Review paper on speech recognition and machine learning

Priyanka Wani
Department of Electronics & Telecommunication, D. Y. Patil Institute of Technology ,Pune, India

Mukesh Ghogare
Department of Mechatronics Engineering, Marathwada Mitramandal's Institute of Technology, Pune

Rashmi Deshpande
Department of Instrumentation Engineering, D. Y. Patil Institute of Technology, Pune

Abstract— Voice is the most expressed form of human communication. Talking is simple, hands-free, and quick, and requires no technological expertise. The practice of automatically collecting and interpreting linguistic information from speech waves using electronic circuits or computers is known as speech recognition. Phonetic information is another name for descriptive information, which is the necessary data in a audio. The main goal of years of research into machine-learning speech recognition techniques was to create voice-recognizing robots. The technique by which a computer understands what someone has spoken is known as speech recognition, sometimes referred to as automatic speech recognition. The likelihood is that you are acquainted with speech recognition technology from phone-centric applications. When a computer prompted you to enter the first name of the individual you wanted to speak with over the phone while you were calling a business, the computer used voice recognition to identify the name you spoke.

Keywords-ASR, recognision

I. Introduction

Text is created by automatically converting a spoken word sequence into an audio file. Speaking to a computer via speech is a more straightforward and pleasant method of communication for humans than using a keyboard and mouse, which need more dexterity and hand-eye coordination [2]. People with physical disabilities or those who are blind find using computers challenging. All of these problems are resolved by speech recognition.

Speech recognition differs from voice recognition in that it does not attempt to identify specific individuals, even though it both employs some of the same basic technologies. Instead, it makes an effort to understand what people say. It makes a distinction between who speaks and what is saying. Speech engines generally follow a similar procedure for identifying what a speaker said, despite their wide variations [1]. The list of words that need to be recognized is loaded by the engine. A grammar is a list of letters like this one.

The speaker's audio is loaded by the engine. A waveform, which is simply a mathematical representation of sound, is used to represent this audio. Fig.1 shows Mathematical Representation of Speech model.

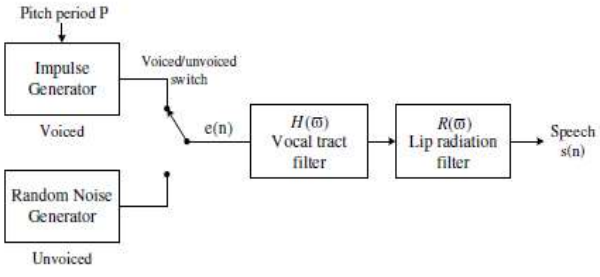

Fig.1: Mathematical Representation of Speech model

For producing speech, which is categorized into 3 independent sections: the vocal tract filter component, the radiation part, and the excitation production part.[9]The excitation

portion is related to either the glottis's vibration, which produces voiced sounds, or the vocal tract's constriction, which creates turbulent airflow and noise-like unvoiced excitation. This model allows one to calculate a voiced speech, like a vowel, as the product of three different (Fourier) transfer functions:

$$S(\varpi) = E(\varpi)H(\varpi)R(\varpi)$$

The distinctive part to determine articulation is the vocal tract transfer function H (w), where the excitation spectrum E (w) and radiation R (w) are mainly constant and well understood a priori [10]. As a result, it merits our specific consideration for how to model it effectively. Speech events categorized in different parts. By far the easiest and most obvious is the vocal chords, which are the source of voice production. The three states that are represented by the commonly used 3 levels representations: 1) Silence (S), in which no sound is produced; 2) Unvoiced (U), where the vocal chords are not vibrating and the resultant expression waveform is periodic or random; and 3) Voiced (V), where the vocal chords are tensed and vibrate frequently when air passes from the lungs, producing a quasi-periodic speech waveform. [9]

