speeding up and slowing down. Speak without pausing between words; a phrase is easier for the computer to interpret than just one word. For example, the computer has a hard time understanding phrases such as, "This (pause) is (pause) another (pause) example (pause) sentence". Start by working in a quiet environment so that the computer hears you instead of the sounds around you, and use a good quality microphone. Try to keep the microphone in the same position do not change once it is adjusted or use microphone head set. Train your computer to recognize your voice by reading clearly the prepared training text in the Voice Training Wizard. Additional training increases speech recognition accuracy. As you dictate, do not be anxious if you do not rapidly examine your words on the screen. Continue speaking and pause at the end of your thought. The computer will display the recognized text on the screen after it finishes processing your voice. Pronounce words clearly, but do not separate each syllable in a word. For example, sounding out each syllable in "e-nun-chi-ate," will make it harder for the computer to recognize what you've said.

Our paper aims at creating an agent which acts as an interface between user and the system. The system operations are performed according to the voice commands given by the user. An agent is used in order to make the user aware of what commands he will be

giving. The commands which will be given by the user are separately stored in an xml file. Using this store of commands new user comes to know about the commands which he can execute in the system. Speech recognition software is "speaker dependent," which means that it must be trained to recognize and understand the user's voice. The user trains the software by reading a selection of paragraphs to the computer and computer analyzes the voice data then creates a unique voice file for each particular user. Thorough training of the system is very important stage for successful voice recognition. Most speech recognition systems require the user to complete an initial training session so that the software can learn the user's vocal style and speech patterns. However, additional training is usually necessary to improve recognition accuracy to a satisfactory level. Some people with disabilities require customized training, set up and technical support. There may be technical difficulties with the software or hardware. In the case of system failure, an alternative method should be readily available for use. Local voice recognition vendors may provide training and technical support packages for additional cost.

