Summary of “A review of Large-vocabulary Continuous-speech Recognition.”

Siddharth Sharma

This paper discusses main components of speech recognition system in late 90’s.
The basic pipeline to convert speech to text is as follows:

1. A front-end signal processor converts speech waveform to sequence of acoustic vectors (compact representation of short time speech spectrum).
2. Phones are modelled using HMM’s. These can be stacked together to create HMM’s for words.
3. Language modelling is done to predict words based on previous words.
4. Sequence of words that have maximum probability according to above language model and word probabilities are extracted.

These steps are discussed in more details below.

Front-End Parameterization

Spectral characteristics of input signal are assumed constant. Input signal is divided into blocks and then several transformations like Fourier transform are used to compute required spectral estimates.The final preprocessing generally involves spectral information compressing to get low order coefficients.The transforms which mimic human hearing sensitivity(linear up to 1000 Hz and logarithmic afterword) has also shown increase in accuracy. These transforms have evolved so that the vector sequence generated satisfies constraints of acoustic model. For example acoustic model assumes vectors to be uncorrelated with neighbours but they are not, so this is remedied by appending first and second order differentials.

Acoustic Modelling

Here we calculate the likelihood of vector sequence **Y** given a word w. Here phones are modelled instead of words because of impracticality of modelling words in large vocabulary systems. These phones models can be joined easily to form word models. In practice simply modelling individual phones is not enough due to contextual variances so triphones are used, but this has too many parameters to estimate and not enough data. Tied Mixture models, in which data with states which are acoustically indistinguishable are pooled, are used to solve this problem. Phonetic decision trees are used to cluster states to be tied together.
The required probability P(Y|M) is calculated by Forward-Backward algorithm which efficiently sums over all possible state sequences or alternatively can be done by Viterbi Algorithm(which assumes highest probability sequence is dominant. Emission and Transition probability estimation can be done by Baum-Welch Algorithm. P(Y,X|M) is dominated by output probability which is generally modelled using Gaussian Mixture Models.

Language Modelling

It gives mechanism for predicting probability of a word given previous words. In general N-gram models are used for this estimation. Large number of n-grams can cause problem of training data sparsity. Discounting(distributing excess probability to less frequently occurring words) and backing-off(replacing ngram with scaled (n-1)gram probability) are used to remedy these problems.

Decoding
Sequence of words that maximises P(W)P(Y|W) is a search problem. Depth-first search methods like A* decoder and stack decoders or Breadth-first methods are generally used. These methods cannot be directly used on word graphs as they do not allow trigram language model and triphones to be used. In general words are replaced by sequence of model representing its pronunciation. And identical phone model in identical context are merged to reduce size. As whole tree is intractable in general only parts that are being currently required are being constructed and even in them pruning(beam search) based on some score of tree nodes(like probability till that part which can be calculated by adding log state transition, log language model and log output probabilities from start to tree node) is used to make tree manageable. The scores can be calculated easily by token passing(similar to viterbi search).

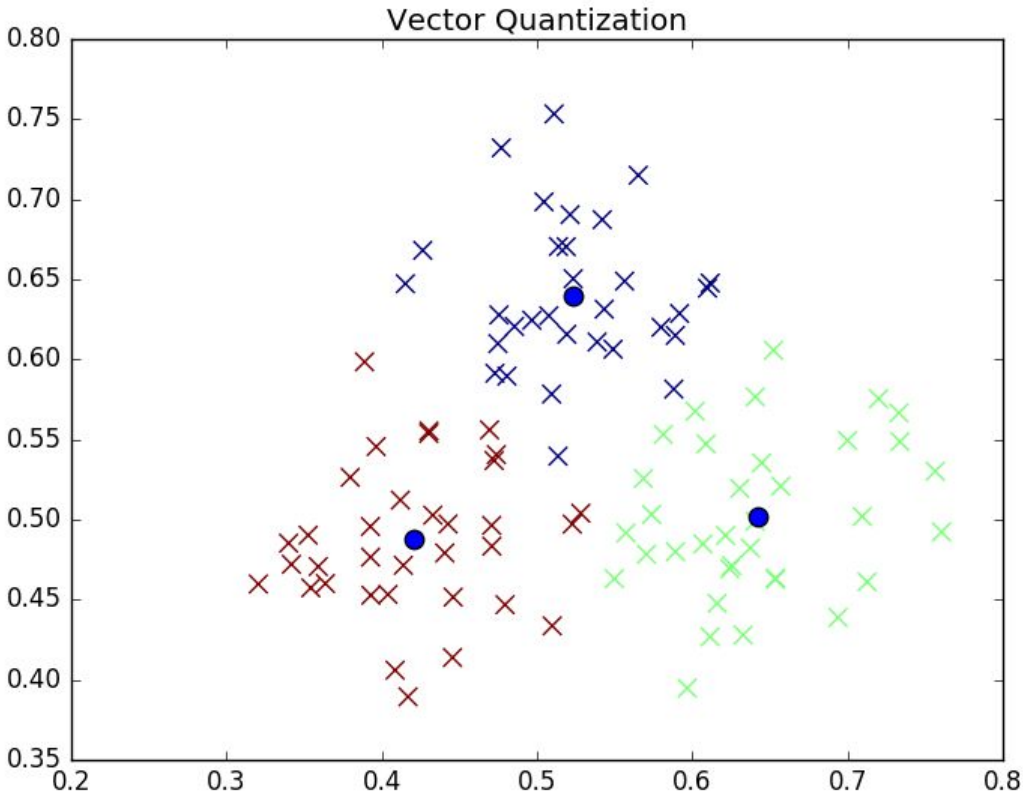
Benchmarking
US Advanced Research Project Agency CSR Evaluations were used. The main test was hub test H1 which had two parts one containing fixed training data and in other any data could be used. The lowest error rate at that time was of HTK LV. It was 7.2% word error rate.