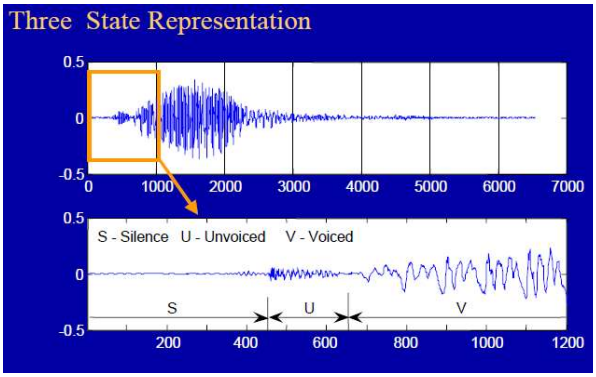

Fig.2: Three state representation of Speech Signal

The linear predictive coding (LPC) spectral evaluation framework and a filter bank spectrum analysis model are the two main approaches to spectral analysis. a method known as Vector Quantization It further lowers the information rate of signal processing by encoding continuous spectral representation by a "typical" spectral form in a finite codebook of spectral shapes.

II. History

The initial device to identify speech to any considerable extent was the 1920s-era Radio Rex toy, which is credited with starting the 20th century's speech recognition history. The incapacity of this toy is to reject noises that are not in its lexicon. Since it served a practical goal that many of the modern laboratory equipment that followed could not typically do, the toy may be considered beneficial in some ways. Despite being quite basic, it represented a key idea in speech recognition for many years: save a representation of the distinctive qualities of the intended sound and use an algorithm to match this feature to spoken input. The first real word recognizer was a system developed at Bell Laboratories in 1952 by Davis K., Biddulph R., and Balashek S. This system could be trained to identify digits from a single speaker.

The data was assembled by David and Selfridge in 1962, and it included a comparison of several speech-recognition studies conducted during the previous ten years, including both of the recognizers previously stated. Generally speaking, spectral tracking was done, along with sound and word detection and small-scale testing on participants.

Similar advances were made in automatic speech recognition research throughout the 1960s. In 1964, Martin used neural networks to recognize phonemes, and he was able to achieve high accuracy for numerous speakers.

Significant advances in the 1960s contributed to the development of voice recognition and were eventually of important significance for recognition.Furthermore, two novel techniques for sequence pattern matching were created: the statistical Hidden Markov Model (HMM) and the deterministic Dynamic Time Warping (DTW).

III. Speech Parameters

Whether an Automatic Speech Processing (ASP) system is speaker independent (SI) or speaker dependent (SD) is one of its qualifying factors. An SD system is one that has undergone testing and training on the same speaker. A discontinuous collection of speakers is used to test and train a SI system. For better accuracy, large vocabulary systems designed for personal computer use have typically been SD. Even though a lot of systems have been purportedly SI, a lot of them will perform very badly when used by speaker who is not native to the language. Whether the task involves identifying solitary speech, continuous speech, or keywords is another description. The first kind of job involves identifying words alone, and it is typically easier than identifying continuous speech, where word boundaries are less obvious. Keyword spotting is a third task type that sits in between the first two.

There is an additional parameter introduced by the size of the lexicon. The degree of speech recognition difficulty is significantly influenced by speaking style. An Automatic Speech Recognition (ASR) task's difficulty is also influenced by the recording settings.

The quantity of zero crossings in a predetermined frame length is known as the zero crossing rate. The total square value of the spoken word waveform for each frame makes up the spectral energy. (Noise-like sounds have more energy concentrated in frequencies; nasal and vowel-like sounds have less energy concentrated in frequencies).

The fundamental terms that are necessary to comprehend speech recognition technology are listed below.

Utterance: A computer-interpreted utterance is the vocalization, or speaking, of one or more words of same meaning. One word, a couple of words, a paragraph, or even several sentences might be considered an assertion.

Speaker Dependency: Systems that rely on a particular speaker are built around it. For the correct speaker, they are typically more accurate, but for other speakers, they are significantly less accurate. They anticipate that the voice and speed of the speaker will remain constant. Systems that are speaker independent are made to work with a range of speakers. Typically, adaptive systems begin as speaker-independent systems and then use training methods to adjust to the speaker in order to improve the accuracy of speaker recognition.

Dictionaries: It is often known as vocabularies, are collections of terms or expressions that the SR system is capable of recognizing. Larger vocabularies are more challenging for computers to recognize, but smaller vocabulary is generally easier. Not every entry in this dictionary has to be a single word, unlike regular dictionaries. They may consist of one or more sentences. Very large vocabulary can have a hundred thousand or more recognized utterances, whereas smaller vocabularies can contain as few as one or two (e.g., "Wake Up").

