International Journal of Advance Research, IJOAR .org
Volume 1, Issue 7, July 2013, Online: ISSN 2320-9194

SPEECH DRIVEN PERSONAL SCHEDULING SYSTEM

Devi Thirupathi1, Ashwin K. Whitchurch2, Vishnupriya.R3
1 Head i/c, Department of Computer Applications, Bharathiar University, Coimbatore, India.
2 Circuitects Electronic Solutions (P) Ltd, Bangalore, India.
3 Research Scholar, Department of Computer Applications , Bharathiar University, Coimbatore, India.

1tdevi5@gmail.com
2ashwinkw@ieee.org
3vispriyaraju@gmail.com

Abstract—
If you are a busy person, a scheduler would probably be the most indispensable tool you carry. Schedulers come in all forms – diaries, memo pads, computer based scheduling programs, organizers and PDAs (Personal Digital Assistants). All these devices use handwriting or a keyboard/pointing device as the primary mode of input, which is not always efficient, particularly when the user does not have time to note anything down. The Speech Driven Personal Scheduling System (SDPSS) proposed in this paper uses speech as a mode of input which is very convenient and useful for a quick note of things. Also, the use of a wireless phone for the purpose eliminates the need to carry a PDA in addition to a phone.

Keywords:

PCM Voice Codec, Speech Recognition, Mobile Based Scheduling System, Text Parser.

1. INTRODUCTION

Schedulers usually use handwriting recognition (Graffiti) or a keyboard based interface for input. While a keyboard based interface is good to use, it could become a difficult task when the user is in a hurry. Similarly, Graffiti can be equally hard to use in such situations. In such cases, an alternate mode of input would be the solution. In this paper, a Speech Driven Personal Scheduling System (SDPSS) is proposed which uses speech as the primary mode of input.

Speech is the easiest, convenient and most common way for people to communicate. Speech is also the fastest way than typing on a keypad and more expressive than clicking on a menu item. For these reasons, speech applications are important, especially for users with low literacy or little lettering knowledge, such as those in the developing regions [1]. Devices which support speech are ubiquitous in the market. Some of these devices include wireless phones with support for voice enabled dialing and voice commands, PDAs (Personal Digital Assistants) with voice driven command interfaces, etc. But, these devices are not effective and need to be programmed or trained before they can be used efficiently. Most of them work by relating a recorded voice command to the particular system command. On the other hand, mobile internet devices, internet tablets, smart phones and cell phones have widely proliferated the market, both in developing nations and in the low socio-economic communities of the developed world, sometimes to the extent that users are more likely to have these mobile devices than a personal computer. Thus mobile phones with voice recorder will be better choice for this system [1].

The recognition capabilities of voice-based applications are limited. But this does not place a restriction on the capabilities of this SDPSS system because the voice command-set to be recognized by a scheduling application is also small. So such a voice based SDPSS system would be ideal for scheduling applications.

The SDPSS system envisioned: 'John is busy with some activity; suddenly he remembers that he has to attend an important meeting tomorrow, he just records this message into his personal voice recorder and continues with his activity. Later, when he reaches his office, all he needs to do is just plug-in the recorder into a docking station connected to the USB (Universal Serial Bus) port of the PC. The tasks he had recorded are automatically converted to text and stored in his personal scheduling software such as Microsoft Outlook. Figure 1 shows an illustration of the SDPSS system in action.

The SDPSS system proposed in this paper is designed to work as a scheduling aid which can recognize certain phrases which are used frequently for scheduling tasks and thus need no training or prior voice recording from the user. Examples of such phrases include 'Meet Joe at 9 PM tomorrow', 'Doctor's appointment on September 2 at 9 AM', 'Call John when I get home' or 'Attend Project X meeting on Friday morning'. In order for the program to recognize such phrases as given above, in addition to recognizing speech, it should have some smart capabilities built-in to automatically categorize and order the tasks in the user's scheduling program.

