Report of Desktop Voice Assistant using Python Programming Language

Abstract - **In Today's world with continuous development in science and technology, it is becoming hard to predict what level of advancement and innovations humans will bring. But the experts say the recent future will be defined by Artificial Intelligence. And it seems very true that in near future AI will aid humans in every possible way. For this reason, we have focused on one application of AI in which machine would understand human's language, adjusting and interacting in it. So, we have developed Desktop Voice Assistant with python which allows users to do simple task of opening applications like google, YouTube, Wikipedia, Playing Music, Displaying date and time and various other applications on a PC. Not only this also a user can interact with it by asking questions like "How are you" or "Tell me a Joke". The main task of our voice assistant is to reduce the time to give input commands to open application and understand how the proposed system can change the way of interactions between end user and the device.**

Index Terms - Artificial Intelligence, Desktop Voice Assistant, Python, Text to Speech, Speech to text, Virtual Assistant, Artificial Speech Recognition.

.

I. INTRODUCTION

Artificial Intelligence is a method to make a computer think and a way to implement how human can think. AI is a study of how human brain think new ideas, learn things, make decision and work, when it tries to solve problems. And finally, this study outputs intelligent software systems. The aim of AI is to improve computer abilities which are related to process of human knowledge, for example, reasoning, learning, and problem-solving. The intelligence is intangible. It is composed of

- ❖ Reasoning
- ❖ Learning
- ❖ Problem Solving
- ❖ Perception
- ❖ Linguistic Intelligence

The aim of AI research is learning, knowledge representation, reasoning, planning, natural language processing, Pattern Recognition, Information Retrieval and move and control objects. The amount of data that is generated, by both humans and machines is exponentially large. Humans by no means have such great ability to absorb all that data, interpret, and make complex decisions based on given amount of information. As an example, most humans can think out how to win at tic-tac-toe. But few people would be considered grand masters of checkers game, with more than 500 x 1018, or 500 quintillion, different moves. Computers are extremely efficient at calculating combinations and permutations to come down at the best action to be taken. AI and deep learning models are going to be the future of business decision making.

Today every Multinational Company is adapting to Voice Assistant so that their user can receive assistance of machine through their voice. So, with the Voice Assistant we are moving to the new heights of advancement where we are able to speak to our machine. These types of virtual assistants are very helpful for senior citizen, blind & disabled people, children, etc. by making sure that the interaction with the machine is kept as easy as possible for people. Even blind people can interact with it using their voice only.

The Desktop Voice Assistant with Artificial Speech Intelligence, which takes the user input in form of voice which are audio waves and process it and returns the output in Text and then action is to be completed and the search result is presented to the end user.

Here are some of the tasks that can be implemented with the help of voice assistant: -

- Reading Wikipedia
- Composing an Email
- Search on web
- Play a music or video
- Setting a reminder and alarm
- Run any program or application
- Date and Time

These are some of the examples, we can do many more things according to our requirement. The Voice Assistant that we have developed is for Windows Users. The voice assistant we have developed is a desktop-based built using python modules, functions and libraries. We have used python modules and libraries for making this project and we have used Machine Learning for training our model, Basically, this model will easily run. Depending upon the usage for which the assistant is required for user. And these can be achieved with the help of Machine learning and Deep Learning. With the help of Voice Assistant there will be no need to write the commands repetitively for performing particular task. Once model is created it can be used any number of times by any number of users in the easiest ways

So, with the help of Desktop Voice assistant, we will be able to control many things around us without even the use of mouse or keyboard on one platform.

II. LITERATURE REVIEW

It is a generally accepted fact that human speech is very important to our routine personal and professional lives, and Speech-to-Text can be used in a lot of applications. Basic

audio data consists of sounds and noises. Human speech is an exception to that. In Speech-to-Text problems, the training data includes:

- Input features: audio clips of spoken words
- Target labels: a text transcript of said words

