Discriminative Reranking for LVCSR Leveraging Invariant Structure

Masayuki SUZUKI[1], *Gakuto KURATA*[2], *Masafumi NISHIMURA*[2], *Nobuaki MINEMATSU*[1]

[1]The University of Tokyo, Tokyo, Japan
[2]IBM Research - Tokyo, Kanagawa, Japan
{suzuki, mine}@gavo.t.u-tokyo.ac.jp, {gakuto, nisimura}@jp.ibm.com

Abstract

An invariant structure is one of the long-span acoustic representations, where acoustic variations caused by non-linguistic factors are effectively removed from speech. We present in this paper a new method to leverage the invariant structures as features of discriminative reranking for Large Vocabulary Continuous Speech Recognition (LVCSR). First we use a traditional HMM-based LVCSR system to get a list of N-best candidates with phone alignments and construct an invariant structure for each candidate using its phone alignment. Here, the invariant structure is composed of lengths between every two phonemes in the candidate. Then we estimate a score of each phoneme-pair in the invariant structure, and rerank the N-best candidates using a weighted sum of the phoneme-pair scores, where the weights are trained discriminatively by averaged perceptron. Experimental results show a relative CER improvement of 6.69% over the baseline HMM-based LVCSR system.

Index Terms: Invariant Structure, LVCSR, Discriminative reranking

1. Introduction

Discriminative reranking provides an additional gain to a baseline system for some kinds of tasks, such as syntactic parsing [1], machine translation [2], LVCSR [3], and so on. One reason of the gain is the ease with which many arbitrary features that are intractable within the baseline system can be integrated into the reranking model. Here, the selection of features play an important role for the performance improvement [4]. As for LVCSR, various language features (e.g. word n-gram counts) have been explored within the discriminative reranking paradigm, and it is called Discriminative Language Model (DLM) [5, 6]. More recently, some kinds of acoustic features such as duration n-gram have been investigated [7, 8].

We propose in this paper a method to leverage an *invariant structure* as a feature of discriminative reranking for LVCSR. The invariant structure is proposed by Minematsu [9], where the acoustic variations caused by non-linguistic factors are effectively removed from speech. The invariant structure is composed of lengths between every two phonemes in the utterance so that it possesses long-span acoustic contrast information. Since the invariant structure is made up of non-local acoustic features, it is difficult to use invariant structures directly for Hidden Markov Model- (HMM-) based LVCSR. On the other hand, discriminative reranking approach can leverage not only local features but also non-local features including the invariant structure, and it might be useful to improve the performance. Actually, invariant structures have already been used in N-best reranking for continuous digits recognition, and an experimental result showed that a simple weighted sum of an HMM-based score and a structure-based score improved the performance [10]. However, the invariant structure has not been applied to LVCSR yet. In addition, no discriminative approach has been done yet to calculate the structure-based score. In this paper, we leverage invariant structures to improve discriminative reranking for LVCSR. We calculate a score for each phoneme-pair in a candidate generated by an HMM-based LVCSR system. This score reflects how appropriate the length between the phoneme pair is. Then we use these scores as features for discriminative reranking model.

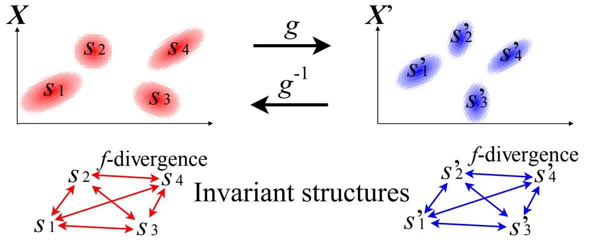

Figure 1: Invariant structures. This structures are composed of 4 acoustic events (phonemes) so that they have $4 \times (4-1)/2 = 6$ edges. Each edge length (f-divergence) of the invariant structures is invariant to any invertible transformation g and g^{-1}.

2. Related works

2.1. Invariant structure

Voices of two speakers show different timbre because they have different vocal tract lengths and shapes. In studies of speaker adaptation and voice conversion, speaker difference is often modeled mathematically as an invertible transformation in the cepstrum domain. This fact indicates that if we can find any transform-invariant features, they will be robust features.

A necessary and sufficient condition for a feature to be invariant with any continuous and convertible transform is that the feature is f-divergence. f-divergence is a family of divergences and the well-known Bhattacharyya distance is a kind of f-divergences. Consider a feature space $\boldsymbol{X}$ and M events $\{s_i\}_{i=1}^{M}$ (e.g. phonemes) in $\boldsymbol{X}$. Each event is described as a distribution $s_i(\boldsymbol{x})$ in the feature space. Assume there is an invertible transformation $g : \boldsymbol{X} \rightarrow \boldsymbol{X}'$ which transforms $\boldsymbol{X}$ into a new feature space $\boldsymbol{X}'$. In this way, M events $\{s_i\}_{i=1}^{M}$ in $\boldsymbol{X}$ is mapped to $\{s'_i\}_{i=1}^{M}$ in $\boldsymbol{X}'$. Here, f-divergence between two distributions s_j and s_k $(1 \leq j < k \leq M)$ is invariant with any kind of arbitrary invertible transform g. Therefore, it is equal to f-divergence between s'_j and s'_k.

