ISCA Archive
http://www.isca-speech.org/archive

6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

UNIFIED DECODING AND FEATURE REPRESENTATION FOR IMPROVED SPEECH RECOGNITION

Li Jiang and Xuedong Huang

http://research.microsoft.com/srg
Microsoft Research
Redmond, Washington 98052, USA

ABSTRACT

In this paper we propose a unified framework for decoding and feature representation based on the *Maximum A Posterior* (MAP) principle. The search space is augmented with an additional feature stream dimension such that different feature representations can be utilized for different phonetic context under the HMM decoding framework. We also provide a theoretic explanation for the unified framework. It gives us "supervised" signal processing and feature extraction for the recognition system, which has reduced the word recognition error rate by 15% on a large-vocabulary continuous speech recognition task when multiple feature streams are used simultaneously.

1. INTRODUCTION

A traditional speech recognition system has a feature extraction module that is typically detached from other acoustic and language models. The Unified Stochastic Engine (USE) [1] can jointly optimize acoustic and language models, but the link between feature representation and acoustic/language models remains missing. In this paper, the USE is extended to include feature representations such that different phonetic models can use optimal feature representation that has maximum a posterior probability.

There are a number of different speech feature representations for modern speech recognition systems. Mel-Frequency Cepstrum Coefficients (MFCC) is one of the most popular representations because of its superior recognition accuracy in most situations. However, it is sub-optimal in the sense that the parameters of such a representation cannot be adjusted dynamically for a wide range of acoustic-phonetic contexts. For MFCC, the parameters of the window size, bandpass filters, and the dimension of the representation are all fixed. It is plausible that different features or features with different parameters can better describe certain acoustic-phonetic classes. For example, different window size should result in different time/frequency resolution and better time resolution is often needed for rapidly changing transient classes.

The effort to find a "perfect" feature representation remains an ongoing quest by many researchers. In this paper, we want to attack the problem from a different angle. Instead of using a single perfect feature representation, we can have a massive parallel feature representation matrix that consists of a large number of different feature representations. We hope such a combined stream can be used to cover most of situations needed for robust speech recognition. For example, we could have MFCC with a large number of varying parameters so that it has a good quantized coverage in terms of time/frequency resolution. We can integrate these features with other components in the speech recognition system. Thus, all the features can be considered and different features or a combination of them can be used for different acoustic units in an optimal way that is consistent with the HMM framework. The combination mixture weights can depend on the acoustic unit and the relevant context, estimated by criteria such as Maximum Likelihood (ML) or Minimum Classification Error (MCE) criterion. This is similar to the motivation of the original USE – to jointly optimize the feature representations and acoustic/language models.

There are a number of special cases of this general unified framework. For example, the mixture weights can be dependent on acoustic units such as senones. They could be estimated with the minimum classification error (MCE) criteria. A further simplification would be to have uniform weights for all the feature streams. In fact, the uniform- weight combination of frame-level scores had been investigated by Kingsbury and Morgan [2], Hallberstadt and Glass [3], Kirchhoff and Bilmes [4]. In these systems, the emission probabilities are combined (with certain rules such as sum or product) using the mixture weights and decoding is performed using the combined probabilities. Kinsbury and Morgan combined 4 different classifiers for clean and reverberant speech. Hallberstadt and Glass used multiple segmental features with product rule and applied it to phonetic and word recognition. Kirchhoff and Bilmes investigated different combination rules and also investigated the use of confidence measures to determine weights. Both reported considerable performance improvements by using multiple feature representations.

In our implementation, the weights of different feature streams depend on runtime score of these feature streams. They are thus not only context dependent but also test data dependent. The selection of different features guarantees a *Maximum A Posterior* (MAP) probability for the test data. The *multiple-feature decoding* uses the decoder to select feature for each acoustic unit based on the MAP probability. It defers the decision until the end of the utterance. We can have the decoder fully integrated with feature selection procedure. Namely, search space is augmented with an additional dimension. Consequently, all features are considered in decoding and the optimal feature selection path based on the MAP criteria can be achieved.

