Volume 2, Issue 2, February 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

The process of Feature Extraction in Automatic Speech Recognition System for Computer Machine Interaction with Humans: A Review

[1]Bhupinder Singh, [2]Rupinder Kaur, [2]Nidhi Devgun, [2]Ramandeep Kaur
[1]Dept. of Computer Sc. & Engg., IGCE Abhipur, Mohali (Pb.), India
[2]Dept. of Computer Sc. & Engg., Lovely Professional University Jalandhar(Pb.), India

Abstract- ***Automatic speech recognition (ASR) is an interesting field and lot of research work has been done by many research followers in the area of ASR and research in digital signal processing now computer hardware is able to communicate with humans in human language. Beside the presences of all these advancements and research in digital signal processing, computer machines are unable to match the performance of their human utterances in terms of accuracy of matching and speed of response. So now days the focused area of speech recognition process is to build a speaker independent ASR system. The reasons behind that it's vast number of applications, and drawbacks of available techniques of automatic speech recognition. In this paper we will discuss one of the process of speech recognition namely Feature Extraction. Commonly used spectral analysis, Parametric Transform and Statistical Modeling techniques of feature extraction are discussed in detail.***

Introduction

Automatic speech recognition by computers is a process where speech signals are automatically converted into the corresponding sequence of words in text or converted signal are used to operate machine with speech commands. The goal of Automatic Speech Recognition is to develop techniques and systems that enable computers to accept speech commands as an input [1]. The speech recognition problem may be interpreted as a speech-to-text conversion problem. Users want their voices, speech signals in to be transcribed into text by a computer. Speech recognition systems can be separated in several different classes by describing what type of utterances they have and the ability to recognize[2]. These classes are based on the fact that one of the difficulties of ASR is the ability to determine when a speaker starts and finishes an utterance. Isolated words[3], Connected words [4][5], Continuous speech and Spontaneous speech are main classes that are used in speech command recognition The speech recognition process is divided broadly into three parts, as shown in figure 1. Next we will discuss spectral analysis, Parametric Transform and Statistical Modeling techniques of feature extraction.

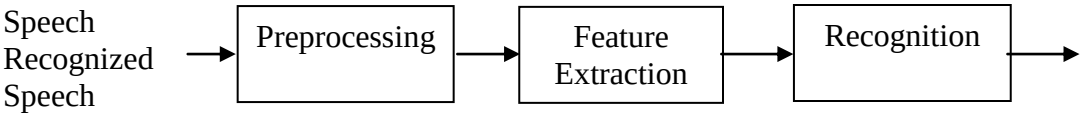

Figure 1: Speech Recognition Process

Feature Extraction

In speaker independent Automatic Speech Recognition Feature extraction is the process of retaining useful information of the signal while discarding redundant and unwanted information or we can say this process involves analysis of speech signal [6]. However, in practice, while removing the unwanted information, on may also lose some useful information in the process [7]. Feature extraction may also involve transforming the signal into a form appropriate for the models used for classification. In developing an ASR system, a few desirable properties of the features are:

- High discrimination between sub-word classes.
- Low Speaker variability.
- Invariance to degradations in the speech signal due to channel and noise.

The goal is to find a set of properties of an utterance that have acoustic correlates in the speech signal, that is, parameters that can somehow be computed or estimated through processing of the signal waveform,. Such parameters are termed features. Next step after the preprocessing of the speech signal in the signal modeling is feature extraction. Feature extraction is the parameterization of the speech signal. This is intended to produce a perceptually meaningful representation of the speech signal. Feature extraction typically includes the process of converting the signal to a digital from (i.e. signal conditioning), measuring some important characters of the signal such as energy or frequency response (i.e. signal measurement), augmenting these measurements with some perceptually-meaningful derived measurements (i.e. signal parameterization) and statistically conditioning these numbers to form observation vectors. The objective with feature extraction to attained are:

- To untangle the speech signal into various acoustically identifiable components.
- To obtain a set of features with low rates of change in order to keep computations feasible.

Feature extraction can be subdivided into three basic operations: spectral analysis, parametric transformation and statistical modeling [8]. The complete sequence of steps is summarized in next figure 2.

