A Cost Effective Implementation of a Voice Assisted Home Automation System

B. Bharathi*, Sidhartha Chatterjee¹

¹Faculty of Computing, Sathyabama University, India.
* bharathi.cse@sathyabamauniversity.ac.in, ¹siddharthkchatterjee@gmail.com

Keywords: home automation system, micro controller, raspberry pi, listener, markov models.

Abstract - Home Automation Systems are mainly targeted towards people with physical difficulties for movement. The proposed Voice Assisted Home Automation System (VAHAS) enables one to control all electrical equipment and peripheral devices by means of just one’s voice. By means of interfacing the devices (which could include lights, televisions, air conditioners and so on) with the Linux powered main device running our software, via RF enabled switches, the system responds to the user’s voice commands and intelligently activates, deactivates and changes the state of the appliances. The paper elucidates the actual implementation of a low cost device and shows the results produced by the system. Scalability and customer accessibility of the system is also considered during design.

Introduction

Home automation system is an extension of automating all household services. Though there are number of home automation products available in the market, most of them lack to ability to support multiple languages, cost effectiveness, continuous learning ability, etc as summaried in the table 1 below. For people with movement difficulties voice becomes an easy means of communication. Technology has evolved over the past decade to support people Graphical User Interfaces and voice over systems for communication.[1]

Modern NLP algorithms are based on machine learning, especially statistical machine learning. The paradigm of machine learning is different from that of most prior attempts at language processing. Prior implementations of language-processing tasks typically involved the direct hand coding of large sets of rules. The machine-learning paradigm calls instead for using general learning algorithms - often, although not always, grounded in statistical inference - to automatically learn such rules through the analysis of large corpora of typical real-world examples. A corpus is a set of documents that have been hand-annotated with the correct values to be learned. [6][7]

Table 1 Comparison of existing systems

<table>
<thead>
<tr>
<th>Factors</th>
<th>Estimated Price</th>
<th>NLP Input</th>
<th>Dynamic Learning</th>
<th>Multiple Language Support</th>
<th>Security</th>
<th>Google Search Capability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our System</td>
<td>$325</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Ubi</td>
<td>$299</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Voicepod</td>
<td>$649</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Ivee Sleek</td>
<td>$229</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>mControl</td>
<td>$170</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>HAL2000</td>
<td>$660</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
System Description

The focus of this work is more to decipher natural sentences spoken by the user using Natural Language Processing. Natural Language Processing is a field of computer science, artificial intelligence, and linguistics concerned with the interactions between computers and human natural languages. As such, NLP is related to the area of human-computer interaction. Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input, and others involve natural language generation. The working of the system as given in figure 1 is given below.

Fig 1 Data Flow Architecture

- Microphone receives input from user.
- Analog signals are sent to the Audio Interface (Focusrite Scarlett 2i2).
- Audio Interface converts audio signals to Digital Signals
- Audio data is converted into text.
- Text parsing is carried out using a corpus.
- Select tokens are used for instruction mapping.
- Commands are sent to the switch for carrying out the required operation.

The hardware components used and their connectivity is shown in figure 2. The connections are made as follows:

- Blue Spark is connected to the Focusrite Scarlett 2i2 using an XLR cable.
- The Focusrite Scarlett 2i2 is connected to the Raspberry Pi via USB.
- The Raspberry Pi is powered using a Belkin USB Power Hub.
- The Raspberry Pi is connected to the Arduino Uno R3 via USB.
- Breadboard is connected to the Arduino using a relay system.
- There are multiple resistors and a single LED mounted on the Breadboard.

Fig 2 System Architecture
Hardware Requirements

- The Raspberry Pi serves as the main device (server) and is responsible for receiving the audio signals from the microphone phone, converting them to text and mapping them to instructions to be sent to the switch (client).
- The Focusrite Scarlett 2i2 is the audio interface which helps in transferring the audio signal from the microphone to the Raspberry Pi. It is a USB powered interface with phantom power. This interface has two inputs and one audio output.
- The Blue Spark is the microphone used in this system to receive the instructions from the user. The audio signal received is transferred to the audio interface (Focusrite Scarlett 2i2) which is then sent to the Raspberry Pi.
- The Arduino Uno R3 is the switch (client) in this system. The Arduino Uno R3 is connected to the Raspberry Pi via USB. The instructions sent from the Raspberry Pi are received by the Arduino which is in turn sent to a breadboard with RGB LEDs.
- Xbee is used in this system to wirelessly connect the Raspberry Pi to the Arduino. One Xbee shield is connected to the Raspberry Pi while another is connected to the Arduino and the connection is made wirelessly using the Zigbee protocol.

Software Requirements

- The NLP engine is programmed on the operating system developed solely for the Raspberry Pi called the Rasbian.
- Sphinx is the speech recognition system used for converting the speech received from the microphone to text. Sphinx is a realtime speech recognition engine making use of Hidden-Markov Models.
- This system uses Arduino IDE a cross-platform application written in Java, to program the instruction mapping functions on the arduino.
- Firmata is used for communication between the Main Device (Raspberry Pi) and the Arduino.
- Racket is used in this system to send wireless serial commands to the Arduino using a protocol called Firmata.