Accuracy: The degree, to which a recognizer can identify utterances accurately, or how well, is a measure of its ability. This entails determining if a spoken utterance is outside of its lexicon in addition to accurately identifying the utterance.

Training: Some speech recognition software can adjust to a speaker's voice. When the system had this capability, training might be possible. When a speaker repeats a common or standard phrase, an ASR system learns from this and modifies the corresponding algorithms to fit that specific voice. Accuracy of a recognizer is typically increased with training. Speakers who have trouble speaking or pronunciation certain words can also benefit from training. ASR systems with sufficient instruction should be able to adjust as long provided the speaker can reliably repeat an utterance.

IV. Technique Used

An overview of a Word Recognition (WR) system with three main steps: pre-processing, extracting MFCC feature vectors, and matching these vectors against a database using a pattern recognition algorithm. This is a common approach in speech processing systems. Let's break down each step:

- The block diagram (Fig. 3) likely illustrates the flow of information between the pre-processing, feature extraction, and pattern recognition components.[9]

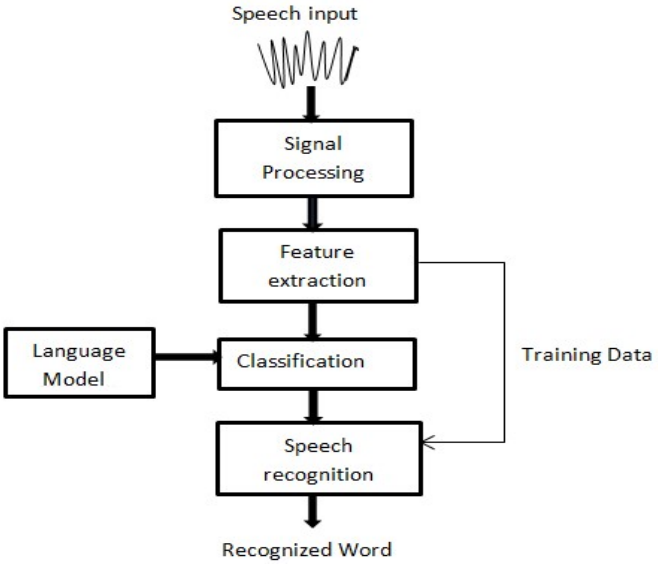

Fig.3: Proposed Model for SR

Input Signal: Recording audio/utterance can be done in several ways. A good place to start is to compare the recently recorded sample with the ambient audio levels (or, in some situations, the acoustic energy). Because speakers frequently leave behind "artifacts" like breathing, sighing, teeth chattering, and echoes, end point detection becomes more difficult.

Signal Processing (End Point Detection): There are several methods for achieving pre-filtering/pre-emphasis contingent on additional characteristics of the identification system. An essential component of speech signal processing processes, such as speech recognition that is automatic, is endpoint detection, which distinguishes speech from non-speech segments in digital speech signals. A strong endpoint detector can increase a speech recognition system's precision and speed. These approaches can be roughly divided into two kinds. One is threshold-based. In order to categorize each sample, this type of approach typically extracts the acoustic characteristics for each sample first, and then compares the feature values with predetermined thresholds. The alternative approach is pattern-matching,

which requires estimating the speech and noise signal model parameters. A recognition process and a detection method are comparable. When compared to the pattern matching method, the thresholds-based approach is easier to use, quicker, and requires less training data to train models.

Feature Extraction:

- MFCC stands for Mel Frequency Cepstral Coefficients, which are widely used in speech and audio processing.
- These coefficients capture essential features of the speech signal, especially focusing on the frequency components that are most relevant to human hearing.

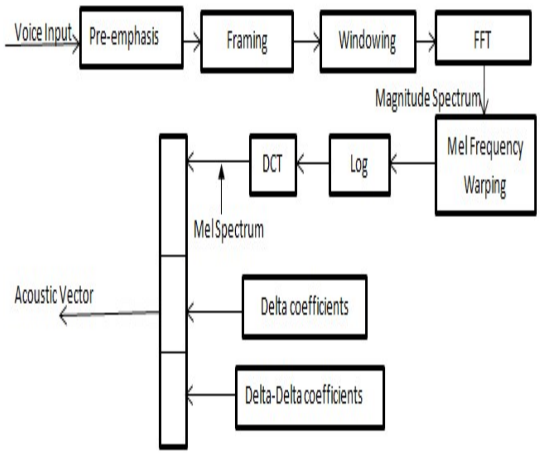

Fig.4: Model of MFCC

Procedure to obtain coefficient are-

1. Sampling:

- The voice signal is considered quasi-stationary, so it is divided into short frames (typically 20-30 ms) where the characteristics are assumed to be stationary.
- This simplifies the analysis, allowing for the consideration of the signal over short durations.