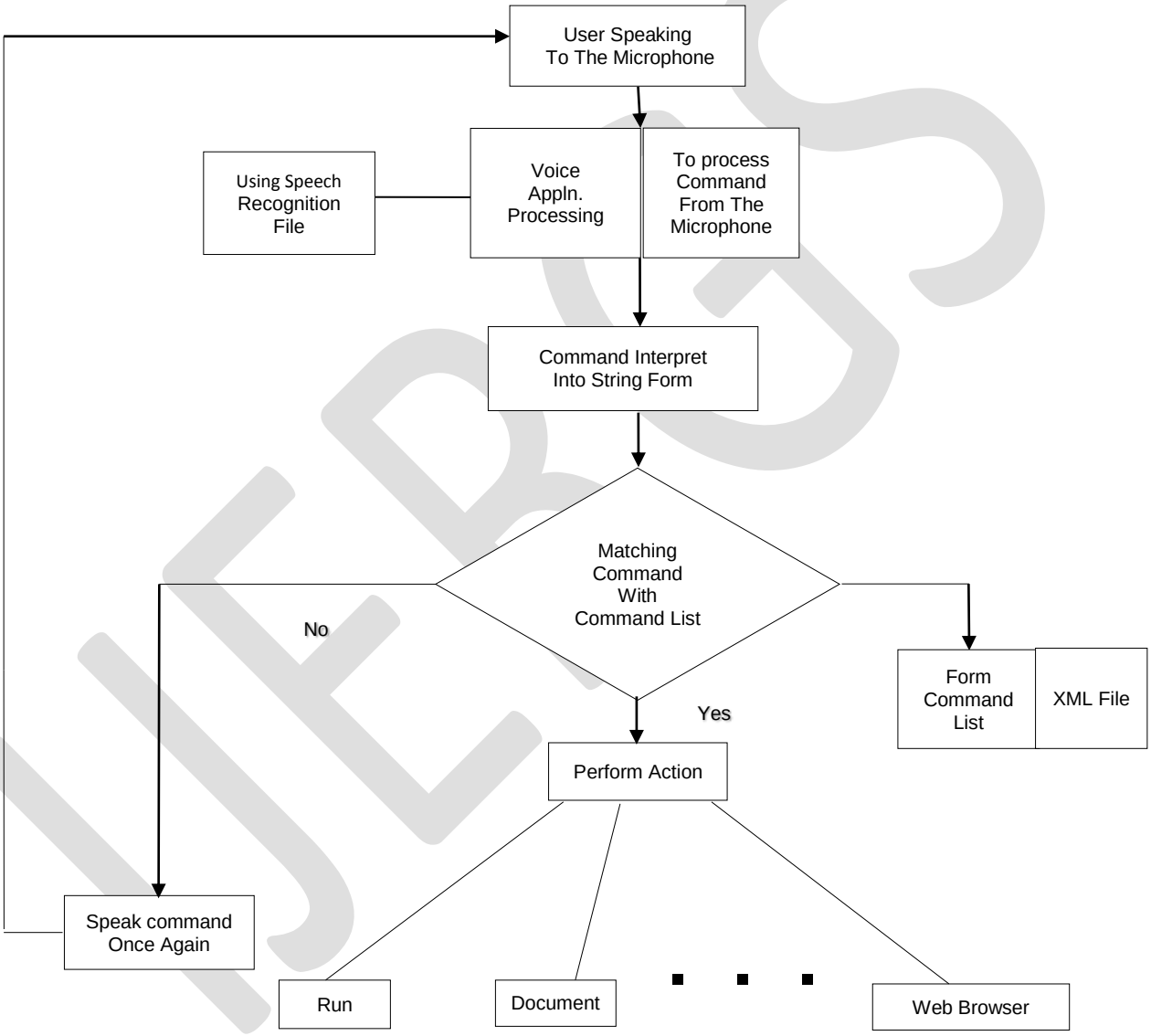

Figure. 1: General Approch of Speech Recognition.

IMPLEMENTATION

Here we proposed to write code by using a .NET-compatible language such as C Sharp. C Sharp can be used to create applications that will run in the .NET platform. We created various forms like formabout, formprofilechange, formaccuracychange, formfavorites, formaddfavorites, and formcommands in C sharp.net. Also we created various xml files like xmlactivate, xmlstart, xmldeactivate,

xmlstickykeys, xmlabout, xmlalphabeticstate, and xmlnumericstate. XML schema describes the SAPI 5.0 SR Command and Control grammar format which is based on the XML framework. The schema is included in the speech grammar compiler tool (gramcomp.exe or gc.exe). We can use the SDK components and redistributable SAPI/engine run-time to build applications that incorporate speech recognition and speech synthesis. Using this speech recognition we create a system which acts according to the voice commands given by the user.

