

Voice recognition – let's make things easier
By Waseem Ahmad
[wahmad@titledevelopments.com]

Abstract

The importance of software in business is no longer in doubt and now user friendliness; response time and accuracy are the key features of every successful software application. Voice or speech recognition is one step towards easier to use software. Instead of using a keyboard or mouse, the user simply talks to his/her computer. This paper presents a brief introduction to speech/voice recognition technology and describes some of Title Developments work and expertise in this area. It is important to note in this paper that voice and speech have the same meanings.

1. Voice Recognition – Introduction

Software is an essential business need today. But the question is: can we make it more user-friendly, and simpler to use? The answer is "YES", and voice recognition is one important answer. It allows a user to dictate text into a computer or control it by speaking certain commands (such as open MS Word, pull down menus, save or delete work). Currently voice/speech recognition applications allow a user to dictate text at up to 160 words per minute [1].

Voice recognition uses a neural net to "learn" to recognize a user's voice [1]. To achieve this, the user is given some sample text to speak. In this way the software overcomes the problem of different accents and inflections.

With Voice Recognition software e-mails, memos and reports can be input by dictation, and a user can tell the computer what to do. Speaking into a microphone produces the same result as typing words manually with a keyboard. Voice recognition software applications are designed with an internal database or grammar file of recognizable words or

phrases. The program matches the audio signature of speech with corresponding entries in the database or the grammar file.

Turning speech into text might sound easy, but it is in fact an extremely difficult task. The problem lies in individual speech patterns and accents, compounded by the natural human tendency to run words together [2].

2. Types of Voice Recognition Software Applications

2.1 Speaker dependent systems

This type of system requires the user to "train" the software to recognize the particular stylized patterns of speech which will be used. People commonly use such programs at home or at the office, and Email, memos, letters, data and text can be input by speaking into a microphone.

2.2 Discrete speech systems

This type of system requires the user to speak clearly and slowly and to separate words. Continuous speech systems are designed to understand a more natural mode of speaking. Discrete speech voice recognition systems are typically used for customer service routing. The system is speaker independent, but understands only a small pool of words or phrases. The caller is given a question and then a choice of answers, usually "yes" or "no." After receiving an answer, the system escalates the caller to the next level. If the caller replies with an answer that can't be recognized the automated response is usually, "Sorry, I didn't understand you; please try again," with a repeat of the question and available answers. This type of voice recognition is also referred to as grammar constrained recognition [3].

2.3 Automatic Speech Recognition [ASR]

Automatic Speech Recognition [ASR] is a model of voice recognition applications designed for dictation, and is different from previous models. This type of application does not strive to understand what is being said, only to identify the words spoken. There are many words in the English language which sound alike, so it is very easy for mistakes to occur in these applications. ASR software is often found on digital voice recorders such as MS Voice recorder.

3. Components for voice recognition

3.1 Basics

Every speech recognition system uses four key operations to listen to and understand human speech [4]. They are:

- a. Word separation - This is the process of identifying discreet portions of human speech. Each portion can be as large as a phrase or as small as a single syllable or part of a word.
- b. Vocabulary - This is the list of speech items that the speech engine can identify.
- c. Word matching - This is the method the speech system uses to look up a speech portion in the system's vocabulary - the search engine part of the system.
- d. Speaker dependence - This is the degree to which the speech engine is dependent on the vocal tones and speaking patterns of individuals.

To develop applications, one also needs to look into the following concepts and technologies.

3.2 SAPI:

The Speech Application Programming Interface [SAPI] was introduced by Microsoft in 1995, and allows a system to recognize human speech as input, and create human-like audio output from printed text. This facility provides a new dimension to human - PC interaction [4]. SAPI is part of the Windows Open Services Architecture [WOSA] model. Speech recognition [SR] and text-to-speech [TTS] services are actually provided by separate modules called engines [4] [5]. Every version of Windows since XP includes SAPI and an English TTS engine. Users can use these or a third party speech engines as long as it conforms to the SAPI standard. Figure 1 [6] shows the basic SAPI architecture and its interaction with an application. [DDI = Device Driver interface, API = Application Programmer interface.]