2. Windowing:

- Each frame is multiplied by a window function to avoid aliasing effects and minimize discontinuities.
- Common window options include Rectangular, Hanning, and Hamming. The Hamming window is often preferred for speech recognition as it is compatible with the Mel scale and exhibits good performance.
- Every frame has the Hamming window function applied to it in order to maintain continuity at the endpoints and avoid sudden changes. Hamming window has highest side lobe attenuation and larger transition width of 8π/M where M is filter order. Hamming window is used to avoid Gibbs phenomenon.[9]

The Hamming window is defined as-

$$W[n] = \begin{cases} 0.54 \quad 0.46\cos(2\pi n/N), & 0 \le n \le N-1 \\ 0 & \text{, otherwise} \end{cases}$$

For the purpose to maintain continuity between the beginning and ending points of each frame and avoid sudden changes at the end, each framework is multiplied by the hamming window [10].

3. Fourier Transformation:

- After windowing, a Discrete Fourier Transform (DFT) is used to each sample to convert the signal domain.

$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{\frac{-i2\pi nk}{N}}, k = 0 \ldots N-1$$

Since the FFT algorithm is the fastest for calculating DFT, it is occasionally used to faster the counting rate of DFT by a factor of 100 [9]. The vocal tract parameters are represented by a slowly varying envelope, while the fundamental frequency is represented by rapid variations in the FFT log.

4. Logarithmic Spectrum:

- The log of the spectrum is taken to represent the amplitude of spectral lines in decibels (dB).
- This representation captures faster changes corresponding to the normal frequency and slowly changing envelope corresponding to vocal tract parameters.

5. Frequency Warping:

- Based on how frequencies are perceived by human hearing, the spectrum is scaled using the Mel method. - The Mel scale is approximately linear below 1000 Hz and logarithmic above. It is defined by setting 1000 mels equal to 1000 Hz as a reference point.

Formula to calculate Mel Scale is-

$$Mel(f) = 2595 * \ln(1 + f/700)$$

Here Mel (f) denotes perceived frequency and f is original frequency [9].

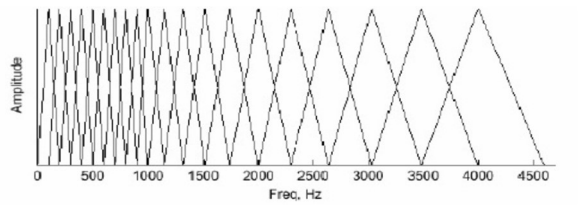

Fig.5: Filter bank of Mel Scale

- Mel frequency warping is used to represent the spectrum in a way that is more consistent with human perception.

The overall goal of these steps is to transform the raw audio signal into a representation that is more suitable for capturing the distinctive features of speech, which can then be used for further processing, such as word recognition. The use of MFCCs is a common technique in speech signal processing due to their effectiveness in capturing relevant information for speech analysis.

Classification:

- Classification is the final step in the word recognition process, where the extracted features (such as MFCC coefficients and their derivatives) are used for pattern matching.

Vector Quantization:

Vector mapping from a huge vector space to a finite number of regions within that space is known as vector quantization (VQ). A codeword is a central term for each region, which is referred to as a cluster. A codebook is an alphabetical list of all code words.

