School of Engineering and Applied Sciences Ahmedabad University
MA202-Probability and Random Processes

Speech Recognition System

Group 9

Om Thakkar (201501109)
Shivam Raval (201501088)

Background and Motivation

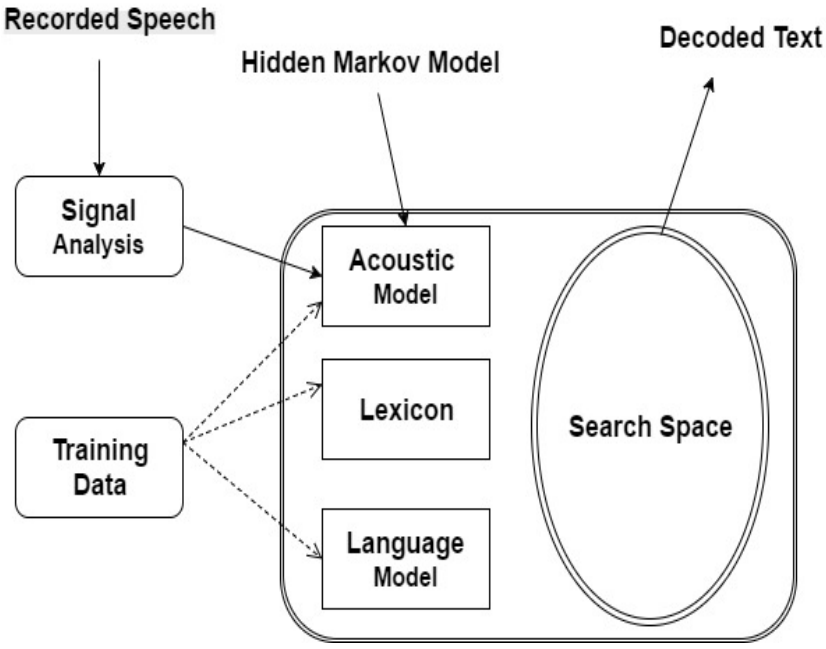
Speech Recognition System also commmonly known as Speech to Text is a software technology that lets the user control the computer functions and dictate text by voice hence reducing human efforts. Speech Recognition System or the problem of Automated Speech Recognition has been studied over past several decades. Speech Recognition Systems have revolutionized the approach of people with disabilities and have made lives easier by reducing Human Efforts.

Early speech recognition systems tried to apply a set of grammatical and syntactical rules to speech. If the words spoken, fit into a certain set of predefined rules, the Speech Recognition System could determine what the words were. However, there were certain limitations such as different densities of voice, varied accents, change in dialects and mannerisms in the human speech which eventually had a lot of flaws in the approach. Today's Speech Recognition Systems use powerful and statistical modeling systems. The current systems use concepts like Probability and Mathematical Functions to determine the likelihood of an event.

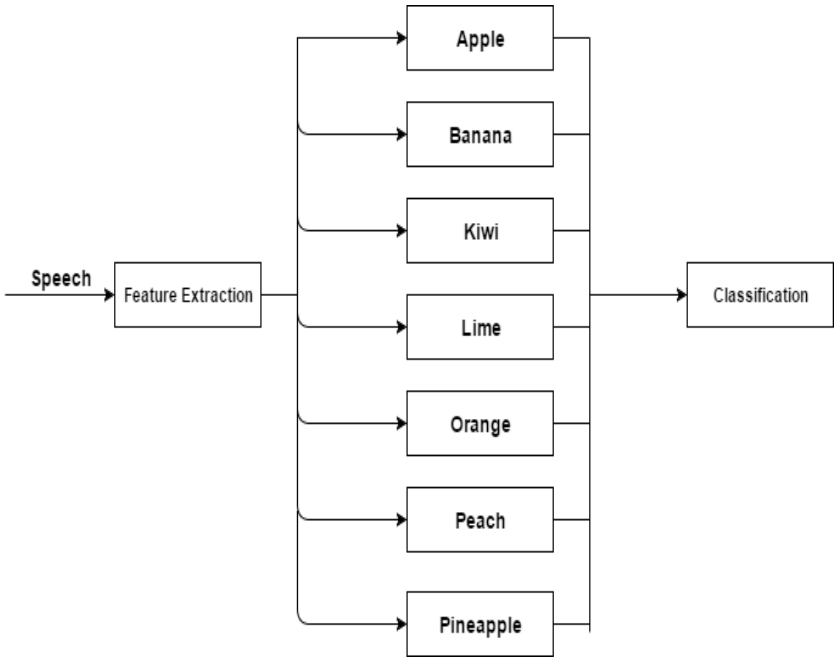
Hidden Markov Models and Neural Networks methods involve complex mathematical functions. In HMM, speech generation is modeled as a Stochastic Process. Hidden Markov Models are expressed in terms of phonetic states and acoustic emissions and are parameterized by transition and emission probabilities. In this model, each phoneme is like a link in a chain, and the completed chain is a word. However, the chain branches off in different directions as the program attempts to match the digital sound with the phoneme that's most likely to come next. During this process, the program assigns a probability score to each phoneme, based on its built-in dictionary and user training.

