**Lab\_2**

preranasam edited this page now · [6 revisions](https://github.com/preranasam/IoT/wiki/Lab_2/_history)

**Title**

Raspberry Pi based Keyword frequency detection system

**Introduction**

This lab was about creating a Raspberry Pi system to detect keywords from .wav file. If the sentence has a keyword repeated 4 times in the sentence, it will count it and display the number of times the keyword has occurred. We will be using SpeechRecognisition package. Recognizing speech requires audio input, and SpeechRecognition makes retrieving this input easy. Instead of having to build scripts for accessing microphones and processing audio files from scratch, SpeechRecognition will have you up and running in just a few minutes. The SpeechRecognition library acts as a wrapper for several popular speech APIs and is thus extremely flexible. One of these—the Google Web Speech API—supports a default API key that is hard-coded into the SpeechRecognition library. That means you can get off your feet without having to sign up for a service.

**Objective**

1. To write a python script that will detect the keyword from .wav file and display it on screen.
2. To convert the .wav file (sound) to text.
3. To count the number of times the keyword has occurred and display it on screen.
4. Use live speech from a microphone to detect the keyword and display its frequency.

**Approaches/Methods**

Our first approach was to find an appropriate .wav file. Then convert it into text data. Then process the text data to count the number of times the keyword occurs and display it on screen. We then proceeded to record live speech using a microphone.

**Work Flow**

1. Make sure the Raspberry Pi has following requirements fulfilled:

* Python 2.7, or 3.3+ (required)
* FLAC encoder (required only if the system is not x86-based Windows/Linux/OS X)
* PyAudio (required only if Microphone is used)

1. Install Flac by the command: sudo apt-get install flac
2. Install python SpeechRecognition package using command: pip install SpeechRecognition
3. Write a python script to convert audio into text. Use 'recognizer' class to recognize the audio.
4. Read the audio .wav file using AudioFile(). Record the audio file using
5. Define keywords to be counted.
6. Use the 'recognize\_google()' method of recognizer with default Google Web Speech API. The recognize\_google() method will throw a 'speech\_recognition.RequestError' exception if the API is unreachable.
7. Process the string to count the keywords and display it on screen.
8. To record the audio using a microphone, use Microphone() class of speech\_recognition.
9. Use Listen() class to record the live audio.
10. We have used lcddriver() library to interface LCD with Raspberry Pi and display the frequency of words on it. We require to install 'sudo apt install python-smbus' and use the 'lcddriver.py' and 'i2cdriver.py' files. We also require to enable the I2C support in Raspberry Pi.

**Circuit Diagram**



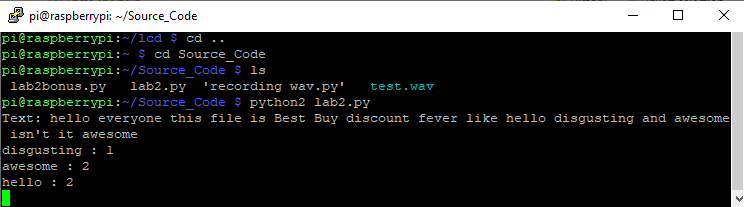


**Video**

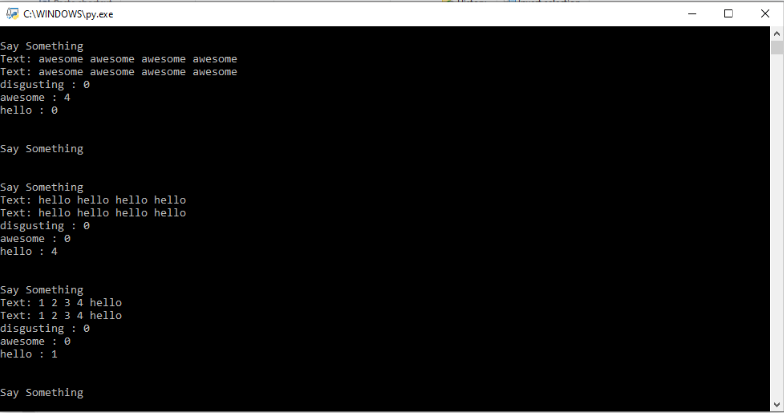
[RaspberryPi\_based\_Speech\_Frequency\_Detection](https://youtu.be/90YdmCwjRNM)

**Parameters/Output**

1. Using .wav file



1. Using Microphone



**Evaluation and Discussion**

Initially, we were facing issues with the installations of different libraries. We then were able to run the python script for conversion. We also faced problems with the lcddriver() for interfacing the LCD. But, then we used all the required files for LCD and placed them in the same folder as that of the code. Using a microphone was tricky as it was giving various errors.

**Conclusion**

Through this Lab assignment, we learned how to use the Google Speech Recognition class for converting the audio into text. Interfacing a microphone and counting the frequency of keywords in live audio was tricky.

**Github Links:**

1. [Prerana](https://github.com/preranasam/IoT/wiki/Lab_2)
2. [Zeal](https://github.com/zealpatel1990/iot_fall_2019/wiki)
3. [Ashish](https://github.com/akkipant/IoT-Fall-2019/wiki)
4. [Lee](https://github.com/sflcoder/IoT/wiki)