The experiments were conducted on a large vocabulary continuous speech recognition task. The data set is a speaker-dependent database with a male and female speaker. We used three feature streams: MFCC, a modified PLP [5], and an auditory feature [6].

The experimental results showed that multiple-feature decoding could improve the performance significantly.

This paper is organized as follows. First, we describe our multiple-feature decoding framework. Then we discuss several implementation issues. Following that we discuss our system setup and the data we used in our experiments before we present our experimental results. Finally we summarize our results and compare it with the frame-level score combination and ROVER [7].

2. THE UNIFIED FRAMEWORK

In the general unified framework, we can have a massive feature matrix that covers the whole acoustic space with quantized resolution. For different context-sensitive acoustic units, a mixture of feature representations can be used. The mixture weights can be a function of acoustic units and their contexts. Optimization criteria such as ML or MCE can be used to derive these weights.

Here we are particularly interested in run-time feature selection that is a special case of the general framework. Figure 1 illustrates such a special case, which we refer as multiple-feature decoding. Assume we use three different feature streams, the optimal path (shadowed area) for word *was* indicates that feature 3 should be used for phone /w/, feature 1 used for phone /ao/ and feature 2 used for phone /z/.

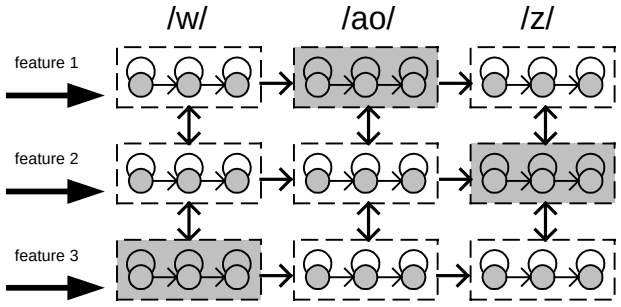

Figure 1. An example of multiple feature decoding

In speech recognition, we need to find the optimal word sequence W^* that maximize the probability of the word sequence given the acoustic observation *A*. Namely,

$$W^* = \arg\max_W \; p(W|A) \tag{1}$$

To simplify discussion without losing generality, we assume there are multiple feature streams $\{F_i\}$, $i \in [1,\ldots N]$ associated with *A*, and there are *K* phonetic or word segments for the utterance we are considering. These segments have accurate segmentation already. We can rewrite Equation (1) as follows:

$$\begin{aligned} W^* &= \arg\max_W \sum_{I\in\Psi}\prod_{k=1}^{K} p(W^k|F_I^k, A)p(F_I^k|A) \\ &\cong \arg\max_W \sum_{I\in\Psi}\prod_{k=1}^{K} p(W^k|F_I^k)p(F_I^k|A) \end{aligned} \tag{2}$$

where Ψ is a set that contains all the possible feature stream permutation paths from the 1st to the *K*th segment, $I=(I_1, I_2 \ldots I_k), I_k \in [1,\ldots N]$, maps to one of the feature streams $\{F_i\}$ in the *k*th segment, F_I^k denotes the acoustic feature stream in the *k*th segment, W^k denotes the corresponding word or phonetic model in the *k*th segment, and $P(F_I^k|A)$ is the probability of a particular feature stream in segment *k* given the massive feature matrix. Equation (2) implies that the best word sequence should be the one that has the highest overall posterior probability for all the possible feature stream combinations for the given utterance. To identify the optimal word sequence, this is a search problem like conventional speech recognition decoding. Since we have no phonetic segment information as indicated in Equation (2), we need to search for all the possible feature streams and phonetic or word segments.