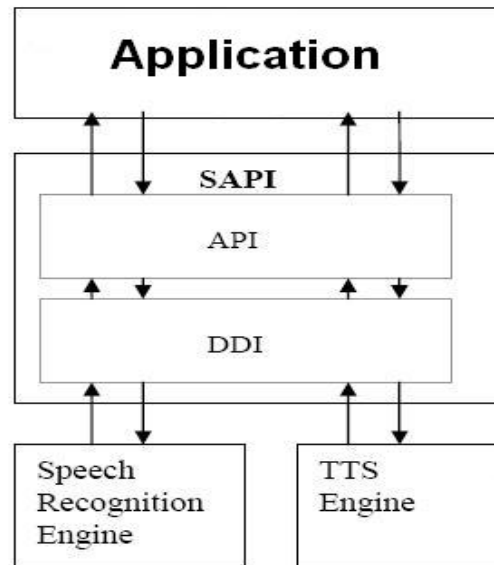


Figure 1: SAPI architecture

3.3 New features in SAPI 5.1:

Telephony functionality:

- a. New objects designed for telephony
- b. New objects designed for using speech and TAPI
- c. Telephony controls

- d. Telephony sample applications
- e. Telephony tools

ActiveX components:

- a. For use in Visual Basic, VB-Script, Visual Basic for Applications, Java, Java-Script
- b. Lots of samples

Other samples and tools:

- a. A Grammar compiler
- b. A Wave editor
- c. An animated mouth
- d. Speech recognition and text-to-speech stress tests

SAPI 5.1 contains new direct text-to-speech and speech recognition APIs, plus support for continuous dictation for use in Voice Dictation applications [4] [5].

3.4 Future of SAPI:

At present, SAPI Applications and systems are most successful as command-and-control interfaces [4]. Up until now the technology has offered only limited voice playback services. Users can get replies and short passages can be read out without trouble, but the playback of a lengthy text is still difficult to understand.

With the help of the generalized interfaces from Microsoft SAPI and the newer versions of MS TTS & SR; new voice applications will be more accurate and efficient than was previously possible. With each new release of Windows, and new versions of the SAPI interface, speech services are bound to become more powerful and more user-friendly [4].

4. Voice recognition Engines

4.1 Voice recognition engines are designed for specific applications, and can be categorized into two types [6]:

- a. Command and Control Applications
- b. Dictation Applications

The first category application's recognizes the voice/speech of the user and executes

it as a command. Where as, the second category application's will only turn the users voice/speech into text.

4.2 Microsoft VRE

The Microsoft speech recognition engine allows a user to insert text into a document using a specific program. The text file is known as a Grammar file. The user can dictate text into any Microsoft Office XP program, Internet Explorer (from 5.0), and Outlook Express (from 5.0). Other software programs might eventually support the Microsoft speech recognition engine [5]. Text cannot be dictated into Notepad at this time [5].

4.3 Some speech engines are language-specific while other speech engines may be region-specific [5]. For example, the *"Microsoft English ASR Version 5 engine is intended for speakers of U.S. English. British, Australian, and other non-U.S. English speakers may have difficulty using it due to variations in accent [5]"*.

4.4 To use speech recognition the following are essential requirements:

- a. A high-quality headset
- b. A microphone
- c. A sound card or USB port

Next comes training of the speech recognition engine to understand a user's voice. The engine looks for the patterns in a saved sample of speech. This training creates a speech profile for the individual speaker [5]. Speech recognition is not designed for completely hands-free operation [5] and at is rarely 100% perfect because of problems such as noise, distortion, etc.

5. Grammar File

The final ingredient in developing a voice recognition application is the Grammar file. Grammar rules are used by the speech recognition application to analyze and identify human speech input, process it, and attempt to understand what a

person is saying. The application educates itself using the grammar file and this has been compared to kids in school who learn grammatical rules and having done so, speak without thinking about those rules. There can be three different categories of grammar files as follows:

a. Context Free Grammar:

Examples of rules in a context-free grammar are something like:

```
<NameRule> = ALT ("Kevin", "Andy")  
<SendMailRule> = ("Send Email to",  
<NameRule>)
```

Context-free grammar has good flexibility when interpreting human speech.

b. Dictation Grammar:

Dictation grammar applications base their evaluations on vocabulary. They convert the human speech into text as accurately as possible. To achieve this they need to have a very rich vocabulary.