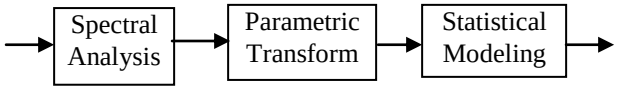

Figure 2: Feature Extraction Process

(a) Spectral Analysis: When speech is produced in the sense of time varying signal, its characteristics can be represented via parameterization of the spectral activity. There are six major classes of spectral of analysis algorithms i.e. Digital filter bank (Power estimation), Fourier Transform (FT Derived Filter Bank Amplitudes, FT Derived Cepstral Coefficients), Linear Prediction (LP, LP Derived Filer Bank Amplitudes, LP Derived Cepstral Coefficients) used in speech recognition system. From these classes, linear prediction gives best results. Types of Linear Prediction are explained as below:

***(i)* LPC (LPC analysis):** Linear Predictive Coding (LPC) has been popular for speech compression, synthesis and as well as recognition. Linear prediction models the human vocal tract as an infinite impulse response (IIR) system produces the speech signal. This is a very important spectral estimation technique because it provides an estimate of the poles (hence the formants) of the vocal tract transfer function. The LPC algorithm is a P^{th}

order linear predictor which attempts to predict the value of any point in a time-various linear system based on the values of the previous P samples. The all-pole representation of the vocal tract transfer function, H(z) can be represented by the following equation:

$$H(z) = \frac{G}{A(z)} = \frac{G}{1 + a_1 z^{-1} + a_2 z^{-2} + ...a_p z^{-p}}$$

The values *a(i)* are called the prediction coefficients while G represents the amplitude or gain associated with the vocal tract excitation. For the speech signal *s(n)* produced by a linear system, the predicated speech sample *ŝ(n)* is a function of *a(i)* and prior speech samples according to:

$$\hat{s}(n) = \sum_{i=1}^{p} a(i)s(n-i)$$

LPC analysis involves solving for the *a(i)* terms according to a lest error criterion. If the error is defined as:

$$e(n) = s(n) - \hat{s}(n)$$

$$s(n) = \sum_{i=1}^{p} a(i)s(n-i)$$

Then, taking the derivative of the square error with respect to the coefficients *a(j)* and setting it equal to zero gives:

$$\frac{\partial}{\partial a(j)}(s(n) - \sum_{i=1}^{p} a(i)s(n-1))^2 = 0$$

thus, $s(n)s(n-j) =$

$$\sum_{i=1}^{P} a(i)s(n)s(n-j) \text{ for } j = 1,....., P$$

There are tow principal methods for solving above equation for the prediction coefficients *a(i)*. The first is an auto correlation method, which multiplies the speech signal by a Hamming window or similar time window, assuming that the speech signal is stationary within and zero outside, the analysis window. The autocorrelation solution to equation can be expressed as

$$R(j) = \sum_{i=1}^{P} a(i)R(|i-j|) \quad j = 1,....., P$$

Where, *R(j)* is an even function and is computed from:

$$R(j) = \frac{1}{\gamma} \sum_{m=0}^{N-1-j} s(m)s(m+j) \quad j = 1,....., P$$

where, γ is a normalization factor. Once the autocorrelation terms *R(j)* have been calculated, a recursive algorithm named Levinson-Durbin Algorithm us used to determine the values of *a(i)*.

An alternative method for determining the LPC coefficients called the covariance method is a direct Cholesky decomposition solution of the following equation.

$$R(j) = a(i)R(|i-j|)$$

This equation can be expressed in matrix form. Unlike autocorrelation method, it does not use a window to force the sample outside the analysis interval to zero. Thus, the limits on the computation of *R(j)* extend from $-P \le n \le N-1-P$.

(ii) LP-derived filter bank amplitudes: Linear prediction derived filter bank amplitudes are defined as filter bank amplitudes resulting from sampling the LP spectral model (rather than the signal spectrum) at the appropriate filter bank frequencies. Now the question is how can one efficiently sample the spectrum given the LP model? A straightforward technique to computer filter bank amplitudes from the LP model involves direct evaluation of the LP model. The spectrum is typically over sampled and

averaged estimates are generated for actual filter bank amplitudes.

(iii) LP-derived cepstral coefficients: Another logical step in this direction would be to use the LP model to computer cepstral coefficients. If the Linear Prediction filter is stable (and it is guaranteed to be stable in the autocorrelation analysis), the logarithm of the inverse filter can be expressed as a power series in z^{-1}.

$$C_{LP}(z)=\sum_{i=0}^{N_c} C_{LP}(i)z^{-1} = \log H(z) = \log\left(\frac{G_{LP}}{\sum_{j=0}^{N_{LP}} a_{LP}(j)z^{-1}}\right)$$

It can solve for the coefficients by differentiating both sides of the expression with respect to z^{-1} and equating coefficients of the resulting polynomials. This results in the following recursion.