Modules of the System

A. Listener - The listener is the first stage of the entire process. It waits for the audio signal. Once it identifies the audio signal, it differentiates between voice commands and noise. This is the pocket-sphinx application that uses continuous listening with silence filtering to automatically segment a continuous stream of audio input into utterances that are then decoded. Each utterance is ended when a silence segment of at least 100 milliseconds is recognized. All of the audio received in this module is sent to the Noise Removal module of this system.

B. Noise Removal - This is the first part of the Speech to text conversion process. Noise Removal is an essential part of differentiating between noise and coherent words and instructions. This noise removal algorithm is completely integrated into the Pocketphinx to increase the efficiency of the speech-text operation. The noise need to be differentiated with the speech patterns and syllables of the user issuing the voice commands which makes the noise removal algorithm an essential step before the speech to text conversion happens.

C. Hidden Markov Models - The speech to text conversion is done using the Hidden Markov Models (HMMs). When an audio signal of a word is received by the Raspberry Pi, it is broken down into its individual syllables by this module. HMMs are used because they provide a simple and effective framework for modelling time-varying spectral vector sequences. This is important because the way a user speaks and stresses on different syllables has slight variance every time a word is spoken. To account for these differences the HMMs take into account the stress on each
individual syllable and then converts it into a specific word by taking into account the words in the corpus.

Figure 3 illustrates the procedure of using HMMs to process the word “Six”. When the user mentions the word “Six” the speech to text conversion splits it into its individual constituents, which are the syllables ‘s’, ‘ih’, ‘k’ and ‘s’ again. The repetition of syllables occurs when there is an extra stress in the different syllables although it doesn’t change the meaning of the word.

D. Viterbi Algorithm - The search manager commonly uses the Viterbi algorithm. This algorithm is a form of Optimized graph search (heuristic) for finding the most likely sequence of hidden states (phones) in a HMM based on a sequence of observations over time. This leads to it containing a set of confidence values which are especially important when considering the accuracy in understanding speech patterns and uses of syllables. The most important thing is that the confidence values need to increase to help the correct alternatives get chosen.

E. Corpus - The Natural Language Processing in this system is achieved using a corpus. A Corpus or text corpus is a large and structured set of text. The corpus basically contains all of the text which can be understood by the system required. If there is any word mentioned by the user which is not present in the corpus, then the system will not understand it.

RESULTS AND DISCUSSION
The speech to text based system is a phonetic based system and its accuracy is based on multiple factors. The speech to text system has been optimized for real time use. It achieves 90-95% accuracy post usage. Test runs (fig 4) were carried out with male and female participants. And the results averaged 90% average success rate. 30 runs for male and female participants were conducted out of which 27 and 26 of them were declared successful. The end result averaged 90% and 86.67% respectively for male and female participants.

<table>
<thead>
<tr>
<th></th>
<th>Test Runs</th>
<th>Successful Tests</th>
<th>Success Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>MALE</td>
<td>30</td>
<td>27</td>
<td>90%</td>
</tr>
<tr>
<td>FEMALE</td>
<td>30</td>
<td>26</td>
<td>86.67%</td>
</tr>
</tbody>
</table>

Fig 4 Test Result Average

Trigger - One major issue in the system is the fact that the main device is always listening. This means that even general conversations between people is processed by the system. This problem can be solved by using a simple trigger to refer to the system ensuring that only when the user is referring to the system will it try to process conversations and perform commands. At all other times, it shall remain in sleep mode.
Removal of Time Delays - There were multiple time delays in the Development Kit leading to a response time of about 6-7 seconds after the initial testing phase. This was completely untenable for a real time system. The following steps were taken to ensure reduction in time delay.

CONCLUSION
The device presented in this paper the Voice Activated Home Automation System (VAHAS) intends to overcome the deficiencies of the existing devices that were surveyed.

- The VAHAS is largely an offline system, not requiring an active internet connection for its speech to text conversion. It thus aims to eradicate penetration issues in areas without sufficient bandwidth.
- Natural language is largely considered as the next definitive human interface mechanism, and trumps voice command input. The device aims to be easier to use than its predecessors by leveraging the power of natural language.
- The device presented in this paper introduces a new concept of dynamic learning, enabling the device to be personalised through use and making it capable of learning new tasks and intents.

Natural language processing was successfully implemented in this voice activated home automation system which makes the success of this process. This paper produced three consistent results:

- The system’s ability to receive audio and convert it to text.
- The ability to understand natural language.
- The ability to map instructions in accordance to the commands.

REFERENCES


A Cost Effective Implementation of a Voice Assisted Home Automation System
10.4028/www.scientific.net/AMM.704.390