The success of dictation grammar systems depends upon the quality of the vocabulary and most applications are used in a single subject or topic, for example legal or medical [4].

c. Limited domain Grammar:

These applications use a combination of context-free grammar and dictation grammar methods to achieve a limited domain grammar file which have the following elements:

- a. Words - a list of words that are frequently used.
- b. Groups - a set of related words that might be used.

This type of grammar file is very useful where the vocabulary of the system is small. For example systems that use natural language to accept command statements, such as "How can I open a new document?" or "Replace all instances of 'New York' with 'Los Angeles.'" Limited domain grammars also work well for filling in forms or for simple text entry [4].

6. Our experience

We developed a Voice recognition and Call routing application for one of our clients. It was developed using the .NET platform and is a state of art application for routing calls and checking/reading emails through Microsoft Exchange Server integration. When a user calls into the application, he/she is guided through the necessary steps and then his/her voice is taken as an input to trigger the appropriate task to be performed.

The benefits are as follows:

- a. Hand free operation
- b. Fast response times
- c. Maintaining records of important customer requirements
- d. All records calls are stored centrally for later referral
- e. Users can listen to voice mail and e-mail from any touch tone/mobile phone.
- f. Users can browse the interface and manage messages, recorded calls and personal Voice Activated Dialing contacts.
- g. Reduces receptionist workload by allowing customers to ask for a department or person and be automatically transferred.
- h. It is no longer necessary to remember the phone number of each employee, just say the name.
- i. Streamlines the business messaging process; this enables administration from any browser enabled workstation to control of the following:
 - i. Security and user restrictions
 - ii. Maintenance of the system

This application is particularly useful for hospitals, the police, fire brigades, etc. where time is very critical. Businesses with a large number of employees or client centric businesses can also achieve benefits from these applications.

6.1 Features

The two main features of this application are Voice Activated Dialing [VAD] and Messenger Assistant [MA], a brief introduction to them follows.

Through VAD, a user can dial into the system. Our application monitors the call and when the caller speaks, it recognizes the user input as a command and performs the appropriate action. For example it might redirect the call to a specific user or to an Interactive voice response [IVR] system.

MA is integrated with Microsoft Exchange Server and is used for reading emails. A user can dial into the application and the system will read out his/her emails.

6.2 Our approach

Developing a voice recognition application is a complex operation. In this case it was developed in Microsoft .NET and SQL 2000. Different rapper classes were developed using SAPI.

The application was developed using a 4 tier architecture. The 2nd tier was for business logic and the 3rd tier for voice recognition. A Grammar file was used for voice recognition. The application was intelligent enough to update the Database and Grammar file for speech recognition.

The application was divided into a number of modules, and each has very clearly defined interfaces. This enabled a repository of reusable components to be created for use in the future.

Quality Assurance [QA] is very important for every application. Automated testing cannot be done on this type of application because its very nature requires it be tested against different voices and accents, therefore manual testing is required. For testing, the application was trained on our QA personnel's voices and accents. The success ratio in this case was more than 90%, a considerable

achievement. The main problems with the kind of application were discussed earlier "The problem lies in individual speech patterns and accents, compounded by the natural human tendency to run words together [2]". So it is very important to test voice recognition on different voices and different accents. The QA process was very simple in this case, extensive Test Cases writing was not required and thus the overall QA time was shortened.

Rigorous configuration control for this type of application is essential. Version numbers and release notes are developed for every release. Microsoft Visual SourceSafe is used for the automated configuration of code, documentation and application releases.

7. Conclusion

Title Developments welcomed the opportunity to develop this application for an international client. It allowed us to gain practical experience in this important area of advanced technology, and to introduce and validate our ideas on how such applications should be developed and tested. The results achieved have been considered most satisfactory and provide an excellent starting point for future innovations in this field.

8. References

- [1]. <http://www.utoronto.ca/atrc/reference/tech/voicerecog.html>
- [2]. <http://www.about-business.org/Speech-Recognition.php>
- [3]. <http://www.wisegeek.com/>
- [4]. <http://project.uet.itgo.com/sapi.htm>
- [5]. <http://msdn.microsoft.com>
- [6]. Development of a Field-Deployable Voice-Controlled Ultrasound Scanner System by Dalys Sebastian, 2004.