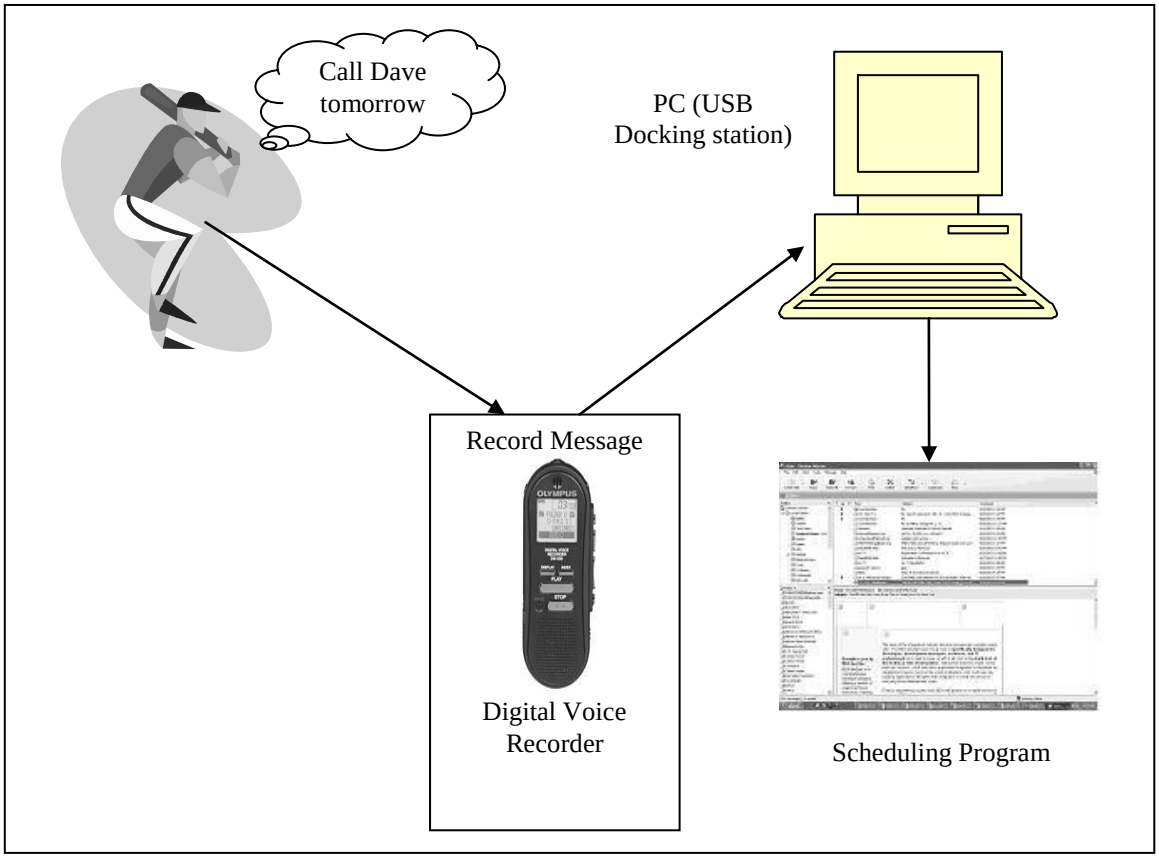


Fig. 1. Proposed Scheduling System.

An example – 'Meet Joe at 9 PM tomorrow', this phrase by itself cannot be useful because the phrase does not include a date field, in this case the program should be able to insert a new record into the user's appointments database with the information 'Meet Joe' as the appointment and '9 PM on ({Today's date} +1)' in the time field of the record. Consider the phrase 'Attend Project X meeting on Friday morning'. In this case, similarly, the program should then assign to the date field, the earliest Friday following {today}.

2. SYSTEM DESIGN

The SDPSS system includes hardware to record the user's voice command (a voice recorder) and software which performs the functions of downloading the voice data from the recording device, extracting information from the data using voice recognition techniques, classification of tasks and commands based on keywords detected from the voice data and entry of the appropriate tasks in the correct sections in the scheduler application/database. Figure 2 shows the components involved in the entire SDPSS system.

In recent years, significant progress has been made in advancing speech recognition technology with the help of speech recognition engine, making speech an effective modality in both telephony and multimodal human-machine interaction [2]. Speech is considered to be the most natural means of communication for human beings. There are several reasons why speech recognition is becoming a standard feature in mobile phones.

