NATURAL LANGUAGE PROCESSING USING PYTHON

[1]Vismaya V, [2]Darvin Reynald J
[1]Student (B.Tech) - Department of IT, Sri Krishna College of Technology, Coimbatore
[2]Student (B.Sc) - Department of Computer Science Application & Software Systems,
Sri Krishna Arts & Science College, Coimbatore
mayadevan1210@gmail.com, wowdarvin@gmail.com

Abstract-This paper focuses on a simplified Natural Language Processing (NLP) system using Python and Raspberry Pi. Natural language processing systems have been used in a wide range of tech industries ranging from medical, defense, consumer, corporate. Most NLP systems used currently requires a subsidiary processing hardware and a default OS. The system proposed in this paper is a standalone NLP system which is open source and can be accessed in remote locations using a simple hardware component. The processes including voice extraction, speech to text conversion, text processing and database management and speech synthesis have been explained in detail along with the python modules used to build the system. By minimizing the hardware components and using open source software, a universal, adaptable NLP system has been proposed.

Keywords: NLP (Natural language processing), RaspberryPI, speech to text conversion, synthesize.

I INTRODUCTION

Natural Language Processing (NLP) is an area of application and research that explores how computers can be used to understand and manipulate natural language speech or text to do useful things. The foundation of NLP lie in a number of disciplines, namely, computer and information sciences, linguistics, mathematics, electrical and electronic engineering, artificial intelligence & robotics, and psychology. NLP researchers aim to gather knowledge on how human beings use and manipulate natural languages to perform desired tasks so that appropriate tools and techniques can be developed. Applications of NLP include a number of fields of study such as multilingual and cross-language information retrieval (CLIR), machine transaction, natural language, text processing and summarization, user interfaces, speech recognition, artificial intelligence and expert systems.

II LITERATURE REVIEW

NLP researchers aim to gather knowledge on how human beings tend to understand and use the language so that appropriate tools and techniques can be developed to make computer systems understand and manipulate natural languages to perform the desired [1][4] Phonological rules are captured through machine learning on training sets. Pronunciation dictionaries are also used for both text-to-speech and automatic speech recognition. Sounds as well as words can be predicted by using the conditional probability theory [7][6] the input to a speech recognizer is a series of acoustic waves. The waves are then sampled, quantified and literally converted to spectral representation. The method of Conditional probability is then used to evaluate each vector of the spectral representation with a system of stored phonetic representation. Decoding is the process of finding the optimal sequence of input observations. Each successful match is later used in embedded training – a method for training speech recognizers. [2] [3] Python and NLTK Module are mandatory for the following tasks. NLTK module is included as follows:

```
Part of Speech tagging and categorizing words >>> text = nltk.word_tokenize("And now for something completely different") >>>nltk.pos_tag(text)
```

Table 1 Part of Speech tagging and categorizing words

The main intention of designing the raspberry pi board is to increase the encouragement on learning, experimentation and innovation for students. The raspberry pi board is portable and low cost. Maximum of the raspberry pi computers is used in mobile phones [8].

III. CATEGORIZING THE COMPONENTS

In this section we categorize the necessary requirement for the process as hardware and software based upon the proper usage of those parts.

A) HARDWARE COMPONENTS

The components needed for NLP implementation can be summarized in the following way:

1) Raspberry Pi

Unlike CPU, the Graphics Processing Unit on the Pi is equivalent to that in a high specification mobile device. It can run 3D games and play high-definition video. With the right software, a TV and a broadband link you can have i-Player, YouTube and other videos services at your fingertips. Python is intended as an integral part of the 'standard' teaching toolkit.

An Outlook model of Raspberry Pi is shown in Fig. 1.

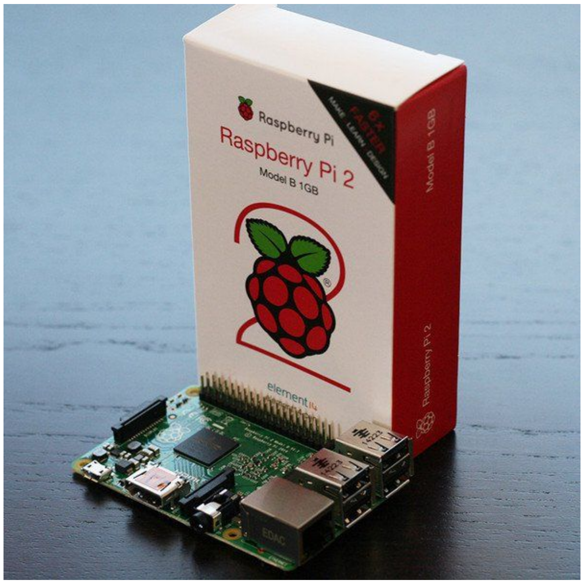

Fig. 1. Raspberry Pi 2

The Pi comes with 512MB of RAM. Programs are stored on the SD card and the Pi is powered on. They are copied into the much faster RAM until the computer is turned off and the RAM is cleared. One of the most convenient aspects of Raspberry Pi is that you can convert it from a media player to a desktop computer just by swapping out the SD card. This is easier than removing a laptop's hard disk. A single chip contains the pi's memory, central processing unit, and graphics chip. The version used in the pi is slower than the ones in i-pad and others but it is fast enough to do the job.