One way to approximate Equation (2) is to find the maximum term based on the feature stream for the whole utterance as follows:

$$\begin{aligned} W^* &= \arg\max_W \{\max_I \prod_k p(W|F_I^k)p(F_I^k|A)\} \\ &= \arg\max_W \{\prod_k \max_{i\in[1\ldots N]} p(W|F_i^k)p(F_i^k|A)\} \end{aligned} \tag{3}$$

This is equivalent to finding the highest probability $p(W|F_i^k)p(F_i^k|A)$ for each segment and to let feature switching across different segment in the same fashion as the beam search algorithm. In general, we have:

$$\begin{aligned} &\sum_{I\in\Psi}\prod_k p(W^k|F_I^k)p(F_I^k|A) \geq \\ &\prod_k \max_{i\in[1\ldots N]} p(W|F_i^k)p(F_i^k|A) \geq \\ &\max_{i\in\text{feature stream}} \prod_k p(W^k|F_i^k)p(F_i^k|A) \end{aligned} \tag{4}$$

Equation (4) provides a theoretic justification for unified decoding and feature representation. We can search the best feature stream in addition to search for the best word sequence. This guarantees that we have a better evaluation function than the one that evaluates each feature stream independently to select the best results, which is equivalent to running decoder with each feature stream and selecting the best one that has the highest probability $p(W|F_i)p(F_i|A)$ for the entire utterance. It is possible that for various segments $p(W|F_i^k)p(F_i^k|A)$ could be very different on different feature streams so the approximation can be significantly underestimated.

In practice, $P(W|F_i)$ can be rewritten in terms of acoustic model probability $P(F_i|W)$ and language model probability $P(W)$ as follows:

$$p(W|F_i) = \frac{p(F_i|W) * p(W)}{p(F_i)} \tag{5}$$

When only a single feature stream is used, $p(F_i)$ is usually ignored, which is the case for most of the speech recognition systems. However, for multiple feature streams, $p(F_i)$ needs to be computed as part of multiple feature decoding. $p(F_i)$ can be regarded as a normalization factor across feature streams. Without it, the scores generated by different feature streams often have different dynamic range and they are not directly comparable.

3. IMPLEMENTATION ISSUES

We discuss a number of implementation issues to modify the conventional decoder.

3.1 Computation of $p(F_i)$

We can express $p(F_i)$ as:

$$p(F_i) = \sum_w p(F_i|W)\,p(W) \quad (6)$$

This means that we can approximate the true $p(F_i)$ by summing over all the active hypotheses in feature *i* during the decoding process. This can be updated for each frame based on partial decoding history.

3.2 Computation of $P(F_i|A)$

$P(F_i|A)$ can be approximated based on the overall feature fitness or VQ distortion of each feature streams for the given test data. A simple assumption to make here is that they have a uniform distribution. It is possible that we can use speech recognition accuracy of each feature stream and other related confidence measures as the weight to balance different feature streams. In fact, Kirchhoff and Bilmes used confidence measures for combination [4].

3.3 Search Pruning

Different feature streams often behave differently during different search stages. Therefore, pruning thresholds and different pruning states need to be introduced. We can allow different beam width to be set for different feature streams and also keep track of their pruning state variables separately.

3.4 Feature Switch Penalty

A hypothesis can have different segment using different features. The switch between feature streams can happen at different level such as acoustic frame level, senone level, phone level, word level and utterance level. A switch penalty can be used. The switch penalties at each level can also be different.

3.5 Book-Keeping

When different hypotheses with different feature stream generate the same word hypothesis, we need to keep track of their scores separately so we can apply switch penalties later on.

4. EXPERIMENTAL SETUP

The experiments were conducted on a large-vocabulary continuous speech recognition (CSR) task. The data used is collected in house for one male speaker (Scott) and one female speaker (Melanie). There were 6030 training utterances for Scott and 6620 training utterance for Melanie. 345 and 387 test utterances were used for Scott and Melanie respectively. The transcriptions of training and test data were extracted from general English reading materials and they did not match the North American Business (NAB) language model we used very well. Therefore the language model perplexity on test data was fairly high.

Microsoft's Whisper speech recognizer [8] was used as the baseline in our experiments. It processes 16kHz PCM data using a MEL-scale cepstrum along with its dynamics into a multi-dimensional feature vector. The acoustic model we used here is a reduced version – a set of HMMs with continuous-density output probabilities consisting of 3000 senones. A mixture of 4 Gaussian densities with diagonal covariances was used for each senone. The phonetic modeling in the system consists of position and context dependent within-word and crossword triphones. A more complete description of the Whisper speech recognition system can be found in [8].