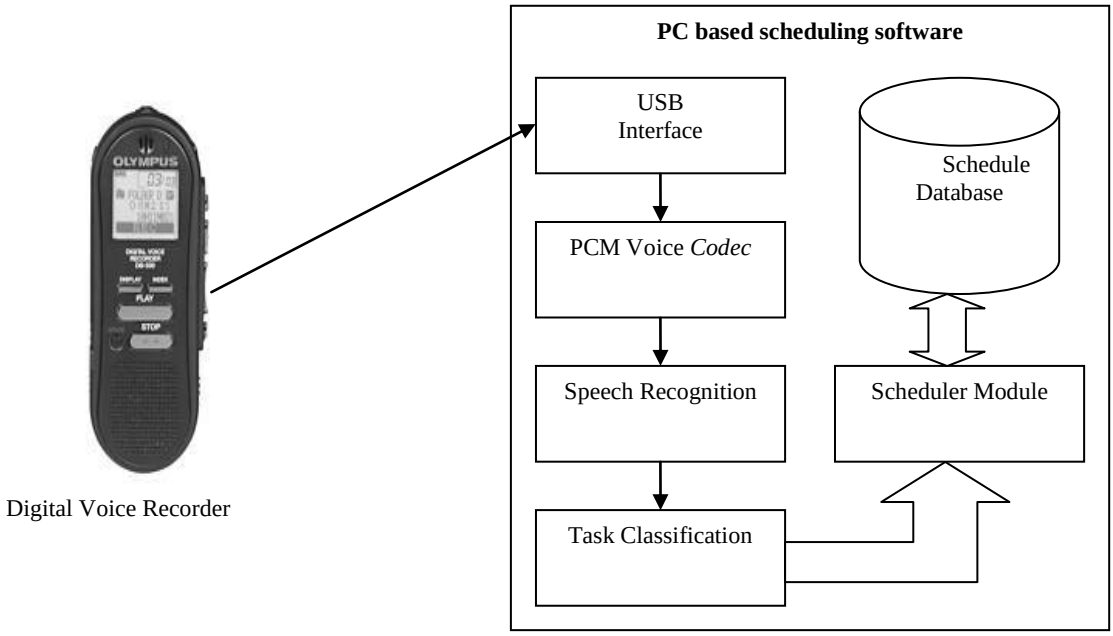


Fig. 2. Overall System Components.

While driving a car mobile phone usage has been regarded as dangerous, because it distracts the driver. The driver must remove the hands from steering wheel to punch the text into keypad to save the programs in the mobile phones [3]. Due to these reasons speech recognition technology becomes essential process in mobile phones. Coupled with the restricted user interfaces, the increase in functionalities of the devices makes speech recognition on mobile devices an even more demanding challenge [4].

Human computer interaction (HCI) is concerned with the design, evaluation and implementation of interacting computing systems for human use and with the study of major phenomena surrounding them [5, 6]. It is an interface and interaction style of a system which enables the users to interact/communicate with system, and allows the users to use it successfully and accomplishing the intended task [7].

A HCI has to be user-friendly and has to speed up the user's work; and it has to be more attractive for non-technical people so that they will feel that the system is easy to use. Even if the system has achieved its objectives, if the HCI is not built properly then it can be very difficult to work with and hence would affect the success of the system. Building a good and successful HCI is a time consuming process. HCI can be available in various forms such as text, graphics, images, voice, sounds, videos and formatted data. Speech plays an important role in communication and is used to express the thoughts. A Voice User Interface (VUI) makes human interaction with computers possible through a voice/speech platform in order to initiate an automated service or process [8]. Voice User Interface (VUI) systems are capable of, besides recognizing the voice of users, to understand voice commands, and to provide responses to them, usually, in real time [9]. This proposed SDPSS system uses voice/speech as input for scheduling daily tasks for busy persons.

