International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 05 Issue: 12 | Dec 2018 www.irjet.net p-ISSN: 2395-0072

Voice to Code Editor Using Speech Recognition

Anurag Singh[1], Ganesh Tambatkar[2], Shashank Hanwante[3], Nitish Agrawal[4], Rahul Hajare[5], Ketki Khante[6]

[1]Team Leader & S.B Jain Institute of Technology, Management and Research, Nagpur
[2,3,4,5]Team Member & S.B Jain Institute of Technology, Management and Research, Nagpur
[6]Project Guide, Dept. of Information Technology Engineering, S.B Jain Institute of Technology, Management and Research, Maharashtra, India

---

Abstract:- *In current speaker-independent speech recognition, it is impossible to attain a high recognition ratio for general purpose applications. However, it is possible to obtain a reasonable recognition ratio for specific applications. A programming language has the specific syntax and vocabularies specified by the language specification based on these characteristics. A Java program editor by voice input is implemented. Your voice is the most feasible way to manifest. Speech recognition does the job readily and simply. Commands can be chained and nested in any combination, allowing the complicated procedure to be accomplished by a single oral phrase. By taking advantage of your brain's natural aptitude for the language you can code more accurately and instinctively.*

***Key Words*: Acoustic and language modelling, Hidden Markov Model**

1. INTRODUCTION

Speech Recognition, it is the ability of the machine or program to evaluate word, idiom or a sentence in spoken expression and convert those words into a machine-readable format. The more sophisticated software has the ability to obtain natural language as well. Speech recollection works using algorithms through acoustic and language modeling.

In addition, acoustic modeling represents the link between linguistic units of speech and audio signals; whereas language modeling matches sound with string to help categorize between words that sound similar. Additionally, Hidden Markov Models are used as well to make materialistic patterns in a speech to enhance accuracy with the system.

Furthermore, it is seen that a person working on a computer cannot work or type for longer duration because if they can then there will be an issue of back or wrist pain that will be pernicious for the human body, but it can be avoided easily by switching from typing to speaking whenever needed.

1.1 PyAudio

It's a Python by default speech recognizer engine which is very helpful and can recognize a phrase or sentence easily and we can also improve its productivity by defining macros so that its understandability should get improve. With the help of it, there will be fewer spelling errors in a code, will enhance the speed resulting in greater mobility and a person with the disability will also get a chance to operate a computer.

2. OBJECTIVES

- To create an editor that will perform all basic tasks.
- To create a more precise editor there will be an option for the user, i.e.; either type else speak the code.

3. PROBLEM STATEMENT

Creating an editor that have options for user-reliability like typing or writing the code through their voice.

4. LITERATURE SURVEY

Speech recognition is the capability of a machine or program to analyze words and idioms in spoken language and convert them to a machine-readable format. Among the earliest utilizations for speech recognition were mechanical telephone systems and medical dictation software. It is often used for transcription, for querying database, and for giving commands to computer-based systems, specifically in professions that rely on specialized vocabularies. It also implements personal companion in vehicles and smartphones, such as Apple's Siri.

Nitin Washani, Sandeep Sharma," Speech Recognition System: A Review", In this paper, they have classified the system into Front End Analysis and Back End Analysis for better understanding and representation of speech recognition system in each part. The main reason for them to classify the system is that it will provide higher accuracy because they were facing issues related to noise, vocabulary size, and domain so they came up with their idea of Front-End Analysis and Back End Analysis.

Voice code, "Advanced Voice Control Platform", in these articles, writer faced some issues because of working on computer for longer duration, so they came up with the idea that, by taking advantage of our brain's natural aptitude for language we can control our computer more efficiently and naturally, so they created a plugin for UNIX based system. Nuzhat Atiqua Nafis, Md. Safaet Hossain, "Speech to Text Conversion in Real-Time", In this paper, they have used a method to design a text-to-speech version module by the use of Matlab. That method is simple to implement and involves the much lesser use of memory spaces.

