1.5 ASR SYSTEM

Charalambos Theodorou

1 Description of Acoustic and Language Models

To start up with the ASR system should be Multilingual in order to support the various languages of the countries that the client have requested(USA, UK, South Africa and India). ASR objective is to the most likely word sequence W* according to the acoustic observations, the acoustic, the pronunciation lexicon and the language model (Figure 1).

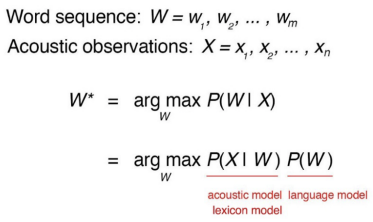

Figure 1: Word sequence and acoustic observations.

The first step is to extract features from audio frames using a sliding window. Each audio frame will contain say 39 MFCC features. This forms a sequence of our observations X (frames: x_1, x_2, x_3,..., x_i , ...). The extracted features include information of the formats in identifying a phone as well as their first-order and second-order derivative in understanding their context. The next step is to reverse engineering the internal sequence of states (s_1, s_2, s_3, ...) from the observed feature vectors X. The key idea is finding a state sequence (a path) that maximizes the likelihood of the observations. Figure 2 contains the probability of a path and the likelihood of an observation given an internal state. Let's look into the second part closer. Each word will be modeled by a pronunciation lexicon and an HMM. The self-looping of a state allows ASR to handle different duration of phones in utterances. The likelihood of an observation given a state will be modeled by an m-component GMM.

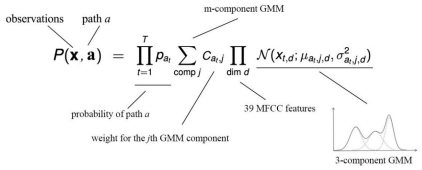

Figure 2: The probability of a path and the likelihood of an observation given an internal state.

Searching for such sequence one-by-one, even with limited sequence length T, is hard. With a vocabulary size of k, the complexity is $O(k^T)$ and grows exponentially with the number of audio frames. Viterbi decoding can solve the problem recursively. For each time step, Viterbi decoding computes the maximum path for a node using results from the last time step. So we can search the exact maximum path in $O(k^2T)$.

Nevertheless, to increase ASR accuracy, we need to label the phone with its context also. Unfortunately, this grows the internal states to 3×50^3 states if we start with 3×50 internal phone states. To

address that, some labels with similar articulation will share the same acoustic model (the GMM model). Figure 3 demonstrates the journey from 3 states per context-independent phone to 3 states per triphone using GMM.

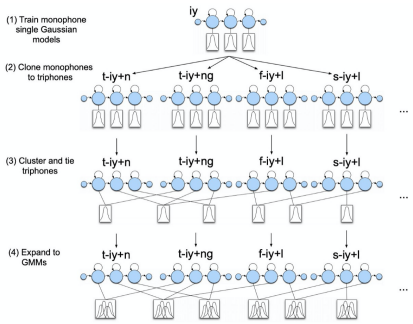

Figure 3: Journey from 3 states per context-independent phone to 3 states.

2 Preparing the Acoustic and Language models to optimize the order taking accuracy.

First thing first in oder to optimize the order taking accuracy it will be necessary to collect more data(one was could me data mining) for the models and train them further. Training commercial quality ASR takes weeks using a cluster of machines because ASR trains the model in stages (multiple passes). To improve the accuracy of ASR, the transition probabilities between words need to be compute using a language model with the most common one language model the bigram(In a bigram (2-gram), the next word depends on the previous word only). Viterbi decoding finds the best state sequence, both FB and Viterbi training are EM algorithms in optimizing the HMM model parameters and the acoustic models in alternating steps. Viterbi training is less computational intense but in early training, the acoustic models are less mature. FB may be considered. In practice, speed and simplicity may dominate the training result. When the HMM models become more mature, the difference in the soft assignment and the hard assignment are not significant and will produce the same result.

