



SEEK WISDOM, ELEVATE YOUR INTELLECT AND SERVE HUMANITY !



SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

Digital signal processing

Project title: voice recognition

Date: Feb 20, 2022

Table of Contents

Introduction	1
Problem statement.....	2
Objective	3
Methodology.....	3
Simulation and Discussion.....	4
Conclusion	5
Reference.....	6

Introduction

Digital signal processing (DSP) is an important technology used in voice recognition systems. It is used to analyze, modify, and manipulate audio signals in order to recognize and interpret spoken language. DSP techniques are used to filter out background noise, amplify weak signals, and detect patterns in the audio signal. One of the most common DSP techniques used in voice recognition is **feature extraction**. This technique involves extracting features from the audio signal, such as the frequency, amplitude, and duration of the signal. These features are then used to create a model of the speaker's voice. This model is then used to identify the speaker's voice when they speak. Another DSP technique used in voice recognition is **pattern recognition**. This technique involves analyzing the audio signal to detect patterns in the signal. These patterns are then used to identify the speaker's voice. Pattern recognition is used to identify the speaker's voice even when there is background noise or other interference. In conclusion, digital signal processing techniques are essential for voice recognition systems. **Feature extraction** and **pattern recognition** are two of the most commonly used DSP techniques used in **voice recognition**. These techniques help to filter out background noise, amplify weak signals, and detect patterns in the audio signal to identify the speaker's voice.



Problem Statement

Accessing services or as simple as using smart phones or personal computers is not an easy Task for peoples with some disability. Voice recognition is undoubtedly one of the most exciting advancements in technology today and could greatly benefit individuals with disabilities. Voice recognition technology enables users to control electronic devices, like computers and phones, through simple voice commands. This makes using a computer or other device much easier for people with mobility issues, as they do not have to physically interact with the device. Additionally, voice recognition software can also be used to allow disabled people to easily access the internet, which can open up a world of resources and information for them. For the ordinary people also voice recognition has revolutionized the lifestyle.

On this project several python voice recognition modules are used to implement an effective voice recognition system.

Objective

General Objective:

The general objective of this project is to develop a method to implement voice recognition system using python.

Specific Objectives:

1. Investigate the most commonly used techniques and parameters for voice recognition.
2. Design and develop an algorithm for voice recognition.
3. Validate the developed algorithm in terms of performance and accuracy.
4. Implement the algorithm in an application that allows users to interact with the system.

Methodology

The methodology for this project will involve the use of both theoretical analysis and practical implementation to demonstrate the effectiveness of speech recognition module. The methodology for this project can be broken down into the following steps:

1. ****Data collection****: The first step in developing a voice recognition system is to collect a large dataset of speech samples. This dataset should include a wide range of voices, accents, and languages to ensure the system can accurately recognize speech from a diverse population.
2. ****Pre-processing****: Once the data is collected, it needs to be preprocessed to prepare it for analysis. This involves cleaning and normalizing the data, removing any irrelevant information, and ensuring that the data is in a suitable format for analysis.
3. **** Model Selection****: There are many different libraries that can be used for voice recognition, including neural Speech recognition, PyAudio, Pocket sphinx and some others. The most appropriate model will depend on the size and complexity of the dataset, as well as the specific requirements of the project.
4. ****Model testing****: The model needs to be tested on a separate dataset to evaluate its accuracy and performance. This testing phase should include both quantitative metrics, such as accuracy and precision, as well as qualitative evaluations, such as listening to transcriptions and identifying errors.

5. **Model Refinement**: Based on the results of the testing phase, the model may need to be refined and retrained. This can involve adjusting the model parameters, collecting additional data, or modifying the feature extraction process.

Simulation and discussion

Step 1: Installing and importing the necessary libraries

```
import sys
import pyttsx3 #pip install pyttsx3
import speech_recognition as sr #pip install speechRecognition
import datetime
import wikipedia #pip install wikipedia
import webbrowser
import os
import random
```

Step 2: Getting a reference to a pyttsx3.Engine instance

An application invokes the `pyttsx3.init()` factory function to get a reference to a `pyttsx3Engine` instance. During construction, the engine initializes a `pyttsx3.driver.DriverProxy` object responsible for loading a speech engine driver implementation from the `pyttsx3.drivers` module. After construction, an application uses the engine object to register and unregister event callbacks; produce and stop speech; get and set speech engine properties; and start and stop event loops.

```
engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
engine.setProperty('voice', voices[1].id)
```

Step 3: defining different types of functions necessary for the implementation.

Our functions enables us to break down or decompose a problem into smaller chunks, each of which performs a particular task.

speak(audio): This method returns the voice version of any string that is provided as argument.

wishMe(): This method returns what the code has to reply in the first place when it begins running. Like getting the exact time when the program starts running and it replies with a message according to the time.

takeCommand(): This is the place where all the magic comes into play, the **Speech_recognition** module will play its role. The three builtin functions of the **Speech_recognition** module: **Recognizer()**, **Microphone()** and **Recognize_google()** will play significant roles. This function returns the recognized speech as **query**.

performCommand(): This function performs the required thing after it takes the query that we got from **takeCommand()** method as an argument. It checks different **if-else** statements and performs the one that matches the query that we got from our **takeCommand()** function.

sendMessage(value): This function sends the text to telegram group that is given as an argument.

randomGenerator(): This function generates a random integer between 0 and 4 inclusively, and returns the string version of it.

Step 4: The built-in method `*"main"***` will finish our steps as it is an entry point of our python code when it starts running.**

```
if __name__ == "__main__":
    wishMe()
    while True:
        # if 1:
            query = takeCommand().lower()

        # Logic for executing tasks based on query
        performCommand(query)
```

Conclusion

In conclusion, voice recognition technology has come a long way in recent years and has become an increasingly important tool in our daily lives. From our phones and computers to our cars and home appliances, voice recognition allows us to interact with technology in a more natural and intuitive way. However, while the technology has made great strides, there is still much work to be done in terms of accuracy, accessibility, and inclusivity.

One of the main challenges facing voice recognition is overcoming the limitations of current algorithms and models. As machine learning and artificial intelligence continue to evolve, we can expect to see significant improvements in accuracy and reliability. In addition, efforts to improve accessibility and inclusivity are crucial to ensuring that everyone can benefit from this technology, regardless of age, language, or ability.

Despite these challenges, the potential benefits of voice recognition technology are clear. By enabling hands-free operation, improving accessibility for those with disabilities, and streamlining communication and interaction with technology, voice recognition has the potential to transform the way we live, work, and communicate. As we continue to explore the possibilities of this technology, it is important to remain mindful of its limitations and work towards developing solutions that are both effective and inclusive.

References

<https://pyttsx3.readthedocs.io/en/latest/engine.html>