A seminar Report on

Artificial intelligence in Speech Recognition

*BACHELOR OF TECHNOLOGY*

*IN*

*INFORMATION TECHNOLOGY*

By

Harshitha Aditham (01)

Nadar Rajeshwari (26)



**Usha Mittal Institute of Technology SNDT Women's University**

**JuhuTara Road Sir Vitthaldas Vidyavihar Santacruz(W) Mumbai**

**400049**

**CONTENTS:-**

1. Introduction
2. What is Speech Recognition?
3. Speech Recognition Process
4. AI in Speech Recognition
5. Speech Recognition Softwares
6. Algorithm
7. Program on Speech Recognition
8. Advantages & Disadvantages
9. Conclusion

**WHAT IS SPEECH RECOGNITION ?**

**Speech recognition** is the inter-disciplinary sub-field of computational linguistics that develops methodologies and technologies that enables the recognition and [translation](https://en.wikipedia.org/wiki/Translation) of spoken language into text by computers. It is also known as automaticspeechrecognition (**ASR**), computerspeechrecognition or speechtotext (**STT**). It incorporates knowledge and research in the linguistics, computer science and electrical engineering fields.

The speech recognition can be either speaker dependent or speaker independent. Speech recognition is the translation of spoken words into text. Speech Recognition systems made more than 10 years ago also faced a choice between discrete and continuous The speech recognition is the process by which a computer or a system identifies the spoken words.

speech. It is much easier for the program to understand words when we speak them separately, with a distinct pause between each one. Most modern systems are capable of understanding continuous speech.

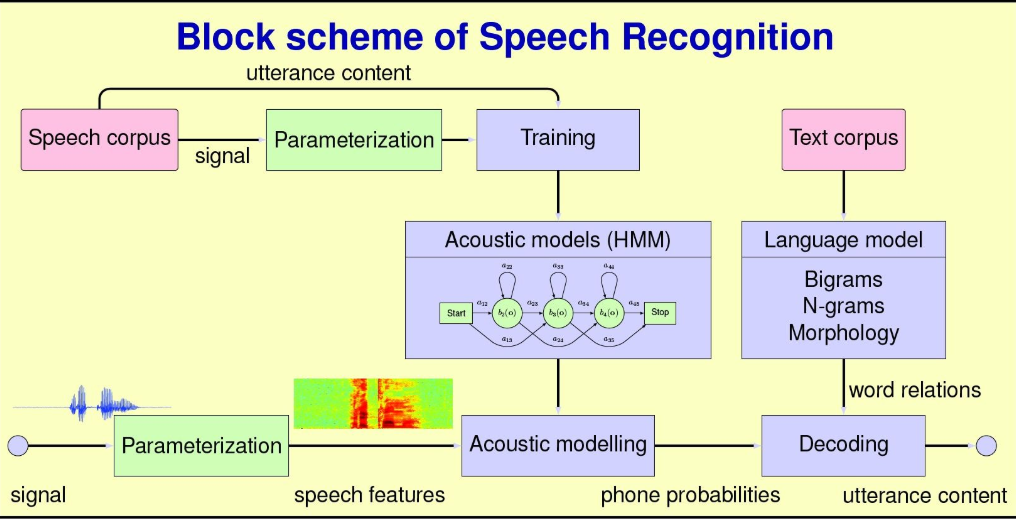
**Standard CONVERSION OF SPEECH TO TEXT**

The first component of speech recognition is, of course, speech. Speech must be converted from physical sound to an electrical signal with a microphone, and then to digital data with an analog-to-digital converter. Once digitized, several models can be used to transcribe the audio to text.

Most modern speech recognition systems rely on what is known as a [Hidden Markov Model](https://en.wikipedia.org/wiki/Hidden_Markov_model)(HMM). This approach works on the assumption that a speech signal, when viewed on a short enough timescale (say, ten milliseconds), can be reasonably approximated as a stationary process—that is, a process in which statistical properties do not change over time.

In a typical HMM, the speech signal is divided into 10-millisecond fragments. The power spectrum of each fragment, which is essentially a plot of the signal’s power as a function of frequency, is mapped to a vector of real numbers known as [cepstral](https://en.wikipedia.org/wiki/Cepstrum) coefficients. The dimension of this vector is usually small—sometimes as low as 10, although more accurate systems may have dimension 32 or more. The final output of the HMM is a sequence of these vectors.

