A VOICE INTERFACE FOR THE VISUALLY IMPAIRED

Anand Arockia Raj, Rahul.Ch, Susmitha

IIIT- Hyderabad, India
anand@iiit.net
susmitha@iiit.net

IIIT-Hyderabad, India
rahul_ch@students.iiit.net

Abstract: This template gives you guidelines to prepare papers for SETIT 2005 Conference. Use this document as a template if you are using Microsoft Word 6.0 or later. Otherwise, use this document as an instruction set. Please note that use of SETIT 2005 templates is meant to assist authors in correctly formatting manuscripts for final submission. This template must be used for both initial and final submissions. An abstract should be 100 to 200 words, and should clearly state the nature and significance of the paper. Abstracts must not include mathematical expressions or bibliographic references.

Key words: RAVI (reading aid software for visually impaired people), ASR (Automatic speech recognition system) Diphones, Hidden Markov Models, TTS (Text to speech), Lexicon, Speech, Accessibility, COM (Component Object Model).

1. Introduction

Man machine interfaces have always proved a challenging area in natural language research. It is merely utopian to contemplate a natural language interface. The very idea of such a system suggests a very helpful way for enabling the visually impaired and computer laymen to use the latest technologies.

In this paper we aimed at how a computer can help a visually impaired person. A normal person generally communicates with the computer with a mouse or a keyboard. But this is not possible for the visually impaired. It requires lot of training and hard work. Handling a mouse is not at all possible. So, speech recognition is a good way of communication in this case. Then if this user speaks out some command then computer will get his job done.

We have developed an interface that opens applications like internet explorer, Microsoft word, etc and reads the text whatever is there on the screen.

2. Reading Aid for visually impaired (RAVI)

Reading Aid for Visually Impaired is a screen reader which reads out the current screen context details such as displayed controls, windows and messages with their current state and the action so that the visually challenged user can imagine and understand the active screen details and perform the action according to his interests. This software integrates itself with almost all the variants of the windows operating system and reads the menus, text boxes, dialogs and all other widgets of the screen. This helps the user to navigate and open the program or application he wants to use without the help of any person.

The most important feature of this software is its support for Indian languages and usability with almost all the Microsoft suits like MS Word, MS Excel, MS PowerPoint and other applications like Notepad and Internet Explorer. An indigenously developed Text-To-Speech system has been integrated with this interface to provide support for Indian Languages. User can access the text content either with the use of mouse or with the keyboard. Special keys have been defined for
reading out the text from the applications. For example when the users presses HOME key then current line from the current application is read out. Other keys have been provided for reading out the whole document and the provisions for accessing the text content vary depending upon the complexity of these applications. The Indian Language Text to Speech System integrated with the RAVI System is a Unit Selection based System. Unit selection based TTS Systems are able to produce high quality natural sounding speech which in turn has lesser listening effort over longer duration of time.

3. Speech Recognition

To recognize a word, a computer should be trained with some speech data. So, for any speech recognition system a speech database has to be built. The Domain in which we are developing the ASR is important. For a limited domain the set of words which occurs in this domain have to be modelled. A speaker independent speech recognition system requires a large amount of data to be collected from different groups of people age wise and gender wise. It shud also be noted that there is sufficient amount of word models to train the computer. We have used bayesian based HMMS by guassian distribution for developing this.

3.1 Bayseian Formulation

Let $A = \{A_1, A_2, \ldots, A_t\}$ is a sequence of acoustic observations of a speech signal.

Let $W = \{W_1, W_2, \ldots, W_m\}$ be sequence of words.

Given the acoustic observations $A$, the probability of word sequence $W$ is $P(W \mid A)$. For recognition, we choose

$$\text{Best Word} = \arg\max_W (P(W \mid A)) \quad [1]$$

$$\arg\max_W (P(W \mid A)) = \arg\max_W (P(W, A) / P(A)) \quad [2]$$

$$= \arg\max_W (P(W, A)) \quad [3]$$

$$= \arg\max_W (P(A \mid W) * P(W)) \quad [4]$$

A model for the probability of acoustic observations given the word sequence, $P(A \mid W)$, is called an “acoustic model.” A model for the probability of word sequences, $P(W)$, is called a “language model.”

The speech recognition problem can be mainly divided into 3 parts.

3.2 Feature Extraction

First one is the Feature Extraction, which is nothing but speech processing. The speech signal is a continuous signal. But most of the speech processing schemes assume that speech signal change relatively slow with time. This leads to short term processing methods. The speech processing procedure a short segment of the speech signal is isolated and processed. This short segment is called “Frame”. This is repeated with next segment (which may be overlapping with previous) and for all segments. Result of processing on each frame may be a single number or a set of numbers. Processing is first done on these frames. Then the speech signal is transformed from time domain to frequency domain. This could done by applying Fast Fourier Transform or Linear Prediction, or Mel Frequency Filters etc, etc.