Fig. 1 shows two invariant structures composed only of f-divergences. With multiple events, we can obtain a structure by calculating f-divergences between any pair of them. Since

```
Input: Training samples $(x_i, \overline{y_i}, \underline{y_i})$ for $i = 1 \dots I$
Initialization: $\boldsymbol{\alpha}_0^I = \mathbf{0}$
for $t = 1 \dots T$ do
    $\boldsymbol{\alpha}_t^0 = \boldsymbol{\alpha}_{t-1}^I$
    for $i = 1 \dots I$ do
        if $\boldsymbol{\alpha}_t^{i-1} \cdot \mathbf{\Phi}(x_i, \overline{y_i}) + \phi_0(x_i, \overline{y_i})$
                $> \boldsymbol{\alpha}_t^{i-1} \cdot \mathbf{\Phi}(x_i, \underline{y_i}) + \phi_0(x_i, \underline{y_i})$ then
            $\boldsymbol{\alpha}_t^i = \boldsymbol{\alpha}_t^{i-1} + \lambda \left( \mathbf{\Phi}(x_i, \underline{y_i}) - \mathbf{\Phi}(x_i, \overline{y_i}) \right)$
Output: $\boldsymbol{\alpha} = \Sigma_{i,t} \boldsymbol{\alpha}_i^t / IT$
```

Figure 2: A variant of averaged perceptron algorithm. $\overline{y_n}$ and $\underline{y_n}$ shows the highest-WER candidates and the lowest-WER candidates in N-best candidates of x_n, respectively. I is the number of training data. T is the number of iterative training. λ is a parameter of learning rate, which is fixed to a constant value in our experiment.

f-divergence is invariant to any invertible transformation, the obtained structure is robust to speaker difference and any other distortions which can be expressed by an invertible transformation of the feature space (e.g. microphone difference).

2.2. Discriminative reranking for LVCSR

Discriminative reranking for LVCSR takes N-best candidates obtained by a baseline LVCSR system as input, and reranks these candidates based on a set of features. Each candidate is mapped to a d-dimensional feature vector $\mathbf{\Phi}(x, y)$, which is an arbitrary function of acoustic input x and its candidate y. For example, the number of word “foo” or “bar” in the candidates can be a feature. We can realize this by setting the feature vector:

$$\mathbf{\Phi}(x, y) = \begin{bmatrix} \text{the number of a word “foo” in } y \\ \text{the number of a word “bar” in } y \\ \vdots \end{bmatrix}. \qquad (1)$$

Each candidate is also mapped to a scalar parameter $\phi_0(x, y)$, which is a likelihood score obtained by the baseline system.

Then, a d-dimensional parameter vector $\boldsymbol{\alpha}$ associated with the feature vector $\mathbf{\Phi}(x, y)$ is learned discriminatively. $\boldsymbol{\alpha}$ is interpreted as degree of importance for each feature to improve performance. The best candidates y^* under the reranking model with $\boldsymbol{\alpha}$ is obtained through

$$y^* = \operatorname*{argmax}_{y \in \mathrm{NBEST}(x)} \boldsymbol{\alpha} \cdot \mathbf{\Phi}(x, y) + \phi_0(x, y), \qquad (2)$$

where NBEST(x) is all N-best candidates for the acoustic input x.

For training of $\boldsymbol{\alpha}$, we used in this paper a variant of the perceptron algorithm (see Fig. 2). The main idea of this algorithm is to penalize the features associated with the highest-WER candidates $\overline{y}$, and to reward the features associated with the lowest-WER candidates $\underline{y}$ in N-best candidates [14]. Averaged parameter $\boldsymbol{\alpha} = \Sigma_{i,t} \boldsymbol{\alpha}_i^t / IT$ gives correct prediction with a large margin so that it has higher generalization ability [15].

3. Proposed method

We propose in this paper a method to leverage the invariant structure to improve the discriminative reranking for LVCSR. The framework of our proposed method is shown in Fig. 3. Note that the numbers in Fig. 3 correspond to those of the following subsections. Here, two modules of “HMM-based ASR” and “Extract an invariant structure” are the same as those in [10]. The originality of this paper lies in “Calculate structure score” and “Reranking”.