In order to develop such a scheduling application, a large keyword vocabulary is not required. For example, the types of scheduling activities could be classified into appointments, events, contacts, etc. So, some of the keywords the application needs to recognize could be given as ‘meet’, ‘attend’, ‘call’ or ‘lunch’. Thus, the grammar to be recognized by the system could be reduced to these specific set of keywords. This could considerably increase the efficiency and accuracy of the system because it does not have to search through a large database of keywords for a proper match. In other words, a ‘complete dictionary search’ is not required as the ‘dictionary’ itself is limited to these words only.

3. DESIGN CONSIDERATIONS

Speech recognition is a complex computing task and requires a considerable amount of computing resources. So, the performance of the system could vary to a large extent on the configuration of the PC on which it is being used. This should be taken into account when implementing the system [10].

There are various devices which could serve as voice input devices for this system. One solution would be to use a personal digital voice recorder as the portable recording device. The user carries the recorder with him and whenever he/she wishes to record something, he/she simply uses this recorder to record the phrase. Later, at the office, this recorder is placed in a USB cradle which is connected to the desktop PC. The recorded files are automatically downloaded to the PC and the program goes to work in adding the appropriate tasks/appointments/contacts to the user’s scheduling program. Figure 3 envision the following:

3.1 Infix Voice

As the initial task the user wants to record his/her daily schedules in a voice recorder of a mobile phone. After recording the voice, the user has to hook the mobile device to PC.

3.2 USB Interface

The Universal Serial Bus is a top-down serial interface that is driven by the “top” (the computer) [11]. The USB communication is based on FTDI-FIFO (Future Technology

Devices International- First In First Out) controller. Connections to other equipment can be made through hubs, resulting in a branched network. Sending and receiving of data is always initiated by the "top" computer [12]. When the mobile device is hooked with USB port; USB enumeration process will take place. In the USB enumeration procedure the host will send a request to the hub to establish a connection between host and device. The hub sends the host's request by issuing a control transfer. The device should respond to the request by providing requested information. On most operating systems, specific drivers are needed for each device [11].