5. PROPOSED WORK

We are going to implement a project using Python language which is the most suitable language for using Speech Recognition and in addition to that, we will be using PyQt4 for creating an Editor that will enhance our project working i.e. converting voice to the code editor. There is also an alternative way to develop the editor and its functionalities by the use of Electron JS. Electron is an open source library developed by GitHub for building cross-platform desktop applications with HTML, CSS and JavaScript.

Electron is an open source library developed by GitHub for building cross-platform desktop applications with HTML, CSS, and JavaScript. Electron accomplishes this by combining Chromium and Node.js into a single runtime and apps can be packaged for Mac, Windows, and Linux. The calculation and logical implementation can also be implemented using python.

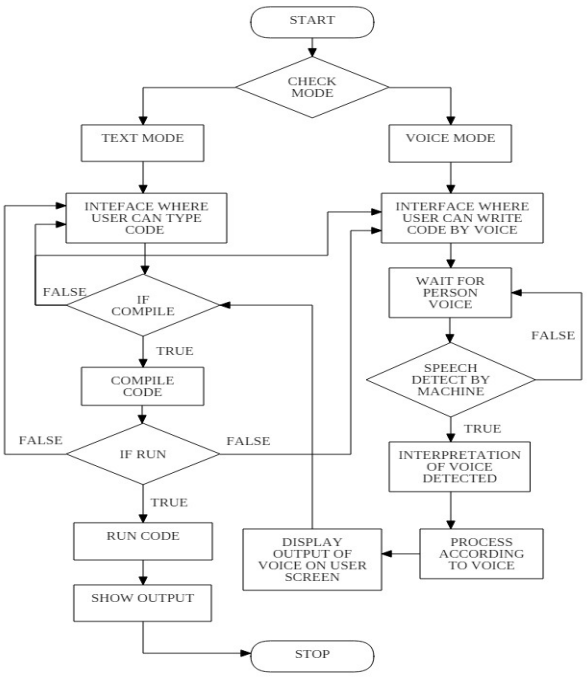

Fig 1: Flow diagram of voice to code editor

The project Voice to Code Editor will have 5 different modules –

- Language Specifying Module – In here, when the user creates a new file, he will be asked that in which language they want to use voice to code editing so that it can fetch the predefined syntax.
- Text Operation Module – Like, every editor, this editor will also perform tasks like reading, write, cut, copy, paste, move to line number etc.
- Commands Specification Module – In here, whatsoever the user speaks, a word or a sentence, it will be taken as a command so that it can perform tasks accordingly.
- Differentiate Text and Symbol Module – It is very important to classify text and symbols so that they cannot create an ambiguity. (Ex – equals to or "=")
- Compile and Run – Similar to every editor, compile and run is necessary so that it can display the output and if the code containing an error in any line, it can specify with line number with its description as well.

6. CONCLUSIONS AND FUTURE SCOPE

6.1 Conclusion

Conclusions that have been made: The editor will work leniently when there are many words or a sentence said by the user without any ambiguity and if any error found it will display the error with line number with its description and if nothing then the output will be shown accordingly.

6.2 Future Scope

Currently, we are developing the project that will work only single language i.e. on Java, but in future, we will build our editor or our project in such a way that it will take inputs of more languages like C, C++, HTML, CSS, JavaScript etc.

REFERENCES

[1] Nitin Washani, Sandeep Sharma, "Speech Recognition System: A Review", International Journal of Computer Applications (0975-8887) Volume 115-No. 18, April 2015.

[2] Nuzhat Atiqua Nafis, Md. Safaet Hossain, "Speech to Text Conversion in Real-Time", International Journal of Innovation and Scientific Research, ISSN 2351-8014 Vol. 17 No. 2Aug. 2015, pp. 271-277.

[3] "A Real time speech to text conversion system using bidirectional Kalman filter", in Matlab. 2016 Intl. Conference on Advances in Computing, Communications and Informatics (ICACCI), Sept. 21- 24, 2016, Jaipur, India.