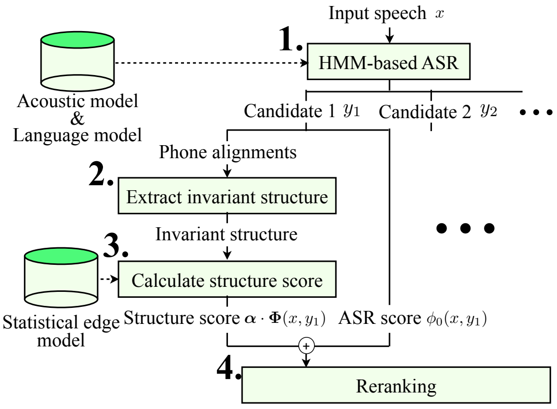
Input speech x
1.
HMM-based ASR
Acoustic model & Language model
Candidate 1 y_1
Candidate 2 y_2
• • •
Phone alignments
2.
Extract invariant structure
Invariant structure
3.
Calculate structure score
• • •
Statistical edge model
Structure score $\boldsymbol{\alpha} \cdot \mathbf{\Phi}(x, y_1)$
ASR score $\phi_0(x, y_1)$
4.
Reranking

Figure 3: Framework of discriminative reranking for LVCSR leveraging invariant structure. The numbers in this figure correspond to those of the subsections of Section 3.

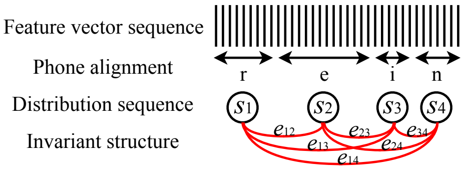

Figure 4: A procedure of extracting an invariant structure from a phone alignment. A hypothesized word is [r e i n].

3.1. HMM-based ASR

We use a traditional HMM-based Automatic Speech Recognition (ASR) system to get N-best candidates. We can also get the log likelihood of the system (we use it as $\phi_0(x, y)$) and the phone alignment for each candidate.

3.2. Extract an invariant structure

We extract the invariant structure for each N-best candidates using the phone alignment. Fig. 4 shows a procedure of extracting the invariant structure from a feature vector sequence and its phone alignment. First we estimate a distribution for each phoneme from the feature vector sequences aligned with this phoneme. Then we extract the invariant structure by calculating f-divergence between each pair of distributions. We denote each edge as $\{e_{ij}\}$, where $1 \le i < j \le M$ and M is the number of phonemes in the candidate.

3.3. Calculate structure score

We use properness of each edge of the invariant structure as feature for discriminative reranking. Here, we use a log likelihood score of each edge as properness and the score was calculated by using Statistical Edge Models (SEM). The SEMs are trained with training samples of phoneme-pair lengths.

The left side of Fig. 5 shows a process of building SEMs from the training data. We make an SEM for each pair of phonemes. Using edge length (f-divergence) of each phoneme-

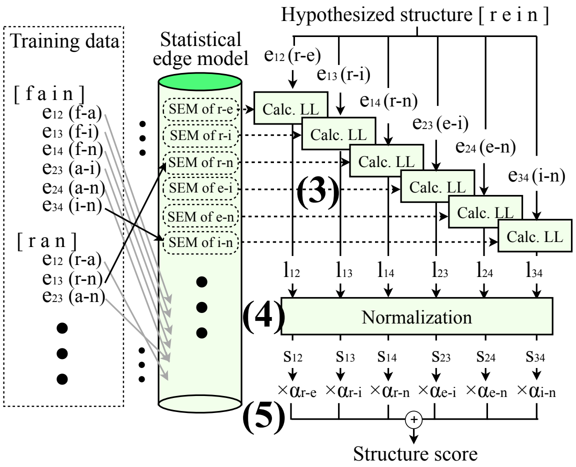

Figure 5: Process for building statistical edge models (SEMs) and process for calculating a structure score. Log likelihood is abbreviated as LL. (3), (4), and (5) in this figure correspond to those of equations in this paper.

pair in the training data, we train an SEM of the phoneme-pair as a K-mixture Gaussian Mixture Model (GMM). Suppose there are P phoneme IDs, we make $\frac{P(P-1)}{2}$ SEMs.

The right side of Fig. 5 depicts how to calculate a structure score for a candidate. Note that the numbers in Fig. 5 correspond to those of the following equations. First, we calculate a log likelihood score of e_{ij} by

$$l_{ij} = \log \sum_{k=1}^{K} w_{p_{ij}}^{k} \mathcal{N}(e_{ij}; \mu_{p_{ij}}^{k}, \sigma_{p_{ij}}^{k}), \tag{3}$$

where p_{ij} is Phoneme-Pair ID (PPID) of e_{ij}. And $w_{p_{ij}}^{k}$, $\mu_{p_{ij}}^{k}$, $\sigma_{p_{ij}}^{k}$ are weight, mean, variance of k-th component of GMM (SEM) for p_{ij}, respectively.