The architecture of Raspberry Pi is shown in Fig. 2.

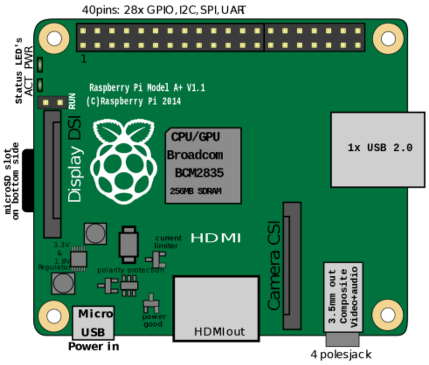

Fig. 2. Architecture of Raspberry Pi

2) MICROPHONE

In general, a **microphone** is any device capable of recording a voice.It is used as an input device for inputting the voice.Usually the microphone is installed in a CD drive, but in the case of raspberry pi it is downloaded as a driver as it is required. Later the Microphone is given a source code or a name for instance to be called during the process.

SPEECH RECOGNITION FROM MICROPHONE:

```
Import speech_recognition as sr
#obtain audio from microphone
r=sr.Recognizer()
withsr.Microphone() as source:
printf("say something!")
audio=r.listen(source)
```

3) SPEAKER

Speaker is used as an output device for sending out the converted text to speech response.

B) SOFTWARE COMPONENTS

1) LINUX

Linux is an open source operating system for computers, mainframes, servers, mobile devices and embedded devices. The Linux OS includes the Linux kernel as well as supporting tools and libraries. Popular Linux OS distributions include Debian, Ubuntu, Fedora, Red Hat, etc., here we are using Debain and the reason is specified.

2) PYTHON

One of the advantages of Python is that it allows us to type directly into the interactive interpreter. We can access the Python interpreter using a graphical interface called the Interactive Development Environment (IDLE). Python very closely resembles the English language. In this paper the functions are called using python.

3) DEBIAN

An operating system is the set of basic programs and utilities that make our computer run. At the core of an operating system is the kernel. The kernel is the most fundamental program on the computer and lets you start other programs.

Debian systems use the Linux kernel which is a piece of software. FreeBSD is an operating system including a kernel and other software in it.

However, the work is in progress to provide Debian for other kernels. The Hurd is a collection of servers to implement different features that run on top of a microkernel. Like a tower-at the base is the kernel, on top of it are all the basic tools. Next is the software that runs on the computer. At the top of the tower is Debian.

4) POCKETSPHINX

Pocketsphinx is a library that depends on another library called SphinxBase. It is a lightweight speech recognition engine. To install Pocketsphinx, you need to install both Pocketsphinx and Sphinxbase. Pocketsphinx can be used in Linux, Windows, MacOS, iPhone and Android. In my paper I am using this pocketsphinx as a speech to text conversion engine. It is converted as an image file and extracted for execution.

5) IBM

The IBM Speech to Text services provides an API that enables you to add IBM's speech recognition capabilities to your applications. The service transcribes speech from various languages and audio formats to text with low latency. This service can also be used instead of pocketsphinx as this provides both broadband and narrowband.

RECOGNIZING SPEECH USING SPHINX:

```
Try:
Printf(“sphinx thinks you said”+r.recognize_sphinx(audio))
exceptsr.UnknownValueError:
printf(“sphinx could not understand audio”)
exceptsr.RequestErroe as e:
printf(“sphinx error; {0}”.format(e))
```

PROCESSING TECHNIQUE:

The whole conversion process is classified into two main sections as follows

1) Speech to text recognition
2) Text to speech conversion

Speech to text recognition

- Before the process begins we must install the speech recognition module, which is the Pocketsphinx as of here.Installation ofpocketsphinx is easy and it requires installation of three components altogether. They are thesphinxbase,pocketsphinx,and pocketsphinx-python.
- SphinxBase is the base package that all of the other Sphinx programs use
- PocketSphinx is the lightweight recognizer to decode phrases faster
- PocketSphinx-python is the wrapper to allow us to program in the best scripting language ever.

Speech recognition can be achieved in many ways on Linux (and so on the Raspberry Pi).

- Speech Recognition Toolkit
- Installing build tools and required libraries
- Building Sphinxbase
- Building PocketSphinx
- Creating a Language Model

The user sends in the input speech to the microphone. The voice is detected and the code sets up the microphone and saves each phrase detected as a temporary file. This file is decoded by the sphinx decoder and is translated into a list of strings in the predefined state. The sphinx base is used as a basic layer for the conversion of the speech text.