3.2.2 Training

Second one is the Training part. This contains the acoustic model and the language model. Given a particular word sequence it gives the probability of acoustic sequence. Language model gives the probability of the occurrence of a word given the previous word sequence. This makes the search states less. But this approach is generally used for large domain recognizers. For our model training is done with our vocabulary.

3.2.3 Decoding

Third one is the Decoding part. This searches for a sequence which gives the maximum probability of $P(W \mid A)$. Which ever sequence gives the maximum probability is selected.

The capturing of speech signal is important while doing any processing on it. It has to be captured satisfying specific parameters like Sampling frequency, bit rate, etc.
3.3 Vocabulary

This is the set of the words which the user will speak to a computer. Computer listens what ever user says and acts accordingly.

1. Internet Explorer
2. Word pad
3. Notepad
4. Yahoo Messenger
5. Msn Messenger
6. Control Panel
7. Search
8. Run
9. Windows Explorer
10. Windows Media Player
11. Adobe Reader
12. Calculator
13. Command Prompt
14. Microsoft Paint
15. On Screen Keyboard
16. Utility Manager
17. Magnifier
18. Sound Recorder
19. Volume Control
20. Character Map
21. Disk Cleanup
22. Disk Defragmenter
23. System Information
24. Recycle Bin
25. Narrator
26. Address Book
27. Outlook
28. Outlook Express
29. My Computer
30. Imaging
31. Microsoft Word
32. Microsoft Access
33. Microsoft Excel
34. Microsoft Front Page
35. Microsoft Powerpoint
36. Real Player
37. Win amp
38. Firefox
39. Netscape
40. Putty
41. Itunes
42. Mplayer
43. Reget Downloader
44. Winscp
45. Network Connections
46. Logoff
47. Turnoff Computer
48. Games
49. Wordweb
50. Matlab

3.4 Data Collection

Speech data was collected from 30 people. Each word in the vocabulary was recorded 5 times so that it would be good for training.

Statistics of data collection

<table>
<thead>
<tr>
<th>Category</th>
<th>No. of people</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age in years</td>
<td></td>
</tr>
<tr>
<td>10-15</td>
<td>2</td>
</tr>
<tr>
<td>15-20</td>
<td>10</td>
</tr>
<tr>
<td>20-30</td>
<td>9</td>
</tr>
<tr>
<td>30-40</td>
<td>6</td>
</tr>
<tr>
<td>40-50</td>
<td>3</td>
</tr>
<tr>
<td>Gender</td>
<td></td>
</tr>
<tr>
<td>Male</td>
<td>20</td>
</tr>
<tr>
<td>Female</td>
<td>10</td>
</tr>
<tr>
<td>Mother Tongue</td>
<td></td>
</tr>
<tr>
<td>Telugu</td>
<td>12</td>
</tr>
<tr>
<td>Hindi</td>
<td>10</td>
</tr>
<tr>
<td>Marathi</td>
<td>2</td>
</tr>
<tr>
<td>Tamil</td>
<td>6</td>
</tr>
</tbody>
</table>
4. Applying Speech Recognition for RAVI

The voice command is given by the user to the speech recognition module. This module calls the applications. If any text is displayed by the application then it the computer reads the text for us.

VOICESHELL ARCHITECTURE

5. Performance

The performance of the system is 99%.

<table>
<thead>
<tr>
<th>Gender</th>
<th>Accuracy %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>99</td>
</tr>
<tr>
<td>Female</td>
<td>98</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Age</th>
<th>Accuracy %</th>
</tr>
</thead>
<tbody>
<tr>
<td>10-15</td>
<td>99</td>
</tr>
<tr>
<td>15-20</td>
<td>100</td>
</tr>
<tr>
<td>20-30</td>
<td>100</td>
</tr>
<tr>
<td>30-40</td>
<td>98</td>
</tr>
<tr>
<td>40-50</td>
<td>97</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mother Tongue</th>
<th>Accuracy %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Telugu</td>
<td>99.3</td>
</tr>
<tr>
<td>Hindi</td>
<td>99.1</td>
</tr>
<tr>
<td>Marathi</td>
<td>98.5</td>
</tr>
<tr>
<td>Tamil</td>
<td>99.2</td>
</tr>
</tbody>
</table>

6. What do visually impaired people expect and how much did we achieve

We did a survey on the needs of blind people for a computer. They feel inconvenient to use a computer because they don’t know what is written on the computer screen. Another problem is they can’t see where mouse is at while accessing the applications. So, there is a great need for this type of interfaces for them.

6.1 How did they respond to this

These are the statistics of their response to this interface.

<table>
<thead>
<tr>
<th>Feedback</th>
<th>% of people</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excellent</td>
<td>75</td>
</tr>
<tr>
<td>Good</td>
<td>20</td>
</tr>
<tr>
<td>OK</td>
<td>4</td>
</tr>
<tr>
<td>Bad</td>
<td>1</td>
</tr>
</tbody>
</table>

7. Future Work

We can extend this idea of using speech for to many the applications. In developing interfaces for the operating systems, we can include these features. Since the present operating systems don’t have these features it would be of great help.

8. References