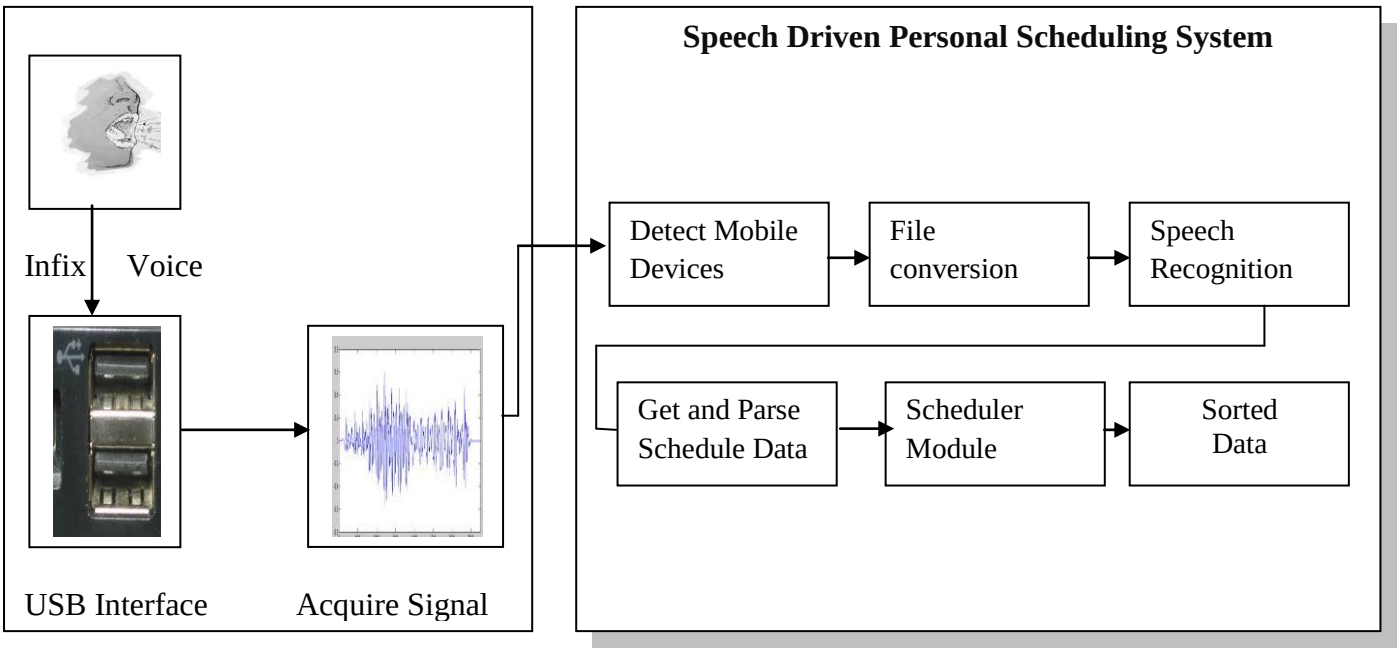


Fig. 3. Architecture Diagram.

3.3 Acquire Signals

Signal should be acquired to decide the speed of the device. The voltages on the signal lines (D+ and D-) are monitored by the hub at each of its ports. The hub has a pull-down resistor of 14.25k–24.8kW on each line. A device has a pull-up resistor of 900–1575W on D+ for a full-speed device or D- for a low-speed device. High-speed-capable devices attach at full speed. On attaching to a port, the device's pull-up brings its line high, enabling the hub to detect that a device is attached. On detecting a device, the hub continues to provide power but doesn't yet transmit USB traffic to the device [13].

3.4 Detecting Mobile Device

After the signal is attained, the device drive should be detected. From the detected device drive, the audio files which are all particularly recorded for scheduler program are opened.

3.5 File Conversion

The voice files are originally in *.amr* (Adaptive Multi-Range) format. *Amr*, however will not supportable in SAPI (Speech Application Programming Interface) which is used for speech recognition. The file will be converted to intermediate format PCM (Pulse Code Modulation) and then PCM format will be converted to *.wav* (Wave) files.

3.6 Speech Recognition

Voice conversion has become more and more important in speech technology. Converting a speech waveform into a sequence of words involves several essential steps. First, speech engine picks up the acoustic signal of the speech to be recognized and converts it into an electrical signal. A modern speech recognition system also requires that the electrical signal be represented digitally by means of an analog-to-digital (A/D) conversion process, so that it can be processed with a digital computer or a microprocessor. For converting voice to text, Microsoft SDK (Software Development Kit) tool SAPI is used.

3.7 Get and Parse Schedule Data

In the converted text, task classification should be made. In the task classification date, time and appointment should be separated. The separated task should be given to the scheduler to schedule the programs.

3.8 Scheduler Module

The separated data will be scheduled according to the date, time and appointment programs. The scheduled programs will be stored in the database.

3.9 Sorted Data

Finally sorted data will be displayed as an output (i.e.) the scheduled program will be given a priority of date and time.

4 SYSTEM DEVELOPMENT

This SDPSS system is developed using Microsoft .Net Framework, and C# as a tool for coding. In the most common form, to install an application, all you need to do is copy the application along with the components it requires into a directory on the target computer. The .NET Framework manages the process of locating and loading the components that an application needs, even if several versions of the same application exist on the target computer. The .NET Framework ensures that all the components the application depends on are available on the computer before the application begins to execute.

The SDPSS system is built using the Microsoft Speech Application SDK. The SDK includes pre-built grammars for various speech inputs such as common terms, numbers and adjectives. Since the application is expected to recognize only a limited set of voice commands, this should be sufficient without any additional dictionaries. This simplifies the development of new grammars and also helps to avoid the training and voice recording stages which is important when considering development of an out-of-the-box voice package [14].

The Speech Application Software Development Kit (SASDK) includes a rich grammar library that contains a robust set of rules written in the W3C-approved Speech Recognition Grammar Specification (SRGS) format, which helps developers obtain abstract, complex concepts from users [15]. For example, gathering something as basic as a recognizable date input is quite complex because the way in which users express dates is highly variable. The grammar library enables the application to make an intelligent determination of the exact date and year.

