

# AIVIES: An Artificially Intelligent Voice Interactive Enquiry System

Binny Khanna<sup>[1]</sup>, Harshil Shah<sup>[2]</sup>, Nikita Kushwaha<sup>[3]</sup>  
 Department of Computer Engineering  
 MPSTME, NMIMS University  
 Mumbai, India

**Abstract**— AIVIES is a voice interactive enquiry system, intended to aid the user with specific enquiries regarding travel and locating various eateries. The system aims to emulate regular conversation and interact with the user via speech recognition and synthesis. The task of enquiry is essentially concerned with human interaction and this system automates that process, thereby rendering the presence of a human as void. With the tremendous growth in natural language processing techniques, it has become a reality to implement speech recognition and synthesis, programmatically. Much like how existing systems provide response to mouse or keyboard inputs, this system responds entirely to voice. The system is an artificially intelligent, simple reflex agent and triggers specific events on the basis of the speech interpreted.

**Keywords**— *Natural Language Processing, Artificial Intelligence, Speech Recognition, Speech Synthesis, Enquiry*

## I. INTRODUCTION

In intelligent systems, a simple reflex agent is one that takes a certain input and executes an associated event on the occurrence of that input. This input could take one of several forms: sensors, gestures, speech etc. Fig.1 depicts a typical simple reflex agent:

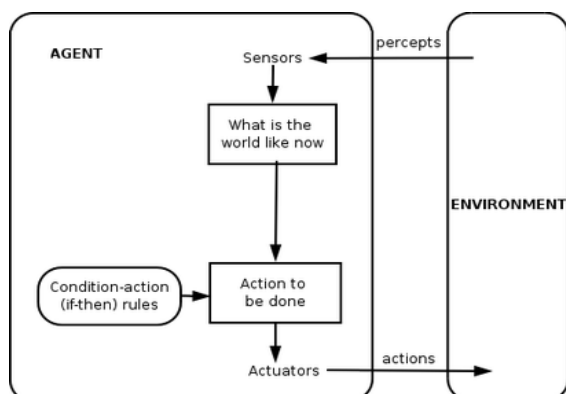


Fig. 1. A simple reflex agent

AIVIES is also essentially a simple reflex agent which uses speech recognition for its input. The 'percept' in this case is the recognized speech and the 'action' is the associated event that will be triggered as a response to the recognized speech. For example, if the percept is 'taxi enquiry', AIVIES will return results as fare, time and distance for the given source and destination combination. The source and destination are

also accepted via speech. The agent function for AIVIES can be given as:

$$[f: S^* \rightarrow E]$$

Where,

S\*- Speech Recognized

E- Event Associated with Recognized Speech

Another component of a simple reflex agent, apart from the percept and associated action, is a list of 'if-then' rules or 'condition-action' rules. This list contains the entire logic of the artificially intelligent agent. It tells the agent which action to execute associated with the interpreted percept. After the system receives an input from the environment, it scans its condition-action rules, locates the action associated with the input it just recognized and then executes the associated event. Fig.2 depicts these components for AIVIES as:

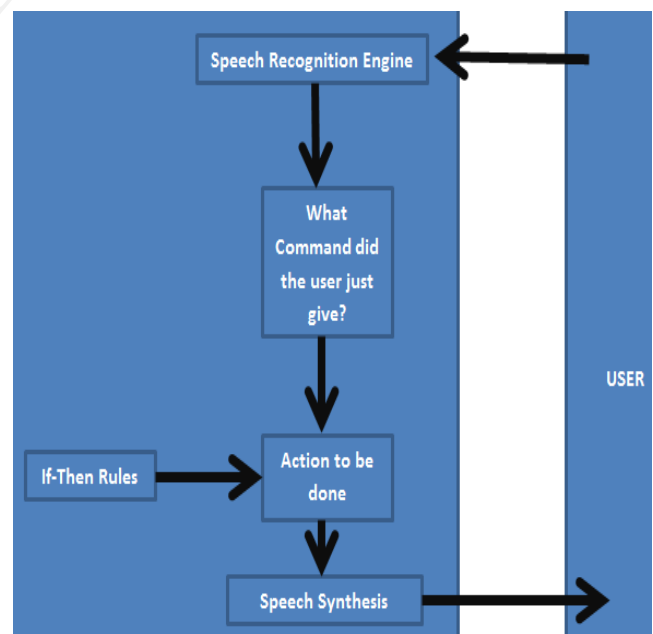


Fig. 2. AIVIES as a reflex agent

As depicted in Fig.2, for AIVIES, the percept is the speech command, the sensor is the speech recognition engine, the actuator is the speech synthesizer, and the action is the event indicated in the if-then rules table. Some of the condition-action rules for AIVIES are shown in table 1:

TABLE 1: CONDITION-ACTION RULES FOR AIVIES

| If (Speech Command)          | Then (Action)   |
|------------------------------|---|
| Make New Enquiry             | Clear all fields and being new enquiry process  |
| Source source_name           | Enter source_name in source field   |
| Destination destination_name | Enter destination_name isdestination field  |
| Make taxi enquiry            | Display fare, time and distance for the 'taxi' mode of transport and the given source and destination combination |
| Send Mail                    | Send current enquiry details to the current account e-mail address  |
| Show time                    | Display current time  |
| Close Application            | Exit System   |

## II. METHODOLOGY

### A. Environment

The system uses Microsoft's SAPI (Speech Application Programming Interface) for the purpose of both speech recognition and synthesis. Dragon's Naturally Speaking and Carnegie Mellon's Sphinx were considered but SAPI yielded the best results. AIVIES is developed in the Microsoft.NET version 4.5, using C# for the program implementation. Two specific libraries were incorporated from SAPI- one for speech recognition and the other for synthesis. Both libraries are extremely rich providing extensive functionality and usability for the application.

### B. Flow of the Application

Once the application is loaded the speech recognition engine is 'live'. The 'grammar' has to now be loaded. The 'grammar' is the part of the speech recognition engine which contains a list of the expected phonemes, words and phrases. The engine can only recognize the words contained in this grammar. As such, it will try to match what the user has said to its existing grammar only. Nothing, outside of the grammar can be interpreted. Thus, the grammar for AIVIES primarily contains the speech commands and the phonemes associated with these commands. Once AIVIES is live, it will try and interpret everything the user says through the microphone. If the speech recognized passes a certain threshold value, the speech input is accepted. On the contrary, failing to surpass the minimum confidence value will cause the speech to be rejected. In case of speech rejection the system will display some possible alternate phrases. However, the speech rejection is rare and the system is highly accurate with very few misinterpretations. Once the speech is recognized, AIVIES will look up the condition-action table and execute the action associated with the recognized speech. The execution of the action can occur in one of two ways. If the event is within the scope of the application, the application will call our event processing function. On the other hand, if the event requires interaction with the operating system (For speech commands, such as 'switch tab', 'scroll

up' etc.), the application will call the OS-interacting function. At this point, during the execution of the event, AIVIES must also provide a speech response to the user, letting him know what exactly is being processed. This helps establish a conversational flow with the user. Fig.3 depicts the flow of AIVIES:

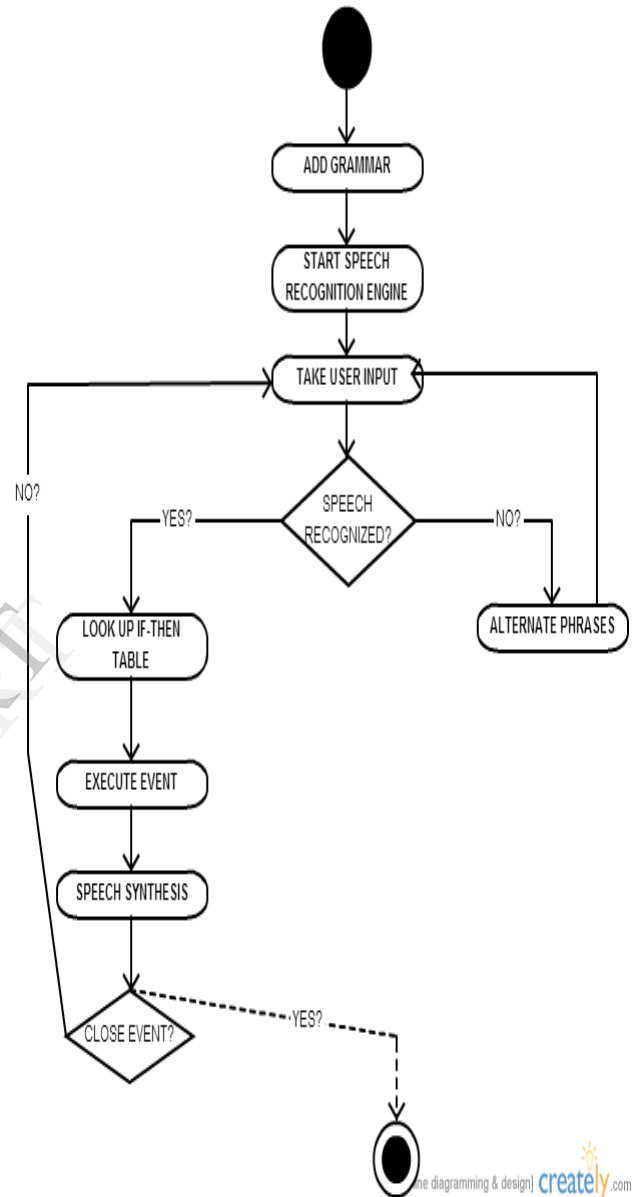


Fig. 3. Aivies flow diagram

The process of speech interpretation continues until the user says the command 'close application' or closes AIVIES manually.

## III. The application

We will now look at some of AIVIES' applications. Once the user has logged in, he may now make one of several different kinds of enquiries mentioned within the help page of the application. The application also has a tutorial option to guide the user through its exact usage. In general the system responds to three kinds of commands: General commands,

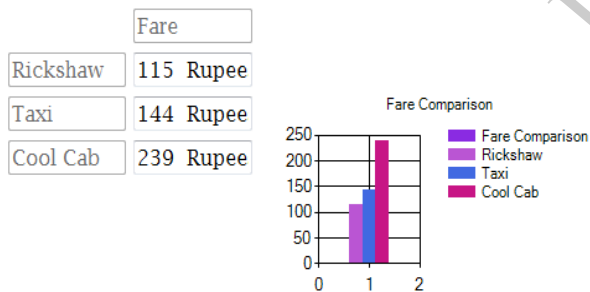
travel enquiry related commands and restaurant enquiry related commands.

#### A. General Commands

AIVIES responds to a wide range of general speech commands, such as 'maximize', 'scroll up', 'display commands', 'hide commands', 'switch window', 'close application' and more. These commands require interaction with Windows. We have developed a function to interact with the OS that passes the appropriate set of keys to Windows. Thus, instead of using the keyboard to navigate, the user's speech serves this purpose. The intent has been to keep the speech commands as intuitive as possible for the lay user.

#### B. Travel Enquiry Commands

If the user initiates a travel enquiry, AIVIES will prompt the user to state his source and destination. It will further establish the mode of transport the user wishes to use. Since, the application is for the city of Mumbai, these will include: taxis, rickshaws, buses etc. Once the user has provided these parameters via speech, AIVIES will return results related to the distance, fare and approximate time of the mentioned commute. It also provides additional actions to the user, such as e-mailing these details or comparing the various modes. Simply saying 'send travel mail', sends all the details associated with the enquiry instance of the user's e-mail address in a completely automated manner. Fig. 4 depicts a comparison chart generated by AIVIES in response to an enquiry. The enquiry is for a commute between two suburbs- bandra and andheri. The comparison is of cost for three different modes of transport:



[Save Chart?](#)

Fig. 4. Comparison Chart

All voice commands are color coded to blue. This implies that anything written in blue, is directly a voice command. Besides, the visual prompt, AIVIES will also prompt the user via speech.

#### C. Eatery Enquiry Commands

The system also aids the users in locating eateries associated with a given set of parameters. These parameters include: budget, area and cuisine. AIVIES will return to the user three eateries best matching his criteria. The choice is based on an algorithm that provides weightage to the different criteria, with a strict adherence to the budget, slight flexibility in the area and a factor for user reviews and quality. The user can

also ask for more information. Saying 'more information' will directly open the websites of the eateries found by AIVIES.