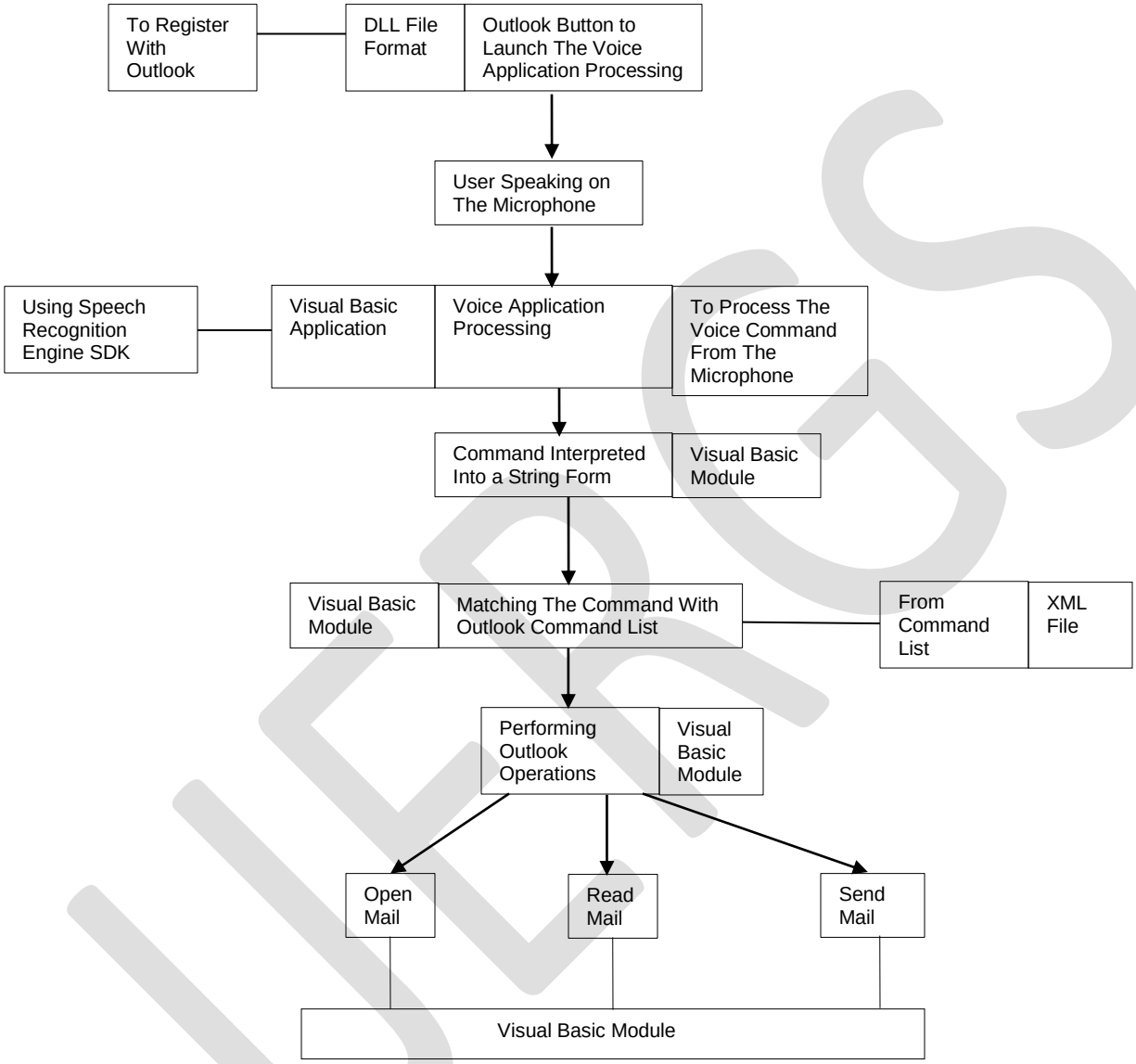

Figure. 2:Flow Chart.

formAccuracychange.cs: This form allows setting the accuracy with which user should speak. This form contains labels like accuracy limit, accuracy percent, a list box which helps to set the accuracy percent and buttons like ok which helps to accept the selected accuracy percent and cancel to close this form.

formAddfavorites.cs: Whenever user gives the command "Add favorites" this form will be loaded. This form allows user to add various applications into favorites form which can be opened with a single command. Here we are adding the program names along with the phrases which are used to open those programs. Here two list boxes namely program and phrase are used which contain the list of programs with corresponding phrases which can be opened. To add a new program with corresponding phrase two textboxes are used. Upon entering the new program and phrase name into the textbox, Add button helps to update the list boxes with newly entered program and phrase. Delete button is used to delete a program from the list box with the phrase. Browse button is used to add the

program to list box from various locations. Save button is used to update code for added or deleted program. Close button is used to close this form.

formCommands.cs: In this form the various commands for which this project works are included. Tree view of visual studio .NET to design this form. By this form user gets knowledge about commands which he can work with.

formFavorites.cs: Whenever user gives the command "form favorites" this form will be loaded. This form allows user to know about various applications which can be opened with a single command. Here two list boxes namely program and phrase are used which contain the list of programs with corresponding phrases which can be opened. Here user can open any program specified in the list boxes with the corresponding phrase as the voice command.

BIOGRAPHIES

[1.] **Mr. Girish. S** received B. E. in Electronics & Communication Engineering from VTU, Belagavi and M.Tech Degree in Networking & Internet Engineering from JNNCE, Shimoga. He is currently working as Assistant Professor in Computer Science & Engineering Department at Sahyadri College of Engineering & Management Mangaluru, India-575007. Email – giriait@gmail.com.

[2.] **Mr. Anooplal. K. S** received B. E. Degree in Information Science & Engineering (ISE) from VTU, Belagavi and M.Tech Degree in Computer Science & Engineering (CSE) from Sahyadri College of Engineering & Management Mangaluru, India, affiliated to Visvesvaraya Technological University (VTU) Belgavi. Email – anooplalks@gamil.com.

CONCLUSION

As an improving technology, not all developers are intimate with speech technology. While the essential accretions of this speech synthesis and speech recognition take only minutes to figure out, there are subtle and powerful capabilities provided by computerized speech.

An effective speech application is one that uses speech to enhance a user's performance of task or enable an activity. Designing an application with speech in mind from the outset is a key success factor. In future it may replace the keyboard and other peripherals.

Most importantly, speech technology does not consistently meet the immense assumptions of users familiar with formal human to human speech communication. Understanding the limitations–as well as the strengths-is important for effective use of speech input and output in a user interface. So here we present in our project a provision for user to open various applications such as Microsoft Word, Microsoft Excel, and Notepad just by saying a word. We can also open calculator and perform various operations in it.

REFERENCES:

1) K. H. Davis, R.Biddulph, and S. Balashek "Automatic Recognition of Spoken Digits", J. Acoust. Soc. Am.24 (6):637-642, 1952.
2) H. F. Olson and H.Belar, "Phonetic Type writer", J.Acoust.Soc.Am, 28(6):1072-1081, 1956.
3) B Fry, "Theoretical Aspects of Mechanical Speech Recognition"; and P. Denes, "The Design and Operation of the Mechanical Speech Recognizer at University College London," J. British Inst. Radio Engr., 19: 4, 211-229, 1959.
4) J. W. Forgie and C. D. Forgie, "Results Obtained From a Vowel Recogniton Computer Program, "J. Acoust. Soc.Am., 31(11): 1480-1489, 1959.
5) J. Suzuki and K. Nakata, "Recognition of Japanese Vowels-Preliminary to the Recognition of Speech," J. Radio Res. Lab,37(8): 193-212, 1961.
6) T. Sakai and S. Doshita, "The Phonetic Typewriter, Inforamation Procession 1962, "Proc. IFIP Congress, Munich, 1962.
7) K. Nagata, Y Kato, and S. Chiba, "Spoken Digit Recognizer for Japanese Language," NEC Res. Develop., No6, 1963.
8) T. B. Martin, A.L. Nelson, and H. J. Zadell, "Speech Recognition by Feature Abstratction Techniques, "Tech. Report ALT-TDR-64-176, Air Force Avionics Lab 1964.
9) T. K. Vintsyuk, "Speech Discrimination by Dynamic Programming," Kibernetika, 4(2); 81-88, Jan-Feb 1968.
10) R. Reddy, "An Approach to Computer Speech Recognition by Direct Analysis of the Speech Wave," Tech. Reprot No. C549, Computer Science Dept., Stanford Univ., September 1966.
11) M. Velichko and N. G. Zagoruyko "Automatic Recognition of 200 words",Int.J.Man-Machine Studies,2:223,June 1970.
12) H. Sakoe and S. Chiba, "Dynamic Programming Algorithm Optimization for Spoken word Recognition," IEEE Trans. Acoustics, Speech, Signal Proc.., and ASSP-26 (1): 43-49, February 1978.
13) Itakura "Minimum Prediction Residual Applied to Speech Recognition", IEEE Trans. Acoustics, Speech, Signal Proc., ASSP-23(1):67-72, February 1975.

14) C. Tappert, N. R. Dixon, A. S. Rabinowitz, and W .D.Chapman, "Automatic Recognition Of Continuous Speech Utilizing Dynamic Segmentation, Dual Classification, Sequential Decoding and Error Recovery," Rome Air Dev.Cen, Rome, NY, Tech.Report TR-71-146, 1971.
15) Jelinek, L. R. Bahl, and R. L. Mercer, "Design of a Linguistic Statistical Decoder for the Recognition of Continuous Speech", IEEE Trans. Information Theory, IT-21: 250-256,1975.
16) Jelinek, "The Development of an Experimental Discrete Dictation Recognizer", Proc. IEEE,73(11):1616-1624,1985.
17) L. R. Rabiner, S. E. Levinson, A. E. Rosenberg, and J. G. Wilpon," Speaker Independent Recognition of Isolated Words Using Clustering Techniques", IEEE Trans. Acoustics, Speech, Signal Proc., ASSP-27:336-349, August 1979.
18) Sakoe, "Two Level DP Matching –A Dyanamic Programing Based Pateern Matching Algorith for Connected Word Recognition," IEEE Trans. Acoustics, Speech, Signal Proc., ASSP-27:588-595, December 1979.
19) S. Bridle and M.D.Brown, "Connected Word Recognition Using Whole Word Templates," Proc.Inst.Acoust.Automn Conf.,25-28, November 1979.
20) S. Myers and L. R. Rabiner , " A Level Building Dyanamic Time Warping Algorithm For Connected Word Recognition," IEEE Trans.Acoustics, Speech, Signal Proc., ASSP-29:284-297, April 1981.
21) H. Lee and L. R. Rabiner, "A Frame Synchronous Network Search Algorithm for connected Word Recognition," IEEE Trans. Acoustics, Speech, Signal Proc., 37(11): 1649-1658, November 1989.
22) Ferguson, Ed., Hidden Markov Models for Speech, IDA, Princeton, NJ, 1980.
23) R.Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proc.IEEE, 77(2):257-286, February 1989.
24) R. P. Lippmann, "An Introduction to Computing with Neural Nets," IEEE ASSP Mag.,4 (2):4-22,April 1987.
    a. Weibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, "Phoneme Recognition Using Time-Delay Neural Networks" IEEE Trans.Acoustics, Speech, Signal Proc.,37:393-404,1989.
25) F. Lee, H. W. Hon, and D. R. Reddy, "An Overview of the SPHNX Speech Recognition System," IEEE Trans.Acoustics, Speech, Signal Proc., 38:600-610, 1990.
26) Y. L. Chow, M.O.Dunham, O. A. Kimball, M. A. Krasner, G. F. Kubala, J. Makhoul, S. Roucos, and R. M. Schwartz, "BBYLOS: The BBN Continuous Speech Recognition System," Proc.ICASSP 87, 89-92, April 1987.
27) B. Paul, "The Lincoln Robust Continuous Speech Recognizer," Proc, ICASSP 89, Glasgow, Scotland, 449-452, May 1989.
28) Weintraub et al., "Linguistic Constraints in Hidden Markov Model Based Speech Recognition," Proc.ICASSP 89, Glasgow, Scotland, 699-702, May 1989.
29) Zue, J. Glass, M. Phillips, and S. Seneff, "The MIT Summit Speech Recognition System:A Progress Report," Proc .DARPA Speech and Natural Language Workshop,179-189,February 1989.