Comparison with the Existing Tool

One of the important existing tools for voice recognition is Vlingo that deals with searching, making phone calls and sending messages. Vlingo has no provision to incorporate schedulers and reminders. SDPSS allows the users to give message to scheduler through voice and it schedules the task as well as reminds the schedules to the users.

SIRI is an intelligent personal assistant and knowledge navigator which works as an application for Apple's iOS. SIRI allows one to send messages, schedule meetings, find places, browse information and make phone calls. SIRI uses a natural language user interface to answer questions, makes recommendations and performs actions by delegating requests to a set of web services. It does not require any training. For example, if a question is asked like "What time is it?", it is responding properly. If any appointment is given to the scheduler, for example, "Attend project review on July 15 at 9 AM" is given, it recognizes the task as a person's name like *Arun* and interprets the given sentence as "make a call to that person" and initiates a call. SIRI requires internet access and it accepts human voice as an input with limited vocabulary. Thus, the SIRI tool cannot be directly applied for scheduling based on voice input and hence not suitable for the proposed work of scheduling.

5 RESULTS AND DISCUSSION

The result obtained from the developed SDPSS system is classified tasks. Daily appointments are recorded using mobile phones and the voice files are downloaded to a PC. The *.amr* voice files are converted to *.wav* files by keeping PCM voice *codec* as an intermediary. The obtained *.wav* files are successfully converted to text files using Microsoft SDK tool SAPI. The text files are parsed and given to the scheduler. Finally, scheduled programs with appropriate date, time and appointments will be obtained.

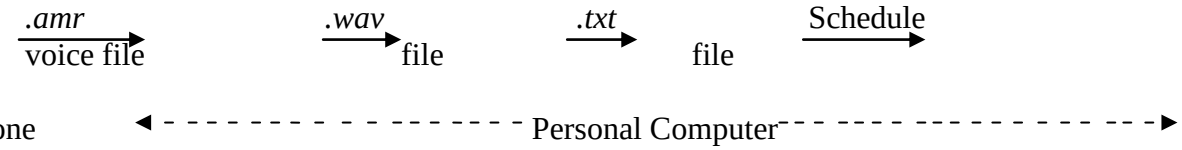


Fig. 4. Conversion Process.

Figure 4 shows the conversion process flow from voice file to parsed data that represents the schedule (i.e.) from mobile phone to PC. For example, if the input voice is "Attend national level conference on July 15 at 11:00AM" then the corresponding parsed data will be "Date: July 15, Time: 11:00 AM and Program: Attend national level conference" as shown in the first row of the Table 1.

Table 1: **Parsed Data.**

Date	**Time**	**Programs**
July 15	11:00 AM	Attend national level conference
July 19	10:00 AM	Appointment with doctor raju
July 20	10:00 AM	Call john
July 28	09:00 AM	Head of the Departments meeting

The developed software system has been tested using 100 sample voice inputs containing 686 words. Out of 686 words, 593 words were recognized, thus yielding 86.44% of success. Figures 5 and 6 show the list of some words given as a part of voice input that are recognized and list that are not recognized by the developed system respectively. Graph 1 depicts the percentage of success. It also shows number of words recognized and number of words not recognized. The system interprets some of the words in a different way, for example, when the word 'goto' is given as a part of the voice input, it is interpreted as 'delta'. Such cases lead to further research and a list of these words is listed in Table 2.

Address	College	District	Journal	Mining	Property
Administrative	Competition	Doctor	July	Multimedia	Provisional
Application	Conference	Draft	June	National	Quotation
Appointment	Consult	Election	Lab	Notification	Report
April	Consulter	Engineer	Lawyer	November	Result
Article	Convocation	Exam	Level	October	Scientist
Assignment	Corporate	February	Library	Office	Secretary
Attend	Counseling	Football	Logistics	Officer	Security
Auditor	Counselor	Function	Lunch	Paper	Seminar
August	Cricket	Government	Major	Passport	Send
Awareness	Customer	Head quarters	Manager	Placement	September
Before	Data	Health	March	Practical	Staff
Center	December	Insurance	Materials	Premium	Student
Check	Details	International	Media	Prime	Submission
Client	Discharge	Interview	Medical	Program	Tax
Collector	Discuss	January	Meeting	Project	Technician