5. EXPERIMENTAL RESULTS

We used three feature streams. First one is the MFCC that is widely used by modern speech recognition systems. The second one is a modified Perceptual Linear Prediction (PLP) feature [5]. It uses Mel-scale filter-bank instead of the standard PLP filter-bank. It has been shown that the modified PLP performs differently from MFCC, especially for mismatched conditions [5]. The third one is an auditory representation based on the auditory model [6]. Here we used a slightly modified version with Har wavelets.

Table 1 shows the baseline results using both the MFCC and PLP feature as a signal feature for Whisper. Modified PLP was similar to the baseline MFCC system but the auditory feature performs much worse than the MFCC baseline.

BASELINE	SCOTT	MELANIE	BOTH
MFCC	6.57%	4.62%	5.62%
MPLP	6.38%	4.94%	5.68%
AUDI	7.23%	5.56%	6.42%

Table 1. Baseline results using three features.

Using the conventional frame-level score combination method, we experimented both the product and sum rule in combination with equal weights. We found that the product rule works slightly better than the sum rule, which is consistent with reports in [3,4]. Table 2 shows the error rate with different feature combination at acoustic frame level. The best result was achieved by using MFCC and MPLP, which provided a marginal 3% error reduction. The AUDI feature did not provide much help since the performance by itself was considerably worse than the other two features. Therefore, for best results using frame-level score combination, all features should have a comparable performance.

COMBINATION	SCOTT	MELANIE	BOTH
MFCC+AUDI	7.04%	4.65%	5.87%
MFCC+MPLP	6.51%	4.35%	5.46%
MFCC+AUDI+MPLP	6.91%	4.09%	5.54%

Table 2. Results using conventional frame-level score combination with the product rule.

In our current multiple-feature decoding implementation, $P(F_i|A)$ is ignored. We used $p(W|F_i)$ only to determine the best feature stream for each decision boundary. Feature switching is also enabled at both phone and word boundary without any switching penalty. The experimental the results are shown in Table 3.

DECODING	SCOTT	MELANIE	BOTH
MFCC+AUDI	5.98%	4.03%	5.03%
MFCC+MPLP	6.29%	4.16%	5.25%
MFCC+AUDI+MPLP	5.64%	3.93%	4.80%

Table 3. Multiple-feature decoding results.

From Table 3 we can see that feature selection not only outperformed the simple score combination method in every case, but also provided significant performance improvement over the baseline. In this case, using all three features, the error was reduced by 15% over the Whisper baseline when a single feature

is used. To make sure that the performance improvement is not coming from the increase on number of parameters, we ran the baseline with MFCC feature and 8 and 12 Gaussian mixtures per senone, which had the same number of parameters with 2 and 3 feature streams respectively. With 8 Gaussian mixtures per senone, the error rate on both speakers was 5.47%, only marginally better than 4 Gaussian mixture baseline. With 12 Gaussians mixtures per senone, the error rate was increased to 6.83%, which shows that the model was undertrained due to insufficient amount of data over the parameter size. Also it is interesting to observe that MFCC+AUDI was performing better than MFCC+MPLP even though AUDI by itself was not as good as MPLP. It is possible that that MFCC and AUDI are more complimentary. It also indicates that multiple feature decoding is more robust than frame-level score combination.

6. COMPARISON WITH ROVER

ROVER [7] is a straightforward approach to combine results from different speech recognition systems. Yet it is effective and simple to significantly improve system performance with a number of different systems for the broadcasting news transcriptions task. It has significantly reduced the word error rate with combination of a few comparable but different systems [9]. It is obvious that ROVER can be applied to multiple feature recognition too. We run the experiment with pure-voting based ROVER (without confidence score) and the results are listed in Table 4. The error reduction was about 5% over the best single feature performance.