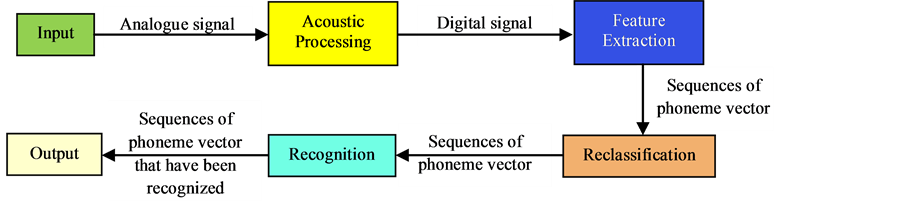
To decode the speech into text, groups of vectors are matched to one or more [phonemes](https://en.wikipedia.org/wiki/Phoneme)—a fundamental unit of speech. This calculation requires training, since the sound of a phoneme varies from speaker to speaker, and even varies from one utterance to another by the same speaker. A special algorithm is then applied to determine the most likely word (or words) that produce the given sequence of phonemes.



**SPEECH RECOGNITION PROCESS**:

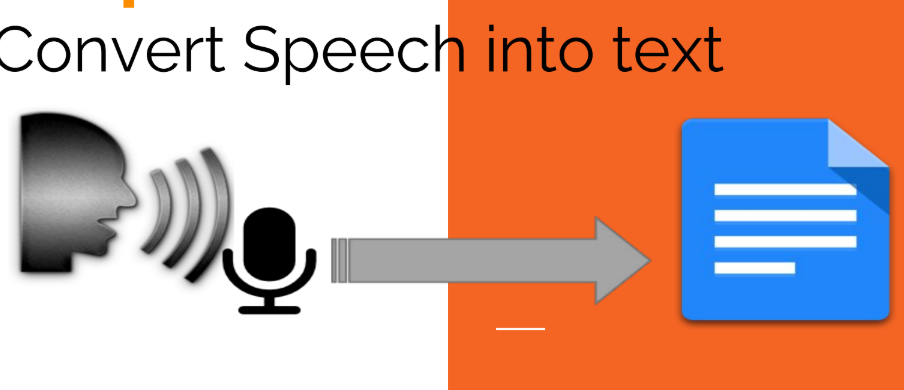
In humans the speech or acoustic signals are received by the ears & then transmitted to the for understanding & extracting the meaning out of the speech & then to react it appropriately. speech recognition enabled computer or devices too, work under the same principle.They receive the acoustic signal through microphone; these signals in analog form & need to be digitalized to be understood

By the system. These signals are then digitalized & sent to the processing unit for extracting the meaning out of the signals & to give the desired output to the users.



Any Speech Recognition system involves following major 5 steps:

1. Signal processing
2. Speech recognition
3. Semantic interpretation
4. Dialog management
5. Response generation

**STEP1:-** 

**STEP 2:-**



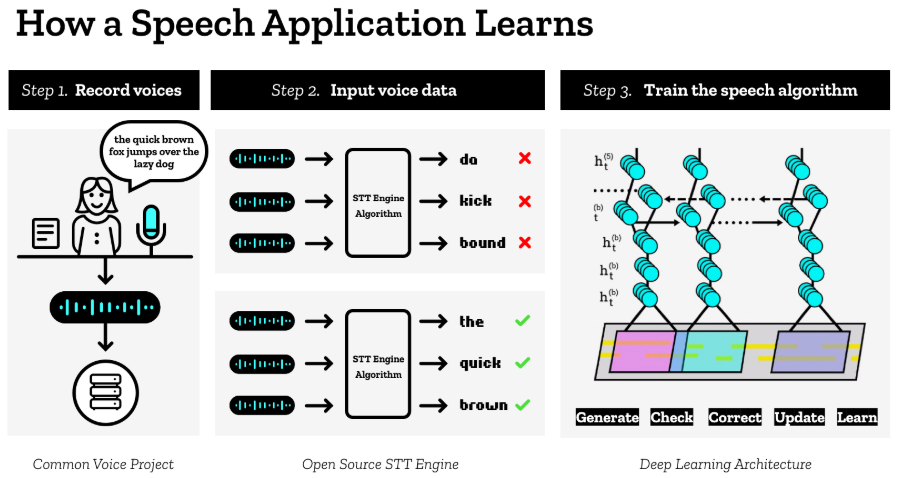
A speech recognizer consists of a number of components. These are learned from data, using a Speech Corpus consisting of recordings of speech and their textual transcriptions. The Speech Recognizer learns to make correspondences between sounds and words.

**Signal Processing**

This processes the signals recorded by the microphone into FeatureVectors that provide a snapshot of what is going on in the speech signal, emphasising those features that are relevant to speech recognition. Typically, 100 feature vectors per second are produced.

**Acoustic Model**

This takes the stream of Feature Vectors and turns it into a stream of Phonemes (or Phoneme Hypotheses). A Phoneme is the unit that is used to construct words and corresponds to a particular speech sound. An important aspect of the Acoustic Model is that it does not make definite decisions about what the stream of Phonemes is but tells us how likely any particular Phoneme is at a point in the speech signal.