#### D. Miscellaneous

Other actions include a 'help' page that lists the various commands and the events they will cause. A tutorial mode exists to help the user understand how exactly to use AIVIES. In this mode, AIVIES will guide the user on every command they give and its usage. The user can also make notes, see the time, see the date, see the color code and display their saved chart. All of these are also speech commands, of course. A comprehensive list of some the commands AIVIES recognized are depicted in Fig.5 below:

| Command          | Action  |
|------------------|---|
| Call Cab         | Opens the web pages associated with cool cabs   |
| Cool Cab Enquiry | Display details associated with the source-destination combination for the cool cab mode of transport |
| Clear All        | Clears all textboxes  |
| Hide Commands    | Hides the command listbox   |
| Homepage         | Returns to the homepage   |
| Rickshaw Enquiry | Display details associated with the source-destination combination for the cool cab mode of transport |
| Make Note        | Opens notepad for you to write any detail   |
| Make New Enquiry | Clears the current instance of enquiry, enabling the user to make a new one                           |
| Maximize Please  | Maximizes the form state  |
| Minimize Please  | Minimizes the form state  |
| Returns          | Same as homepage  |
| Save Chart       | Saves the last enquired chart   |
| Show Commands    | Displays the commands list box  |
| Show Comparison  | Compares the cost of the three modes of transport   |
| Show the Time    | Displays the current time   |
| Show the Date    | Displays the current date   |
| Show Color code  | Displays the color code   |
| Switch tab       | Switches the current tab  |

Fig. 5. Help page for AIVIES

When the user says "show help page", or "help" this help page opens up. Again this page is interactive in the sense that it can dictate the contents of the page to the user. It can also, responds with the action for the user stating any of the commands, while on the help page. Thus AIVIES has facilities for providing travel details, eatery details, computer navigation, comparison charts and extensive help to guide the user to a very natural communication process with the system.

#### IV. CONCLUSION

As Speech Recognition technology improves in terms of accuracy, vocabulary, and its ability to understand natural language, we will see the concept of interactive machines in every arena. From assembly line mechanical tools to intelligent microwave ovens to "writing" a check, we will

have the power to use our voice to instruct the electronic devices we encounter with daily.

It is apparent that humans are trying to create a computing environment in which the computer learns from the user instead of one where the user must learn how to use the computer. Speech Recognition technology is the next obvious step in an attempt to integrate computing into a "natural" way of life.

This effective means of communication, even when perfected, will still present limitations as to how humans can express themselves. With the immense advancements in natural language processing and speech synthesis, it is but evident that speech recognition is the future.

For the years an attempt has been made to interact in more natural means with the computer. Even if such systems existed earlier, application of them in building interactive systems was limited. However, it is now completely possible and must be exploited.

The immense advantage of speech recognition is that it is intuitive to the user. Sure, we have adapted to the use of keyboard and the mouse; however, these are not intuitive devices and took us a long time to adapt to. A well-made speech recognition system can guide even a first time user with ease.

Our aim with this project was to build a system that incorporated these speech recognition techniques and provided for a sound and flawless detection pattern that smoothed the process of communication.

In light of this, AIVIES is an attempt in incorporating speech in the applications we use and using only speech for all communication purposes. In the future, this system can be extended to serve various other kinds of enquiries relating to cinema, shows, domestic flights etc.

This application is a step toward the more natural forms of human-computer interaction, imminent in the near future. At some point in the future, speech recognition may become speech understanding. The statistical models that allow computers to decide what a person just said may someday allow them to grasp the meaning behind the words. Although it is a huge leap in terms of computational power and software sophistication, some researchers argue that speech recognition development offers the most direct line from the computers of today to true artificial intelligence.

#### V. ACKNOWLEDGMENTS

We would like to acknowledge our professor for C#: Mr. Steven Mankina, for teaching us the requisite programming language and familiarizing us with the .NET framework. His expertise was of great benefit to us.

#### REFERENCES

- [1] F. Reena Sharma, S.Geetanjali Wasson, "Speech Recognition and Synthesis Tool: Assistive Technology for Physically Disabled Persons," International Journal of Computer Science and Telecommunications, vol.3, issue 4, April 2012.
- [2] M.A. Anusaya, S.K. Katti, "Speech Recognition by Machine: A review," International Journal of Computer Science and Information Security, vol.6, no.3, 2009..
- [3] B. Singh, N.Kupar, P. Kaur, "Speech Recognition with Hidden Markov Model: A Review," International Journal of Advanced Research in Computer Science and Software Engineering, vol.2, issue 3, March 2012.
- [4] J. Picone, "Fundamentals of Speech recognition," unpublished.
- [5] P.Saini, Parneet Kaur, "Automatic speech recognition: A Review," International Journal of Engineering Trends and Technology, vol. 4, issue 2, 2013.
- [6] L. Schindler, "Web Based Education: A Speech Recognition and Synthesis tool," unpublished.
- [7] L.F.Hodges, "Interacting with Virtual Humans through Voice Recognition and Interaction", unpublished.