A language module is created in the beginning which contains all the predefined sentences. The text is matched with the module and verified. If the texts match a positive response is picked from the database. If the inputted text doesn’t match with the database module the response is searched via online speech recognition modules and the matched database is sent for further processing. Below is the systematic representation of the input-output module:

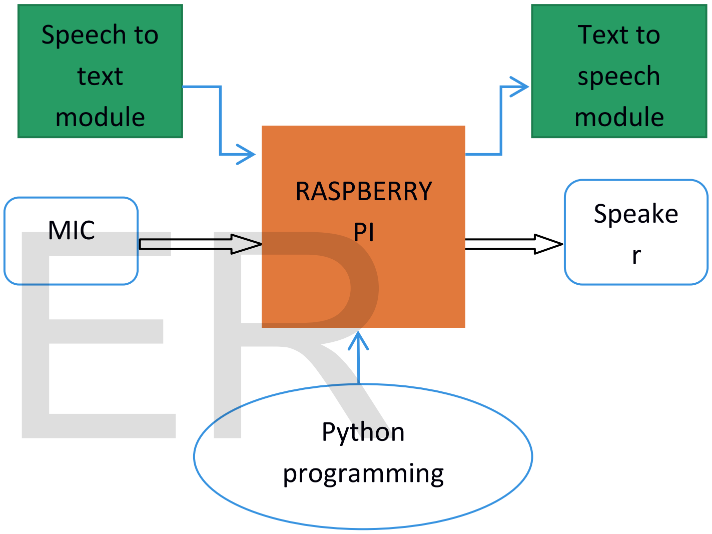

Fig. 3. Python Programming Block Diagram

Text to speech recognition

The converted and processed text is now again converted to speech. To convert it into speech a module called festival is used.

Festival is a **free text to speech** tool. When we pass a text file to festival, it converts the contents of the text file into voice.

Installation of festival is also very simple.

- sudo apt-get install festival

This is used to install festival.

- Try out Festival with:

```
echo “Just what do you think you're doing, Dave?” | festival --tts
```

There are many advantages of using festival rather than other text converters:

- It is Available for Free under an open source license.
- The quality of the voice and the pronunciations are really good.
- It Supports 3 languages - English,Welsh and Spanish

CONCLUSION AND FUTURE ENHNACEMENT

Natural Language Systems are very complicated to design. NLP's future will be redefined as it faces new technological challenges to create more user-friendly systems. It is also forcing NLP more towards Open Source Development. If the NLP community embraces Open Source Development, it will make NLP systems less proprietary and therefore less expensive [9]. Few of the applications of NLP are

- Personal assistant
- Assistance for autistic children- Autism is a life-long developmental disability that affects how people perceive the world and interact with others. Autism is a spectrum condition. Autistic people hear and feel the world differently than the others. This particular module on NLP focuses on providing assistance especially for these autistic children. The future enhancement of this project will be a major hands-on project on complete revival from the disabilities, thus providing them a chance to live a life of their own choice.
- Chat bots- Chatter bots use natural language processing to simulate conversations with users. Web sites are beginning to install chatter bots as Web guides and customer service agents
- Assistance in directing people at public related places.

REFERENCES

[1] Natural Language Processing Gobinda G. ChowdhuryDept. of Computer and Information Sciences University of Strathclyde, Glasgow G1 1XH, UK

[2] Ranjit Bose "Natural Language Processing: Current state and future directions", International Journal of the Computer, the Internet and Management Vol. 12#1, January – April, 2004.

[3] Natural Language Processing Techniques Applied in Information Retrieval-Analysis and Implementation in Python, TulikaNarang, International Journal of Innovations & Advancement in Computer Science IJIACS ISSN 2347 – 8616 Volume 5, Issue 4 April 2016

[4] Bharati A., Chaitanya V and Sangal R, "Natural Language Processing: A Paninian Perspective", Prentice Hall of India, 1995.

[5] Liddy, E. D. In Encyclopedia of Library and Information Science, 2nd Ed. Marcel Decker, Inc.

[6] J. Allan. Perspectives on information retrieval and speech. In Information Retrieval Techniques for Speech Applications: LNCS 2273, pages 1–10, 2002

[7] U.S. Tiwary, TanveerSiddiqui "Natural Language Processing and Information Retrieval" Oxford Higher Education 2008.

[8] Raspberry Pi Technology Ms. Sejal V. Gawande, Dr. Prashant R. Deshmukh, Volume 5, Issue 4, April 2015 ISSN: 2277 128X International Journal of Advanced Research in Computer Science and Software Engineering Research Paper

[9] AVinitha et al, Int.J.Computer Technology & Applications, Vol 4 (5), 751-

758,"Natural Language Processing" Preeti and BrahmaleenKaurSidhu

[10]http://en.wikipedia.org/wiki/Natural_language_processing

[11]http://www.doc.ic.ac.uk/~nd/surprise_97/journal/vol1/hks/int ro.html

[12]http://www.doc.ic.ac.uk/~nd/surprise_97/journal/vol1/hks/te chiq.html