**SPEECH** **RECOGNITION** **INSTALLATION**:

SpeechRecognition is compatible with Python 2.6, 2.7 and 3.3+.

You can install SpeechRecognition from a terminal with pip:

$ pip install SpeechRecognition

Once installed, you should verify the installation by opening an interpreter session and typing:

>>> import speech\_recognition as sr

>>> sr.\_\_version\_\_

'3.8.1'

Speech Recognition will work out of the box if all you need to do is work with existing audio files. Specific use cases, however, require a few dependencies. Notably, the PyAudio package is needed for capturing microphone input.

## **The Recognizer Class**

All of the magic in SpeechRecognition happens with the Recognizer class.

The primary purpose of a Recognizer instance is, of course, to recognize speech. Each instance comes with a variety of settings and functionality for recognizing speech from an audio source.

>>> r = sr.Recognizer()

Speech Recognition makes working with audio files easy thanks to its handy Audio File class. This class can be initialized with the path to an audio file and provides a context manager interface for reading and working with the file’s contents.

**Pyaudio**

Windows

On Windows, you can install PyAudio with pip:

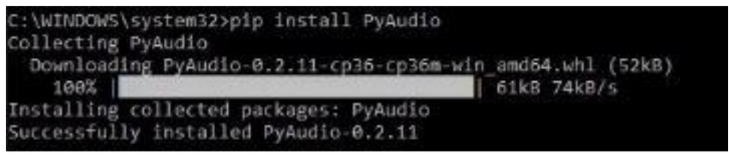
$ pip install pyaudio

Testing the Installation

Once you’ve got PyAudio installed, you can test the installation from the console.

$ python -m speech\_recognition

Make sure your default microphone is on and unmuted. If the installation worked, you should see something like this:



**MICROPHONE:**

Microphone serves as a voice input device. It captures the voice data and input to the computer.

Using microphone along with speech recognition software can offer a completely new approach to input information into your computer

Speech recognition programs, although not yet completely exact have made great strides in accuracy as well as ease of use.

The voice-in or speech recognition approach can almost fully replace the keyboard and mouse.

**Pyglet:**

Piglet is a cross-platform windowing and multimedia library for Python intended for developing games and other visually rich applications.

It supports windowing, user interface event handling. OpenGL graphics, loading images and videos and playing sounds and music**.**



**ALGORITHM:-**

**Speech:**

Start

* Send message through microphone
* Machine converts speech to text
* Text is ready by speak method
* Text is send to chrome path
* Output is displayed in browser

end

**speak**

start

* Text converted into speech
* “Temp” file containing voice is used

end

**Program:-**

import speech\_recognition as sr

import webbrowser as wb

chrome\_path='C:/Program Files (x86)/Google/Chrome/Application/chrome.exe %s'

r=sr.Recognizer()

with sr.Microphone() as source:

print('Say Something!')

audio =r.listen(source)

print ('Done!')

try:

text =r.recognize\_google(audio)

print('Google thinks you said:\n' + text)

wb.get(chrome\_path).open(text)

except Exception as e:

print(e)

**speak:**

from gtts import gTTS

import pyglet

import time.os

def tts(text,lang):

file=gTTS(text=text,lang=lang)

filename=’temp.mp3’

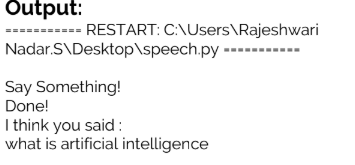
file.save(filename)

music =piglet.media.load(filename,streaming=False)

music.play()

time.sleep(music.duration)

os.remove(filename)



**ADVANTAGES:**

1.people with disabilities can be benefited.

2. organizations- increases productivity, reduces costs and errors.

3. lower operational costs.

4. advances in technology will allow consumers and businesses to implement speech recognition systems at a relatively low cost.

-> cell phone users can dial pre-programmed numbers by voice command.

-> users can trade stocks through a voice-activated trading system .

-> speech recognition technology can also replace touch-tone dialing resulting in the ability to target customers that speek different languages

5. easy to do verification test,

6.Most people find it easy to use and fast entry of texts into documents .

**DISADVANTAGES:-**

1. People with limited arm/wrist may not find it easy to use.
2. Data entry may be slow.
3. Conversations

->involves more than just words (non-verbal communication ;stutters etc.)

* Every human being has differences such as their voice,mouth and speaking style.
* Filtering background noise is a task that can even be difficult for humans to accomplish.

**CONCLUSION:**

Speech recognition has a big potential in becoming an important factor of interaction between human and machine in the near future.