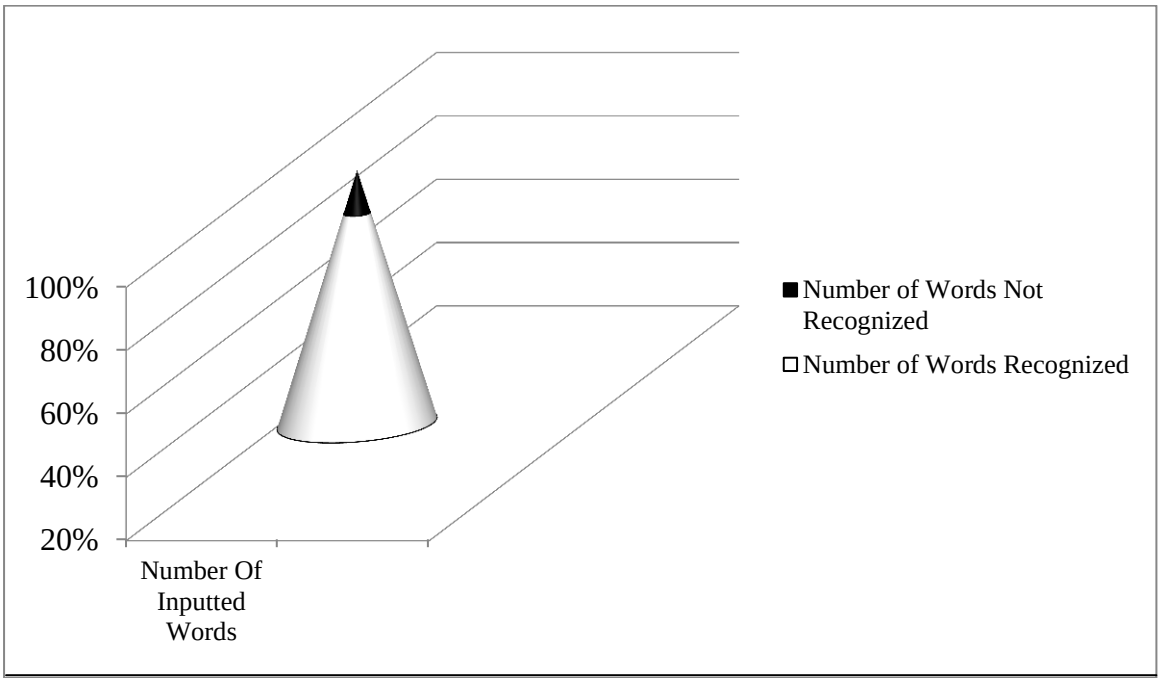
Fig. 5. Recognized Words.

Acknowledgement	Chamber	Game	Informed	Ram
Affect	Check-up	Get	Intake	Register
Afore	Chess	Get degree	Invoice	Submit
Ahead	Debugging	Goto	Inward	Survey
Bank	Deliver	Guide	Meet	Take
Board	Demand	Heedful	Memory	Tournament
Branch	Enroll	Hockey	Menu	Tournaments
Call professor	Exhibition	Hours	Pay	Visit
Certificate	For	Impart	Payroll	
Chairmen	For tax	Income	Quiz	

Fig. 6. Words that are not recognized.

Table 2: **Predicted Results.**

Given word	**Predicted as**	**Given word**	**Predicted as**
Acknowledgement	Acknowledge and	Guide	Died
Affect	Athlete	Heedful	He will
Afore	A fresh	Hockey	Talking
Ahead	I had	Hours	Owners
Bank	Tank	Impart	In part to
Board	Word	Income	Into
Branch	Tulare	Inform	Informed
Call professor	Colton faster	Intake	And take
Certificate	Third to think it	Invoice	In voice
Chairmen	Tiernan	Inward	And ward
Chamber	Steamer	Meet	Meat
Check-up	Jack-up	Memory	Mandarin
Chess	Chance	Menu	Manuals
Debugging	Bounding	Pay	The
Deliver	The labor	Payroll	Royal
Demand	And	Quiz	Plays
Enroll	Enrolled	Ram	Ran
Exhibition	Accidents	Register	Rejected
For	Four	Submit	Subnet
For tax	40 acts	Survey	Sergei
Game	Jeanne	Take	Taken
Get	The	Tournament	Jordan
Get degree	Getting to	Tournaments	To learn and
Goto	Delta	Visit	David