A. *Data Pre-Processing*

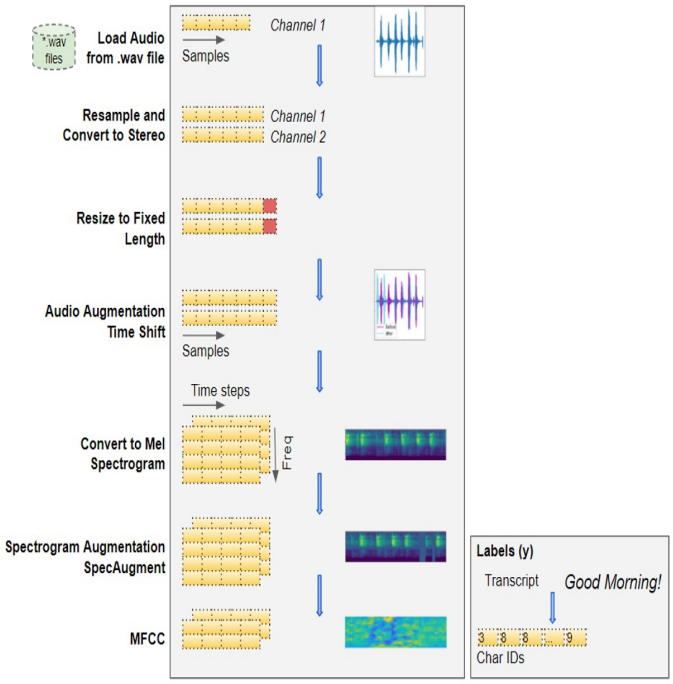

Fig. 1 Load Auto Files.

B. *Architecture*

A regular convolutional network consisting of a few Residual CNN layers that process the input spectrogram images and output feature maps of those images.

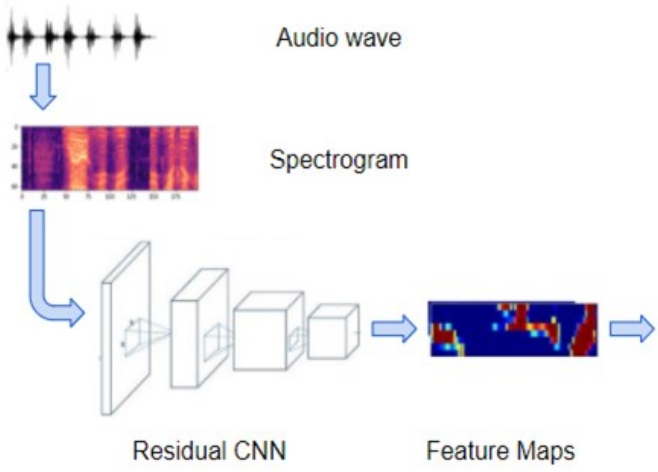

Fig. 2 Spectrograms are processed by a convolutional network to produce feature maps.

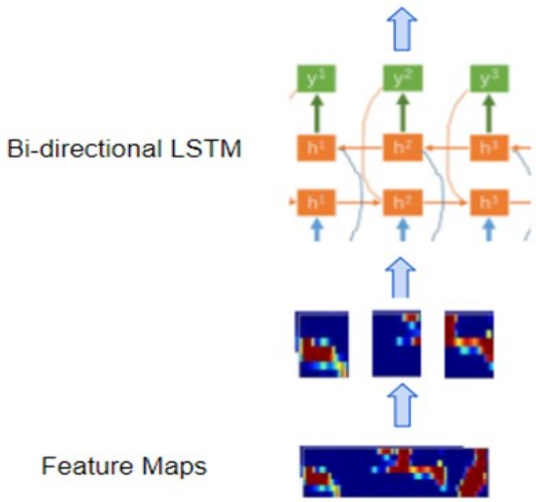

Fig. 3 Recurrent Network Process frames from the feature map.

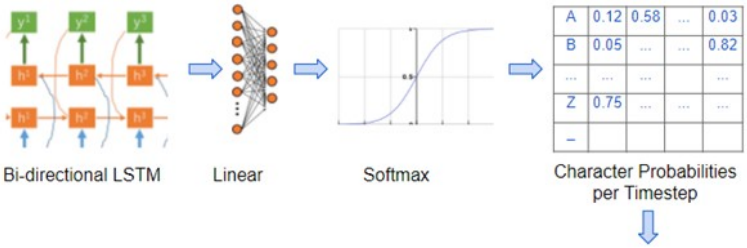

Fig. 4 Linear layer generates character probabilities for each timestamp.