Data will be appropriately displayed and compressed in this scenario. In VQ, a centroid is identified for every cluster following data segregation. VQ was first created to lower transmission bandwidth for speech communication systems. Rather of all the bits required to represent the entire vector, only a representation of the cluster was transmitted. [6] When it comes to speaker recognition, VQ creates vector spaces with the typical vectors of the speaker following feature extraction. A few typical vectors for each speaker are obtained through the use of VQ: the codebook. An audio signal of speaker is "vector-quantized" utilizing each codebook throughout the recognition process, and the distance between a sample and the nearest codeword (each vector of the codebook) of a codebook is calculated. We refer to this distance as distortion. The speaker with the least amount of distortion is chosen for SI applications. It is necessary to use a threshold in SV applications. Frequently, text-independent applications employ VQ. Typically, prior temporal alignment is necessary for its utilization in systems that rely on text. Fig. 5 shows the process of recognition. An acoustic space's two dimensions and just two speakers are depicted in the figure. The triangles are from speaker 2, and the circles represent the acoustic vectors from speaker 1.

During the training phase, each known speaker's training acoustic vectors are clustered to create a speaker-specific VQ codebook. In Fig.6, the result code words (centroids) for speakers 1 and 2 are represented by black circles and black triangles, respectively. A VQ-distortion is

the length of time that separates a vector from a codebook's closest codeword. During the recognition stage, each taught codebook is used to "vector-quantize" each input word of an anonymous speaker, and the total VQ distortion is calculated. The speaker with the least amount of overall distortion, according to the VQ codebook, is found.

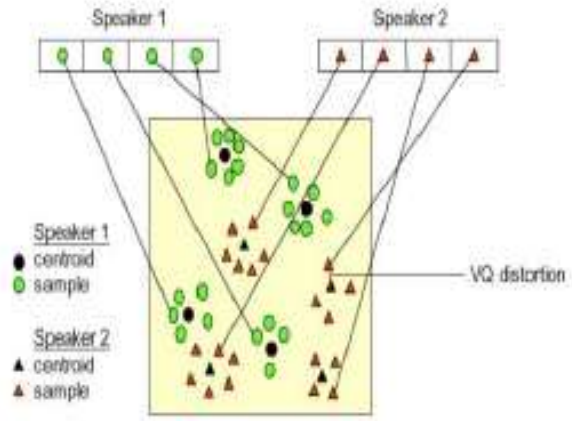

Fig.6:VQ Algorithm

The normal distance between 2 targets that one might calculate is known as the Euclidean distance or Euclidean metric in mathematics, and it may be demonstrated by repeatedly using the Pythagorean Theorem. Although it would be impractical to keep every feature that the MFCC algorithm generates from the voice utterance, vector quantization has been utilized. An important procedure called the LBG VQ algorithm is applied throughout the VQ codebook's training phase. Its purpose is to cluster a set of L train vector into a set of M codebook vectors. This recursive process is the formal implementation of this algorithm. The learning process of the VQ codebook utilizing the LBG algorithm, as outlined by Juang and Rabiner (1993), requires a series of stages.

1. Create a 1-vector codebook, which serves as the training vector set's centroid. Consequently, there is no need for iteration in this stage.

2. Divide each existing codebook yn in half using the following formula to double the codebook's size:

$$y_n^+ = y_n(1+\varepsilon)$$
$$y_n^- = y_n(1-\varepsilon)$$

Here n is the separating parameter and n ranges from 1 to the codebook's current size. The first codebook is typically created by combining all of the features that were chosen vectors for each individual word in a single database. This first codebook's objective is to act as a foundational codebook to test each chosen feature vector against the others. The LBG VQ Matlab function refers to the original codebook as the variable "CODE."

3. Nearest-Neighbor Search: Assign each training vector to the relevant cell (connected to the nearest centroid) by locating the keyword in the present codebook that is the closest (based on similarity measurement). The K-means iterative technique is used for this.

4. Centroid Update: Using the center of the training vectors allocated to each cell, modify the centroid in each cell. By calculating the latest value of the code vector by averaging the speech vector within a cell, the centroid adjustments necessitate updating the codebook as well. A flow diagram illustrating the LBG algorithm's specific steps is shown in Fig. 7[7]. The nearest-neighbor search method known as "cluster vectors" allocates every training scalar to a cluster that is connected to the closest codeword. The centroid update process is called "Find centroids." To find out if the process has converged, "Compute D (distortion)" adds up the distances of each training vector in the nearest-neighbor search.