ROVER	SCOTT	MELANIE	BOTH
MFCC+AUDI+MPLP	6.20%	4.48%	5.36%

Table 4. ROVER results using three features

In comparison to ROVER results, the proposed method also performs significantly better. Once again, this is probably because that multiple feature decoding is a better MAP approximate than ROVER. In addition to better granularity, it enforces the time alignment at the boundaries of each segment with different feature streams. ROVER only uses dynamic programming to match transcriptions for voting purpose. There is no time alignment information in the decision process.

7. EXTENSION OF THE FRAMEWORK

Although we described the general framework in the context of multiple feature decoding, there is no reason why the framework cannot be extended to other tasks. As an example, we discuss two possible scenarios in which the framework can be applied.

The first scenario would be to use it to combine results from multiple systems just like ROVER. Assume there are multiple system configurations; all of them have different feature extraction, different acoustic models and different language models. We can put all the systems in the multiple-system decoding framework and let the decoder choose which system to use for each word in the utterance. It would be a good alternative to ROVER since the combination is based on posterior probability and it will force the time alignment on word boundaries between the words generated by different systems.

The second scenario would be to work with speaker-clustered models. For true speaker-independence, we might need more than one set of acoustic models. One possible solution would be to generate speaker-clustered models. Since the test speaker is unseen, it may not match any of the clustered models perfectly. However, it is quite possible that some of the acoustic units would match a particular clustered model better than the rest. The multiple-model decoding would allow the optimal path to be selected resulting in optimal models being used for each segment.

8. SUMMARY

In this paper we introduced a general framework that links decoding and feature representations for speech recognition based on the MAP criterion. Experiments showed that it is very effective and outperformed other combination schemes such as ROVER. The most critical factor to make such a scheme work is to have a set of multiple features that has the characteristics of consistency and complementariness. By consistency we mean that each feature should provide us with a comparable recognition accuracy. By complementariness we mean that these features should not be correlated and they can be complementary to each other.

9. ACKNOWLEDGMENT

The authors would like to thank Dr. Kuansan Wang for providing the auditory features used in this paper.

REFERENCES

[1] X. Huang, M. Belin, F. Alleva and M. Hwang, "Unified Stochastic Engine (USE) for Speech Recognition", Proc. of ICASSP-93, pp 636-639, 1993

[2] B.E.D. Kinsbury and N. Morgan, "Recognizing Reverberant Speech with RASTA-PLP", Proc. of ICASSP-97, pp 1259-1262, 1997

[3] A. K. Hallberstadt and J. R. Glass, "Heterogeneous Measurements and Multiple Classifiers for Speech Recognition", Proc. of ICSLP-98, 1998

[4] K. Kirchhoff and J. A. Bilmes, "Dynamic Classifier Combination in Hybrid Speech Recognition Systems Using Utterance-Level Confidence Values", Proc. of ICASSP-99, 1999

[5] P.C. Woodland, M.J.F. Gales and S.J. Young, "The Development of the 1996 Broadcast News Transcription System", Proc. DARPA Speech Recognition Workshop, pp. 73-78, Chantilly, Virginia

[6] K.S. Wang and S. Shamma, "Self-Normalization and Noise-Robustness in Early Auditory Representations", IEEE Transactions on Speech and Audio Processing, Vol 2, No. 3, July 1994, pp 421-435

[7] J. G. Fiscus, "A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER)", IEEE Workshop on Automatic Speech Recognition and Understanding, 1997

[8] X. Huang, A. Acero, F. Alleva, M.Y. Hwang, L. Jiang and M. Mahajan, "Microsoft Windows Highly Intelligent Speech Recognizer: Whisper". Proc. of ICASSP-95, Detroit, May 1995.

[9] S. S. Chen, E. M. Eide, M. J. F. Gales, R. A. Gopinath, D. Kanevsky and P. Olsen, "Recent Improvements to IBM's Speech Recognition System for Automatic Transcription of Broadcast News", Proc. of ICASSP-99, 1999