In our Special Assignment, the speech recognition system implemented during our Special Assignment trains one hidden Markov model for each word that it should be able to recognize. The models are trained with labeled training data, and the classification is performed by passing the features to each model and then selecting the best match using Hidden Markov Model and algorithms associated with Probabilistic Modelling like Baum-Welch Algorithm which makes use of Forward Algorithm, Backward Algorithm and Expectation Maximization (EM) algorithm to find the maximum likelihood estimate of the parameters of a Hidden Markov Model.

Block Diagram and Flow Chart :



(a) Block Diagram of Speech Recognition System



(b) Flowchart of Speech Recognition System

About the Probabilistic Model Used

The Probabilistic Model that has been used and implemented in "**Speech Recognition System**" is **Hidden Markov Model**. Hidden Markov Model (HMM) is a statistical Markov Model in which the system being modeled is assumed to be a Markov process with unobserved (hidden) states.

The Hidden Markov Model is a finite set of states, each of which is associated with a (generally multidimensional) probability distribution. Transitions among the states are governed by a set of probabilities called transition probabilities. In a particular state an outcome or observation can be generated, according to the associated probability distribution. It is only the outcome, not the state visible to an external observer and therefore states are "hidden" to the outside; hence the name Hidden Markov Model.

Why Hidden Markov Model ?

In Continuous Automatic Speech Recognition (ASR) System, both the input and output may be varied in length. In early days of ASR, this variable-length input problem was handled by Dynamic Time Warping (DTW) which was quickly subsumed by Hidden Markov Model. HMM captures the temporal elasticity of speech as well as provides a rigorous framework for modeling the relationship between acoustic features (observation) and a relatively small set of phones (hidden states). The properties of HMM are well understood, with many sophisticated and efficient algorithms for training and decoding developed around it. These factors have made HMM incredibly popular in ASR, and have resulted in huge improvement over DTW.

The observable output from a hidden state is assumed to be generated by a Multivariate Gaussian distribution, so there is one mean vector and covariance matrix for each state. We also assume that the state transition probabilities are independent of time, such that the Hidden Markov Chain is homogenous.

Psuedo Code / Algorithm

Let us begin by defining Notations for describing Hidden Markov Model (HMM) as used in the Special Assignment.

Total Number of States : N

An element a_{ss}, in the transition probability matrix A denotes the transition probability from state s to state s', and the probability for the chain to start in state s is π_s.

The mean vector and covariance matrix for the Multivariate Gaussian Distribution modeling the observable output from state s are μ_s and $\sum_s$, respectively. For an observation o, $b_s(o)$ denotes the probability density of the multivariate Gaussian distribution of state s at the values of

o. We will sometimes denote the collection of parameters describing the hidden Markov model as $\lambda = \{A, \pi, \mu, \sum\}$.

Forward Algorithm

The Probability Density Function of an observation $o_1, o_2, ..., o_T$ for a specific model needs to be calculated in order to select the model (i.e word) that most likely generated the speech signal.

$$f(o_1, ..., o_T : \lambda) = \sum_{s_T} f(o_1, ..., o_T, s_T : \lambda)$$

$$= \sum_{s_T} f(o_T | o_1, ..., o_{T-1}, s_T; \lambda) f(o_1, ..., o_{T-1}, s_T; \lambda)$$

$$= \sum_{s_T} b_{s_T}(o_T) \sum_{s_T - 1} f(o_1, ...o_{T-1}, s_{T-1}, s_T; \lambda)$$

$$= \sum_{s_T} b_{s_T}(o_T) \sum_{s_T - 1} f(s_T | o_1, ...o_{T-1}, s_{T-1}; \lambda) f(o_1, ...o_{T-1}, s_{T-1}; \lambda)$$

$$= \sum_{s_T} b_{s_T}(o_T) \sum_{s_T - 1} a_{s_{T-1} s_T} f(o_1, ..., o_{T-1}; \lambda)$$

The recursive structure is revealed as we reduced the problem from needing $f(o_1, ...o_{T-1}, s_{T-1}; \lambda)$ for all s_T to needing $f(o_1, ..., o_{T-1}, s_T; \lambda)$for all s_{T-1}. Let us introduce the forward variable to ease the notation.

$$\alpha_1(s) \equiv f(o_1, S_1 = s; \lambda)$$
$$= b_s(o_1)\pi_s$$
$$\alpha_t(s) \equiv f(o_1, ..., o_t, S_t = s; \lambda)$$
$$= b_s(o_t) \sum_{s'} a_{s's} \alpha_{t-1}(s')$$

The solution can also be expressed as,

$$f(o_1, ..., o_T; \lambda) = \sum_s \alpha_T(s)$$

Baum Welch Algorithm

Baum Welch Algorithm will be used to find the parameters λ that maximize the likelihood of the observations. This will be used to train the hidden Markov model with speech signals. The Baum-Welch algorithm is an iterative expectation-maximization (EM) algorithm that converges to a locally optimal solution from the initialization values.