Graph 1 Illustrates the percentage of no. of words recognised and no. of words not recognised

The advantages of this SDPSS system would be: ① easy way to store as it involves voice input, ② quicker to store, ③ avoids searching for pen and paper to note down, ④ it is handy and reachability of information is high as it is in the mobile phone, ⑤ reduces the time to schedule and thus increases the efficiency and ⑥ automatic. The limitations of this work involves: ① bulk of files cannot be parsed as parsing of the voice input is carried out on a file by file basis and ② abbreviated form of words like IAS, IFS, NET, and GATE cannot be accepted.

Overall SDPSS system proposes a convenient way for the user to record their important activities in their day to day schedule.

5 CONCLUSIONS

This paper discusses the development and implementation of a speech driven personal scheduling system. Such a system would be very useful and convenient over other forms of scheduling systems currently available. The implementation of the recognition client desktop application as a web-based application would enable multiple users to use the same software. Implementation of the concept on a Pocket PC based device initially is also being developed by the authors.

REFERENCES

[1] Anuj Kumar, Anuj Tewari, Seth Horrigan, Matthew Kam, Florian Metze and John Canny, "Rethinking Speech Recognition on Mobile Devices", 2011.
[2] Li Deng and Xuedong Huang, "Challenges In Adopting Speech Recognition", Communications of the Acm January 2004/Vol. 47, No. 1.
[3] Stefan Dobler, "Speech recognition technology for mobile phones", 2000.
[4] Zheng-Hua Tan and Børge Lindberg, "Speech Recognition on Mobile Devices", 2010.
[5] Achim Ebert, Gitta Domik, Nahum Gershon, Gerrit van der Veer, "Building Bridges-HCI and Visualization", 2011.
[6] Hewett, Baecker, Card, Carey, Gasen, Mantei, Perlman, Strong, and Verplank: "ACM SIGCHI Curricula for Human-Computer Interaction; chapter 2: Human-Computer Interaction", Last updated in 2009.
[7] Syed Imran Ali, Sameer Jain, Banshi Lal Patidar, Neeraj Sharma , "A Framework for Modeling and Designing of Intelligent and Adaptive Interfaces for Human Computer Interaction", International Journal of Applied Information Systems (IJAIS) Volume 1– No.2, January 2012.
[8] http://en.wikipedia.org/wiki/Voice_user_interface, Voice user interface - Wikipedia-the free encyclopedia, July11 2012.
[9] Valéria Farinazzo, Martins Salvador, André Luiz S. Kawamoto and João Soares de Oliveira Neto, "An Empirical Approach for the Evaluation of Voice User Interfaces", 2010.
[10] Panagiotis Louridas, Diomidis Spinellis, and Vasileios Vlachos, "Power laws in software", *ACM Transactions on Software Engineering and Methodology*, 18(1):1–26, September 2008.
[11] Arthur Edelstein, Nenad Amodaj, Karl Hoover, Ron Vale, and Nico Stuurman, "Computer Control of Microscopes Using UNIT 14.20 μManager", 2010.
[12] Zbigniew Guzik, Stanisław Borsuk, Krystyna Traczyk and Michał Płomiński "Enhanced 8K Pulse Height Analyzer and Multi-Channel Scaler (TUKAN) with PCI or USB Interfaces", 2004.
[13] Jan axelson, "USB Complete; chapter 4: Enumeration: How the Host Learns about Devices", Fourth Edition" Last updated in 2009.
[14] Microsoft Corporation, Features of the Speech Application SDK v1.0 Beta 3, Microsoft Developer Network (http://msdn.microsoft.com), July 2003.
[15] John Mierau, "Voice Recognition Programs for the Pocket PC", Pocket PC Magazine, 2002.