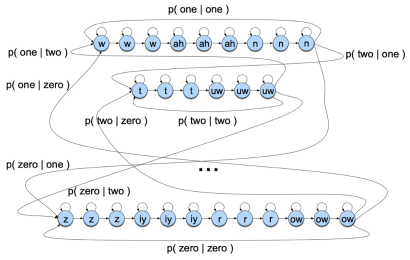

Figure 4: Bigram.

We can then add a language model to LVCSR(Figure 5), but the diagram can become crazily complex. The language model is modeled separately with a corpus using simple occurrence counting. Composing all the models together (H ∘ C ∘ L ∘ G) but with that there are new challenges for LVCSR in training such as pronunciation is sensitive to neighbor phones (context) inside a word and between words, the alignment between HMM states and audio frames are harder for continuous speech and a large vocabulary triggers a lot of states to keep track of. For accuracy, LVCSR needs a complex acoustic model and a far larger number of HMM states to model the problem. The first issue will be addressed by the GMM for now and the second issue will be addressed by triphones to take phone context into consideration. With the basic HMM topologies to be the same as a single word recognizer, many training concepts can be reused. But there are some important exceptions. First, the acoustic model for each HMM state is more complex. That introduces a major headache in training. As the complexity grows, it may get stuck in bad local optima. Like other complex ML

models, we need a good strategy to learn it. Second, we need to perform alignment more frequently. Third, the potential number of acoustic models is so large that similar-sound HMM states need to share the same acoustic model.

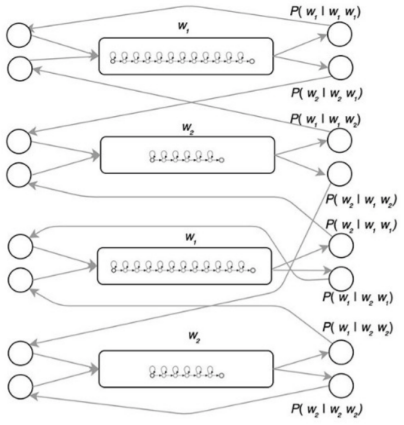

Figure 5: Bigram.

2.1 Tunable

Tuning the parameters α, β can control the sampling of languages during token generation and training examples during multilingual model training respectively. In general, from going from natural frequency (α = 1,β = 1)to uniform frequency(α = 0,β = 0) seems to improve performance of low resource languages while degrading performance on high resource languages. Interestingly, it appears the using a α = 0.5 and β = 0.5 performs best on low resource languages and has less performance degradation on high, mid resource languages compared to sampling at uniform frequency (α = 0 and β = 0). For low resource language, one might assume that sampling a language more frequently will always result in better performance. Sampling at the natural frequency has too much data imbalance to learn an effective shared representation, while sampling at the uniform distribution overfits to the low resource languages.

2.2 Multilingual transfer learning on unseen languages

To improve the accuracy the multilingual models should be train on a large, diverse set of languages to enable the acoustic models to learn language agnostic representations general enough to perform well on completely new languages that the client asked (USA, UK, South Africa, India). To do that, the joint model should be tuned with parameters on the unseen languages(USA, UK, South Africa, India) with approximetly 200 hours of training data or more. Since, the graphemes in new languages, which are being fine tuned, are not present in the decoder of trained joint model, the decoder should be re-initialize for the grapheme set of new language and both encoder and decoder should be rained during fine-tuning.

3 Conclusion

On the previous chapter the ASR model was described and how it could be implemented to recognize food telephone orders. Moreover the preparation of the Acoustic and Language models of how to optimize the order taking accuracy were described(2.1). In addition another method was introduced(2.2) that could be useful for the training of the model in order to be able to recognize the languages that the client wants(USA, UK, South Africa, India). Finally in my opinion the initial most important step is to gather a lot of data in order to train a precise and accurate model.

4 References

Michael P, Bhuvana R, Stanley F. C, Markus N T. (2016). The Big Picture/Language Modeling.

Mark W. An Automatic Speech Recognition System for a Robot Dog.