The M(Maximization) step consists of updating the parameters in the following way:

$$\pi_s := \overline{\pi_s} = \frac{\text{expected number of times in state s at t = 1}}{\text{expected number of times at t = 1}}$$

$$a_{ss'} := \overline{a_{ss'}} = \frac{\text{expected number of transitions from s to s'}}{\text{expected number of transitions from s}}$$

$$\mu_s := \overline{\mu_s} = \text{expected observation when in state s}$$

$$\sum_s := \overline{\sum}_s = \text{observation covariance when in state s}$$

The E(Expectation)-step thus consists of calculating these expectations for a fixed λ. Let $V_s^{(}t)$ denote the event of transition from state s at time step t, and $V_{s,s'}^{(}t)$ the event of transition from s to s' at t. Then we calculate these expectations by using indicator functions and linearity of expectation.

$$\overline{\pi_s} = E[1[V_s^{(1)}]] = P(V_s(1))$$

$$\overline{a_{ss'}} = \frac{E[\sum_t 1[V_{s,s'}^{(t)}]]}{E[\sum_t 1[V_s^{(t)}]]} = \frac{\sum_t P(V_{s,s'}^{(t)})}{\sum_t P(V_{s,s'}^{(t)})}$$

$$\overline{\mu_s} = \frac{E[\sum_t 1[V_s^{(t)}]o_t]}{E[\sum_t 1[V_s^{(t)}]]} = \frac{\sum_t P(V_s^{(t)})o_t}{\sum_t P(V_{s,s'}^{(t)})}$$

$$\overline{\sum_s} = \frac{E[\sum_t 1[V_s^{(t)}](o_t o_t^T - \overline{\mu_s \mu_s}^T)]}{E[\sum_t 1[V_s^{(t)}]]} = \frac{\sum_t P(V_s^{(t)})o_t o_t^{(T)}}{\sum_t P(V_{s,s'}^{(t)})} - \overline{\mu_s \mu_s}^T$$

To be able to calculate these probabilities we need the backward variable which is very similar to the forward variable previously defined.

$$\beta_T(s) \equiv 1$$

$$\beta_T(s) \equiv f(o_(t+1), ..., o_T | S_t = s; \lambda)$$

$$= \sum_{s'} a_{ss'} b_{s'}(o_{t+1}) \beta_{t+1}(s')$$

Renaming the probabilities to the same symbols as used by Rabiner and express them by forward and backward variables:

$$\gamma_(t)(s) \equiv P(V_s^{(t)}) = P(S_t = s | o_1, ..., o_T; \lambda)$$

$$= \frac{f(o_1, ..., o_T | S_t = s) P(S_t = s)}{f(o_1, ..., o_T)}$$

$$= \frac{f(o_1, ..., o_t, S_t = s) f(o_{t+1}, ..., o_T | S_t = s)}{f(o_1, ..., o_T)}$$

$$= \frac{\alpha_t(s)\beta_t(s)}{f(o_1, ..., o_T)}$$

Similarly,

$$\varepsilon_t(s, s^{'}) \equiv P(V_{s,s^{'}}^{(t)}) = P(S_t = s, S_{t+1} = s^{'} | o_1, ..., o_T; \lambda)$$

$$= \frac{\alpha_t(s) b_{s^{'}}(o_{t+1} a) a_{ss^{'}} \beta_{t+1}(s^{'})}{f(o_1, ..., o_T)}$$

Finally, we get,

$$\overline{\pi} = \gamma_1(s)$$

$$\overline{a_{ss'}} = \frac{\sum_t \varepsilon_t(s, s^{'})}{\sum_t \gamma_t(s)}$$

$$\overline{\mu_s} = \frac{\sum_t \gamma_t(s) o_t}{\sum_t \gamma_t(s)}$$

$$\overline{\sum_s} = \frac{\sum_t \gamma_t(s) o_t o_t^T}{\sum_t \gamma_t(s)} - \overline{\mu_s \mu_s}^T$$

To summarize the E-step boils down to computing $\gamma_(t)(s)$ and $\varepsilon_t(s, s^{'})$ for all s, s' and t while the parameters λ are fixed, and then the M-step will update λ by using the calculations done in the E-step. This is iterated until satisfaction.