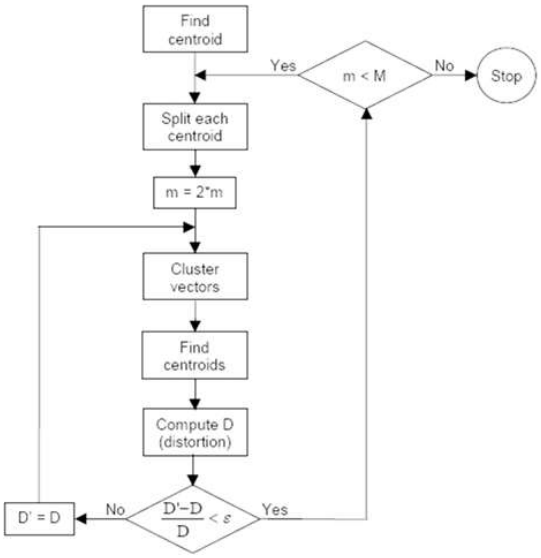

Fig.7: Flow chart of LBG Algorithm

After transforming what is said into its codebook, the Euclidean distance metric is used to determine how similar or unlike two uttered words are from one another [8]. The process of matching an unknown word to its feature vector in the database involves calculating the Euclidean distance that exists among the feature vector and the codebook, or model, of the known words. To determine the unknown word, the aim is to locate the codebook with the smallest distance measurement. For instance, during the evaluation or identification process, the Euclidean distance that lies between each uttered word's characteristics vector and codebook is determined; the word chosen has the shortest average minimum distance, as indicated by the equation follows.

$$d(x, y) = \sqrt{\sum_{i=1}^{D} (x_i - y_i)^2}$$

If yi is the ith characteristics vectors in the codebook, d is the distance of xi and yi, and xi is the ith input features vector. An unknown feature vector is subjected to a basic the Euclidean distance measure and compared to the trained codebook. Therefore, in order to calculate their distance and verify the complete performance, this method requires two inputs: an unidentified spoken word and a trained codebook. The ID numbers issued to each feature vector in the training codebook, along with the square of the error values and distances, are the algorithm's outputs. But this approach selects the feature vector's ID number that is closest to the unidentified feature vector in terms of distance.

V. Conclusion and Future Work

Speech is the primary means of human-to-human communication and information acquisition. We have covered the techniques established at each stage of the speech recognition system in this overview. Here, we matched patterns using VQ and extracted features using MFCC. Both illiterate and disabled people will benefit from this deployment, which will provide them with a crucial channel for communication and education through sophisticated electronics like computers, smartphones, and household gadgets. The same algorithms can be used to achieve speech recognition on a DSP processor. It would be highly beneficial for real-time applications as a result. There are many opportunities for developing a system on an ASIC/FPGA in the future.

REFERENCES

[1] Becchetti, Claudio, and Klucio Prina Ricotti. *Speech recognition: Theory and C++ implementation (with CD)*. John Wiley & Sons, 2008.

[2] Duda, Richard O., and Peter E. Hart. *Pattern classification*. John Wiley & Sons, 2006.

[3] Rabiner, Lawrence R. *Digital processing of speech signals*. Pearson Education India, 1978.

[4] Lawrence, Rabiner. "Fundamentals of speech recognition." *AT&T* (1993).

[5] Kumar, Ankit, Mohit Dua, and Tripti Choudhary. "Continuous hindi speech recognition using monophone based acoustic modeling." *International Journal of Computer Applications* 24 (2014): 1-5.

[6] Aggarwal, Rajesh Kumar, and M. Dave. "Using Gaussian mixtures for Hindi speech recognition system." *International Journal of Signal Processing, Image Processing and Pattern Recognition* 4, no. 4 (2011): 157-170.

[7] Kurzekar, Pratik K., Ratnadeep R. Deshmukh, Vishal B. Waghmare, and Pukhraj P. Shrishrimal. "A comparative study of feature extraction techniques for speech recognition system." *International Journal of Innovative Research in Science, Engineering and Technology* 3, no. 12 (2014): 18006-18016.

[8] Rabiner, Lawrence R. "A tutorial on hidden Markov models and selected applications in speech recognition." *Readings in speech recognition* (1990): 267-296.

[9] Wani, Priyanka, U. G. Patil, D. S. Bormane, and S. D. Shirbahadurkar. "Automatic speech recognition of isolated words in Hindi language." In *2016 International Conference on Computing Communication Control and automation (ICCUBEA)*, pp. 1-6. IEEE, 2016.