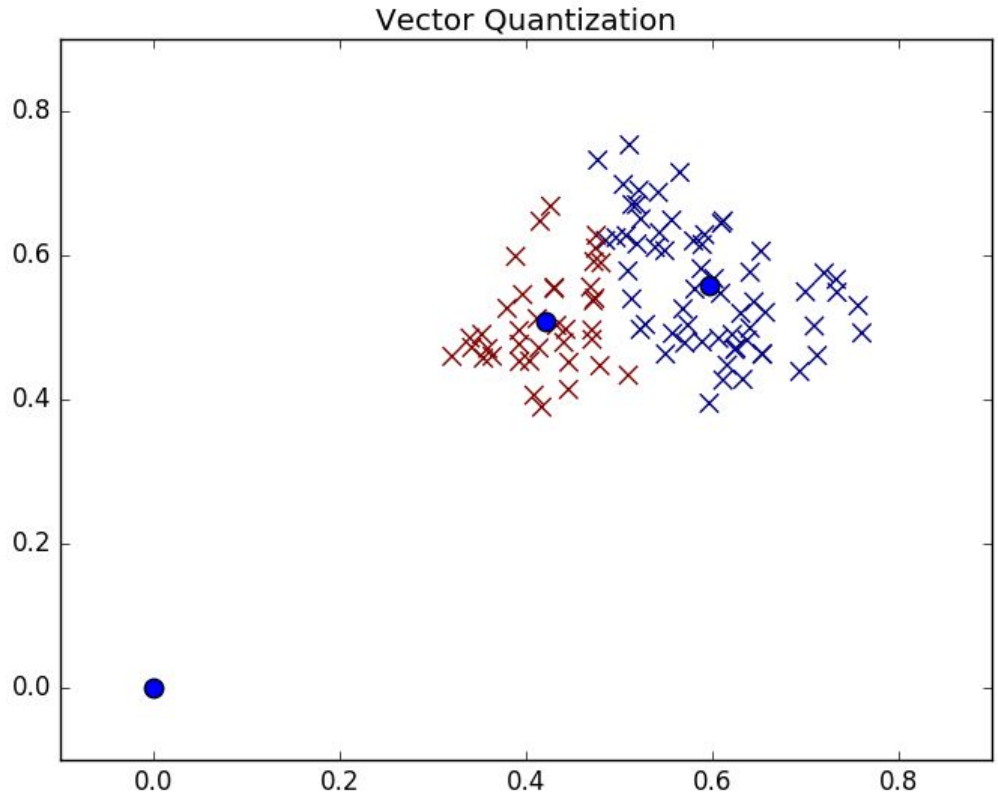
Issues
Speaker adaptation: Can be used to increase accuracy when speakers are regular users or fixed. Both supervised and unsupervised methods can be used though unsupervised methods are prefered due to minimal user interaction. The adaptation should be generally involve adapting based on small amount of new data.
Environmental Robustness: Background noise can increase error rate significantly. This is due to the fact that noise changes the whole distribution rather than just mean and variance. Noise can be dealt by either processing it at frontend or making models that can adapt to it.
Task Independence: Difference in vocabularies in various scenarios like office correspondence or legal documents make LM trained for one scenario not very useful for others. This can dealt by using class based rather than word based language models.
Spontaneous speech: There is large error when models are used for spontaneous speech due to factors like poor articulation, variable speaking rate etc. Not much work was done on this area at this point in time.
Real time operations: Computationally efficient implementation is required for model to be useful in actual world. Several techniques like approximations, lookahead and pruning can be used to make models computationally efficient.

Question 3

If the points from data are taken as initial centers we generally get three clusters with 33 points each approximately.

But if the initial centers are sampled from a uniform distribution we can get one or two clusters with no points. This is because our points lie in approximate x range of [.3, .8] and y range of [.4, .75] so if one of the centers gets initialized to [0,0] it get no point in vicinity. Better way would be to scale points to range [0,1] and then use uniform distribution.




Vector Quantization
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8