Classification

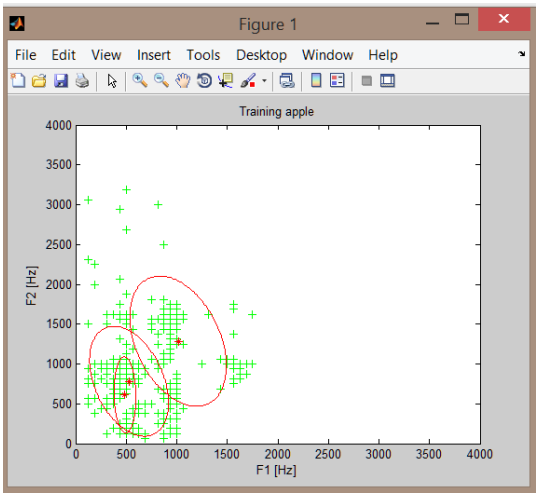
Let λ_i denote the parameter set for word i. When presented with an observation $o_1, o_2, ..., o_T$, the selection is done as follows:

$$\text{predictedword} = \arg\max_i f(o_1, o_2, ..., o_T; \lambda_i)$$

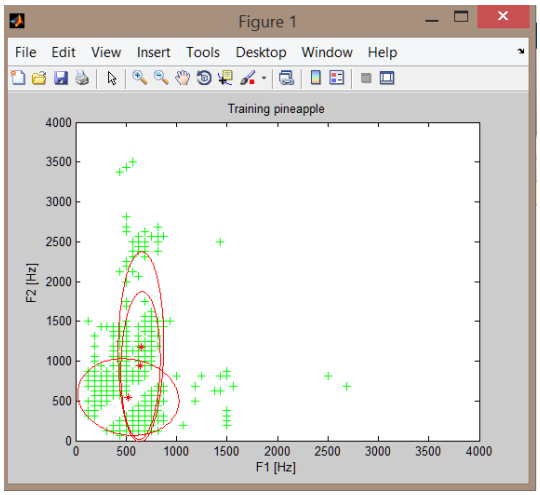
Results/Inferences

For each of the seven words considered, 15 utterances have been recorded for each word making a total of 105 utterances in all. Experimentation indicated that the most important parameters were the number of Hidden States, N and the number of frequencies extracted from each frame, D.

Experimental Setup and Results for words "Apple" and "Pineapple" are as shown below :



(c) Apple



(d) Pineapple

We have tried and experimented the system for different values of N and D. The observations are as follows:

N\D	2	3	4	5	6	7	8
2			21.9%		8.6%		
3	21.0%	15.2%	9.5%	12.4%	1.9%	14.3%	5.7%
4	16.2%	11.4%	8.6%	5.7%	3.8%	6.7%	4.8%
5	13.3%	8.6%	9.5%	4.8%	2.9%	5.7%	4.8%
6	12.4%	10.5%	3.8%	5.7%	7.6%	6.7%	10.5%
7	15.2%	12.4%	6.7%	10.5%	7.6%	2.9%	8.6%
8			12.4%			5.7%	

The inferences that have been derived based on the experimentation are that the results are quite good as compared the simple approach taken, especially in the feature extraction phase. One important thing to note is that this system would not perform well if trained and tested with different speakers. This is because of the different frequency characteristics of different voices, especially for speakers of different gender. It is also interesting to note that when N is too small, there are many apple′s misclassified as pineapple′s, and vice versa, due to the loss of temporal information.

Future Scope

During this Special Assignment, a system for isolated-word speech recognition was implemented and tested. The cross-validation results are good for a single speaker. Two obvious extensions can be that the system supports for several speakers, and support for continuous speech. The first step towards the former would be more, and more robust, features. For the latter the simplest approach is probably to detect word boundaries and then proceed with an isolated-word recognizer.

References

[1] Mathias De Wachter, et al., "Template Based Continuous Speech Recognition", in IEEE Transactions on Audio, Speech and Language Processing, Vol. 15, No. 4, May 2007.

[2] Nirav S. Uchat, "Hidden Markov Model and Speech Recognition", in Seminar at Department of Computer Science and Engineering Indian Institute of Technology, Bombay Mumbai.

[3] Su Myat Mon, Hla Myo Tun, "Speech-To-Text Conversion (STT) System Using Hidden Markov Model (HMM)", in International Journal of Scientific and Technology Research Volume 4, Issue 06, June 2015.

[4] Lawrence R. Rabiner, Fellow, IEEE, "A Tutorial on Hidden Markov Model and Selected Applicaiton in Speech Recognition", Proceedings of the IEEE, Vol. 77, No.2, February 1989.