➔ Initialization $C_{LP}(0) = \log 1 = 0$, $C_{LP}(1) = a_{LP}(1)$

➔ For

$$2 \le i \le N_c,\ C_{LP}(i)=-a_{LP}(i)-\sum_{j=1}^{i-1}\left(1-\frac{j}{i}\right)a_{LP}(j)C_{LP}(i-j)$$

The coefficients C_{LP} are referred to as LP-derived Cepstral Coefficients. Historically, $C_{LP}(0)$ has been defined as the log of the power of the LP error. For now, it is noted that since power will be dealt with as a separate parameter, there is no need to include it in the equations above. It can regard the Cepstral model, in which $C_{LP}(0) = \log 1=0$. The number of Cepstral coefficients computed is usually comparable to the number of LP coefficients: $0.75p \le N_C \le 1.25p$.

The cepstral coefficients computed with the recursion described above reflect a linear frequency scale. One drawback to the LP-derived cepstral coefficients is that it must work a little harder to introduce the notion of a nonlinear frequency scale. The preferred approach is based on a method used to warp frequencies in digital filter design.

(b) Parameter Transforms: Signal parameters are generated from signal measurements through two fundamental operations: differentiation and concatenation. The output of this stage of processing is a parameter vector containing our raw estimates of the signal.

(i) Differentiation: To better characterize temporal variations in the signal, higher order time derivatives of the signal model. The absolute measurements previously discussed can be though of as Zero[th] order derivatives. In digital signal processing, there are several ways in which a first-order time derivative can be approximated. Three popular approximations are:

$$\tilde{S}(n) \equiv \frac{d}{dt}s(n) \approx s(n)-s(n-1)$$

$$\tilde{S}(n) \equiv \frac{d}{dt}s(n) \approx s(n+1)-s(n)$$

$$\tilde{S}(n) \equiv \frac{d}{dt}s(n) \approx \sum_{m=-N_d}^{N_d} ms(n+m)$$

The first two equations are known as backward and forward differences respectively. The first equation is same as pre-emphasis filter. The third equation represents a linear phase filter approximation to an ideal differentiator. This is often referred to as regression analysis.

The signal output from this differentiation process is denoted as delta parameter. The second-order time derivative can be similarly

approximated by reapplying third equation again to the output of the first order differentiator. This output is often referred to as a delta-delta parameter. Obviously, it can extend this process to higher other derivatives.

(ii) Concatenation: Most systems post process the measurements in such a way that the operations can be easily explained in terms of linear filtering theory. Here this notion is generalized in the form of a matrix operator. For research purposes, it is convenient to view the signal model as a matrix of measurements. The signal measurement matrix usually contains a mixture of measurements: power and a set of cepstral coefficients. The concatenation is the creation of a single parameter vector per frame that contains all desired signal parameters. Some parameters such as power, are often normalized before the computation. It is common to simply divide the power by the maximum value observed over an utterance (or subtract the log of the power).

With the emergence of Markov modeling techniques that provide a mathematical basis for characterizing sequential (or temporal) aspects of the signal, the reliance upon dynamic features has grown. Today, dynamic features are considered essential to developing a good phonetic recognition capability because rapid change in the spectrum is a major cue in classification of a phonetic-level unit.

(c) Statistical Modeling: The third step of the feature extraction process is Statistical Modeling. Here, it assumes that the signal parameters were generated from some underlying multivariate random process. To learn or discover the nature of this process, it impose a model on the data, optimize (or train) the model, and then measure the quality of the approximation. The only information about the process is its observed outputs, the signal parameters that have been computed. For this reason, the parameter vector output from this stage of processing is often called the signal observations. A statistical analysis is to be performed on the vectors to determine if they are part of a spoken word or phrase or whether they are merely noise. Speech sounds such as the 'ah' sound in the 'father' exhibit several resonance in the spectrum that typically extend for 120ms. Transitional sounds, such as the 'b' in 'boy' exist for a brief interval of approximately 20 ms. Statistical model in speech recognition is shown in figure 3.

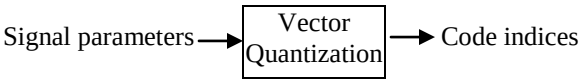

Figure 3: Statistical Models in Speech Recognition

Speech recognition system use extremely sophisticated statistical model, as this is one of the fundamental functions of a speech recognizer. Vector Quantization (VQ) has been useful in a wide variety of speech processing applications and forms the basis for the more sophisticated algorithm.

This basic concept of VQ applied to speech compression is schematically depicted in figure 4. A training speech sequence is first used to generate the codebook. The speech is segmented (windowed) into successive short frames and a vector of finite dimensionality represents each frame of speech. The vector may be in form of sampled data, FFT coefficients, autocorrelation terms, or their transformations (linear or non-linear). Codebook generation requires an iterative process much like a clustering algorithm involving a large number of spectral model vectors (codebook) so that the average spectral distortion from all the input vectors

to the same spectral compression strategy in the codebook generation process is executed in the quantizer. Each input vector is mapped to the codebook entry (code-word) index corresponding to the best match vector. Speech compression or rate reduction is accomplished by using the indexes as storage or transmission parameters. For Vector Quantization, it is necessary to have a measurement of dissimilarity between the two vectors. Distortion measures based upon transformation, which retain only the smoothed behavior of the speech signal, have been applied in speech recognition, speaker identification and verification tasks.