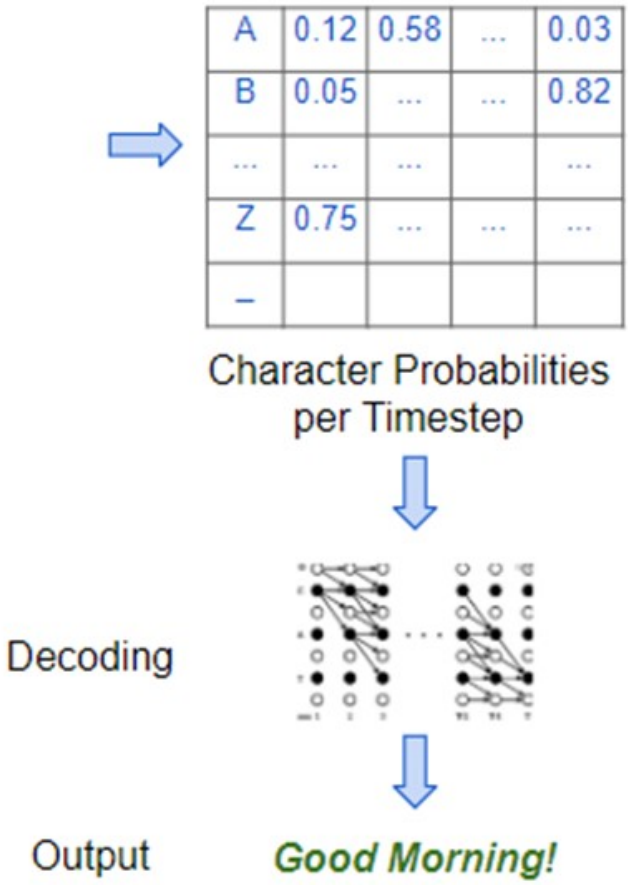

Fig. 5 The model decodes the character probabilities to produce the final output.

C. *CTC Algorithm*

The CTC is used to align the input and output sequences when the input is continuous and the output is discrete, and there are no clear element boundaries that can be used to check the input to the parts of the output sequence.

CTC works in two modes:

- CTC Loss (while learning): It has a ground truth target transcript and tries to teach the network to increase the probability of outputting that correct transcript.
- CTC Decoding (during Inference): Here, we don't have a final transcript to check with, and have to predict the most probable sequence of characters.

III. METHODOLOGY

A 12-step approach to implementing this project was used as follows

A. *Selection of programming language:*

There are various options when it comes to selecting a computer language to program in viz. C, C++, Java, Python, etc. After reviewing options, python was selected due to its simple syntax and greater integration with AI Based Modules.

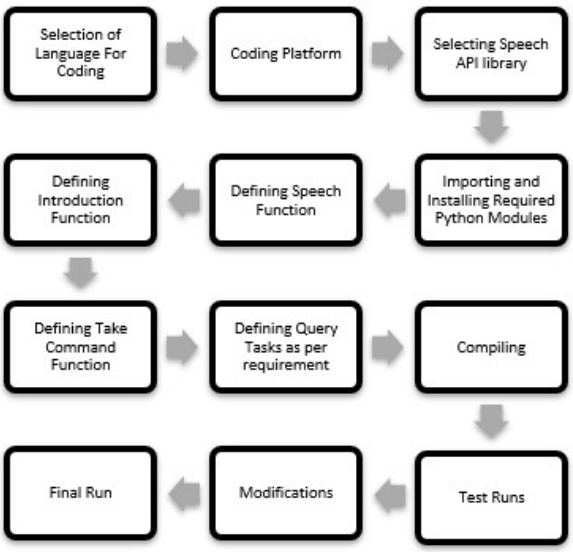

Fig. 6 Step-by-Step Methodology

B. *Coding Platform:*

Once Programming Language was decided, Coding platform for writing, Compiling and running the code was to be decided. Here, VS studio Code by Microsoft was selected keeping in mind the available modules and Speech Library integration.

C. *Selecting Speech API Library:*

The Assistant requires to speak multiple sentences as a part of Interactive Conversation to demonstrate AI Capabilities. For such a Speech API library is needed which contains all the required modules and pre-defined functions for speech capabilities.

There are various options pertaining to Speech API with different data sets and voice characteristics comparison of which are given further ahead.

After researching and shortlisting multiple APIs based on various parameters, SAPI 5 (Speech API 5.0 by Microsoft) was selected due to its inbuilt module integration with VS studio Code and its large and varied Database for speech capacities.

D. *Importing and installing required modules:*

VS Studio Code has some inbuilt and some add-on modules for command routines pre-defined. Some of which are used in this project viz.pyttsx3(SAPI5 -Python integration),Wikipedia, Speech Recognition, OS, Date time etc. Some of these modules were inbuilt while some needed to be installed using pip command. Further, on these modules were imported in the beginning of code.

E. *Defining Speech Function:*

A speech function is needed to be defined for setting assistant speech characteristics like voice type, Volume, Etc.

F. Defining Introduction Function:

To enhance Interactive experience of user, the Assistant needs some Introductory Statements whenever user boots it up. For such an introduction function is developed which wishes the user according to time of day and with basic conversation.

G. Defining Takecommand Function:

The Assistant needs to listen to commands the user gives and recognize the speech input. This is achieved by defining a take command function which takes input via microphone and processes it via a Speech recognition module like google speech recognition and processes it into machine language. Here, it is required to define the natural language of speech input for e.g., English (India) as per user's choice.