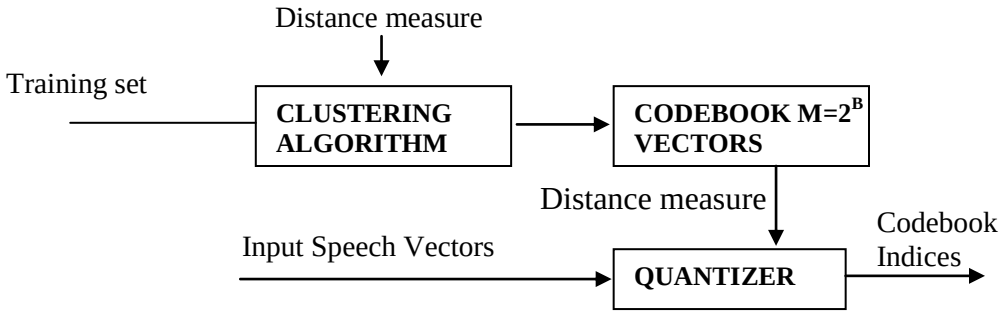

Figure 4: Vectors Quantization Training and Classification Structure

To build a Vectors Quantization (VQ) codebook and implement a VQ analysis procedure, one needs the following:

- A larger set of spectral vectors, *{xj; j=0,…,n-1}*, which form a training set. The training set is used to create the optimal set of the codebook vectors for representing the spectral variability observed in the training set.
- A distance measure between a pair of spectral analysis, so as to able to cluster the training set vectors as well as to classify arbitrary spectral vectors into unique codebook entries.
- A centroid computation procedure: On the basis of the partitioning that classifies the training vectors into the M clusters, choose the M codebook vectors as the centroid of each of the M clusters.
- A classification procedure for arbitrary speech spectral analysis vectors that choose codebook vector closest to the input vector and uses the codebook index as the resulting spectral representation.

Results of System

The results of speech recognition systems are mainly specified in terms of accuracy of matching and speed of detection. Accuracy may be measured and depend upon the result of accurate detection of words, whereas speed is measured with the detection of words with fast response of Automatic Speed Recognition system.

Conclusion

In this review paper have discussed one of the process of speech recognition namely Feature Extraction and its commonly used spectral analysis, Parametric Transform and Statistical Modeling techniques. Six major classes of spectral of analysis algorithms i.e. Digital filter bank (Power estimation), Fourier Transform (FT Derived Filter Bank Amplitudes, FT Derived Cepstral Coefficients), Linear Prediction (LP, LP Derived Filer Bank Amplitudes, LP Derived Cepstral Coefficients) used in speech

recognition system, differentiation and concatenation fundamental operations of Parameter Transforms, Vector Quantization of signal modeling are discussed properly.

References:

1. Yousef and Amir (2010), "*Comparative Analysis of Arabic Vowels using Formants and an Automatic Speech Recognition System*", International Journal of Signal Processing, Image Processing and Pattern Recognition, Vol. 3, No. 2, pp. 11-22.
2. Anusuya M. and Katti S. (2009), "*Speech Recognition by Machine: A Review* " International Journal of Computer Science and Information Security, Vol. 6, No. 3, pp. 181-205.
3. Gubian, M., Arnone, L. and Brofferio, S. (2005), "*A Quantitative Method for Performance Analysis of an Isolated word ASR System"*, in proceedings of 13th European Signal Processing Conference (EUSIPCO), Turkey, pp. 1-4.
4. Rabiner, L. and Sambur, M. (1976), "*Some Preliminary experiments in the recognition of connected digits*", IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 24, Issue 2, pp. 170-182.
5. Rabiner, L. and Levinson, S. (1981), "*Isolated and Connected word Recognition Theory and selected applications*", IEEE Transactions on Communications, Vol. 29, Issue 5, pp. 621-659.
6. Stolcke A., Shriberg E., Ferrer L., Kajarekar S., Sonmez K., Tur G.(2007)*," Speech Recognition As Feature Extraction For Speaker Recognition***"** SAFE, Washington D.C., USA pp 11-13.
7. Kesarkar M. (2003), **"**Feature Extraction For Speech Recogniton**"** M.Tech. Credit Seminar Report, Electronic Systems Group, EE. Dept, IIT Bombay.
8. Becchetti, C. and Ricotti, L. (2004), "*Speech Recognition Theory and C++ Implementation",* John Wiley & Sons, Wiley Student Edition, Singapure, pp. 121-188.