H. Defining Query Tasks:

The assistant is required to perform multiple tasks as per definition. The computer takes these tasks in form of speech input and processes it in form of queries for modules imported.

Multiple queries can be defined according to requirements using ladder if-else loops.

There are multiple modules available for multiple queries or tasks and they can be defined as per requirement using standard syntax.

I. Compiling:

The Code is compiled to check for errors and compatibility issues using inbuilt python libraries.

J. Test Runs:

Multiple Test runs are conducted to check for bugs or glitches. Modifications required are identified.

K. Modifications:

Based on feedback from Testing phase, appropriate modifications and/or enhancements are added to the code.

L. Final Run:

The modified code is run for final stage to identify and rectify any errors if found.

This Completes the entire methodology for implementing the project.

IV. RESULTS AND DISCUSSIONS

1. The code is compiled and run successfully.
2. The Assistant introduces itself correctly and is accurate enough to understand simple commands.
3. The assistant performs given tasks successfully and can perform various functions such as reading out Wikipedia Article Summary, Opening Applications, Playing Music, Opening Websites, Opening Files and Folders and tell the Date and Time.
4. The Code is modular and can be edited as per user's requirements for multiple varied tasks.
5. Similar Architecture can be used to implement Machine learning based Assistant code which can learn user's usage patterns and enhance itself accordingly.

V. COMPARATIVE ANALYSIS

In this section, we will be comparing the Desktop Voice Assistant powered by Artificial Intelligence which includes Natural Language Processing and Google Speech Recognition with respect to the widely used modern Voice Assistant which uses certain Machine Learning Algorithm and level of layers to process the information and produce the output.

Parameters	**Desktop Voice Assistant**	**Siri**	**Google Voice Assistant**
Algorithm	Uses Artificial Intelligence which includes Natural Language Processing and Speech Recognition.	Uses Artificial Neural Network such as Deep Neural Network (DNN) which has multiple layers between input and output.	Uses Deep Learning Algorithm such as Long-Short Term Memory (LSTM) which is an Artificial Recurrent Neural Network (RNN) Architecture.
Learning	It uses Artificial Intelligence which includes a set of pre-defined commands and various libraries in order to perform functions.	-There are more than 2 layers which has a complex non-linear relationship which is used for classification as well as regression. -It has a great accuracy and precision and is mostly preferred.	-The learning takes place by using sequential data or time series. -Some of the variations include LSTM, BLSTM, MDLSTM and HLSTM. It provides accuracy in speech recognition and character recognition.
Privacy Policy	It has set of pre-defined commands through which communication takes place. It does not store audio and there is no cloud server involved.	It does not record audio of the users to improve itself. It uses Transcripts in which only the ones which should be important for analysis are sent to the cloud.	Users have an option to store their audio by using Voice and Audio Activity (VAA). Files are sent to cloud to improve its performance.
Voice Output	Uses Microsoft Voice Output in which David is used for male and Zira is used for female.	They are trained in both male and female voices.	Assistants are only trained using Female voice since they perform better.
Features	-Provides and speaks out Wikipedia information. -It can open limited applications based on number of if else statements used. -It can tell you the current time and date as well. -It can play music from streaming services such as YouTube or from your storage device. -It can play Videos, Movies from your storage device and open TV Shows websites. -Limited Conversation based on if else statements used.	-Provides Wikipedia information in the form of short article. -It can open all types of applications. -It can tell you the current time, set an alarm and timer, open stopwatch and make to-do list. -It can play music from streaming services such as Apple Music and can play Radio Station and read Audiobooks. -It can play videos, TV shows or movies on television streaming from Netflix. -Conversational Commerce.	-Provides Wikipedia Information. -It can open all types of applications. -It can tell you the current time, set an alarm and make to-do list. -It can play music from streaming services such as Spotify or Google Play Music and can play Radio Station and read Audiobooks. -It can play videos, TV shows or movies on television streaming from Netflix. -Conversational Commerce.

Efficiency	36.24%	47.29%	76.57%
Disadvantages	-It does not include Machine Learning which makes it limited to perform certain functions. -The creativity level is limited to certain extent and depends on the person who is programming it.	-The learning process is relatively slow and it requires large amount of data in order to perform better. -The error is propagated back to the previous layer becomes very small which makes the training process insignificant.	-Many issues evolve because of gradient vanishing and exploding problems. -It cannot process long sequences if using “tanh” or “relu” as activation function.

TABLE 1: COMPARATIVE ANALYSIS

VI. CONCLUSION

This project is a proof of concept for the architecture required to develop a complex AI-based and Machine learning enabled Virtual Desktop Assistant which could handle complicated tasks with speech input and upgrade its interactive interface according to users.