Then we normalize the log likelihood scores to fairly compare candidates which have the different number and different length of phonemes. Normalized structure scores are given by

$$s_{ij} = \frac{(f_i + f_j)}{M - 1} l_{ij}, \tag{4}$$

where f_i, f_j are the number of frames of i-th and j-th phonemes in the candidate, respectively. And f_i, f_j are easily calculated from phoneme alignment.

Finally, we calculate a structure score as:

$$\text{Structure-score} = \boldsymbol{\alpha} \cdot \boldsymbol{\Phi}(x, y), \tag{5}$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\Phi}$ are $\frac{P(P-1)}{2}$ dimensional vectors whose elements are corresponding to each PPID. $\boldsymbol{\Phi}$ represents a sum of normalized edge scores $\{s_{ij}\}$ for each PPID:

$$\boldsymbol{\Phi}(x, y) = \begin{bmatrix} \sum_{i<j}^{M} s_{ij} \text{ if } p_{ij} = 1 \text{ , otherwise } 0 \\ \sum_{i<j}^{M} s_{ij} \text{ if } p_{ij} = 2 \text{ , otherwise } 0 \\ \vdots \\ \sum_{i<j}^{M} s_{ij} \text{ if } p_{ij} = \frac{P(P-1)}{2} \text{ , otherwise } 0 \end{bmatrix}. \tag{6}$$

If a specific PPID is not observed in the candidate, its element of the feature vector will be zero. $\boldsymbol{\alpha}$ is trained by the algorithm of Fig. 2 so that it can be interpreted as degree of importance for each PPID to reduce WER.

Table 1: Experimental condition for Japanese continuous digits recognition.

Utterances	1 to 11 continuous Japanese digits
Training data of HMM	27.5 hours / 667 spks / 17316 utters
Training data of SEM	27.5 hours / 667 spks / 17316 utters
Training data of $\boldsymbol{\alpha}$	5.0 hours / 520 spks / 3977 utters
Test data	1.5 hours / 100 spks / 7382 utters
# of HMM states	500
# of HMM Gaussians	15000
# of monophones (P)	18
# of monophone-pair	136
Language model	Unigram that outputs 10 digits (0 to 9) and the end of sentence symbol with equal probabilities
Baseline WER	1.09% (S=67, I=140, D=14 / 20303)*
10-best oracle	0.75% (S=59, I=85, D=9 / 20303)*

* S: # of substitutions, I: # of insertions, D: # of deletions.

Table 2: Experimental condition for Japanese LVCSR

Utterances	Japanese reading utterances
Training data of HMM	352 hours / 1325 spks / 196475 utters
Training data of SEM	24 hours / 100 spks / 13112 utters
Training data of $\boldsymbol{\alpha}$	30 hours / 164 spks / 16733 utters
Test data	1.5 hours / 20 spks / 600 utters
# of HMM states	5000
# of HMM Gaussians	150000
# of monophones (P)	57
# of monophone-pair	1596
Language model	Word 2-gram estimated with modified Kneser-Ney smoothing[12]
# of words	104262
Baseline CER	3.59% (S=422, I=56, D=64 / 15096)*
10-best oracle	1.32% (S=161, I=15, D=24 / 15096)*

* S: # of substitutions, I: # of insertions, D: # of deletions.

3.4. Reranking

We rerank the the N-best candidates by combining the structure score $\boldsymbol{\alpha} \cdot \boldsymbol{\Phi}(x, y)$ and the ASR score $\phi_0(x, y)$. The reranked result is given by (2). If $\boldsymbol{\alpha} = \mathbf{0}$, the score of discriminative reranking model becomes $\phi_0(x, y)$ and the result is exactly the same as that of the HMM-based baseline system. The proposed reranking model has the potential to improve the performance because an invariant structure expresses the contrast between phonemes of an input utterance that the HMM-based system doesn't take into consideration well.

4. Experiments

4.1. Experimental setup

We conducted two experiments: Japanese continuous digits recognition and Japanese LVCSR. Table 1 and Table 2 show the experimental conditions of them, respectively. We used our conventional HMM-based ASR system to generate 10-best candidates with phone alignments [13]. We trained acoustic models (AM) of phonemes, and the HMM states were clustered by using a phonetic decision tree.

Distributions of the phonemes to form a invariant structure were estimated from 13-dimensional PLP feature sequences which were aligned to the middle state of the corresponding HMM. We assumed that the distribution is a Gaussian and the mean of the Gaussian was estimated in a maximum likelihood

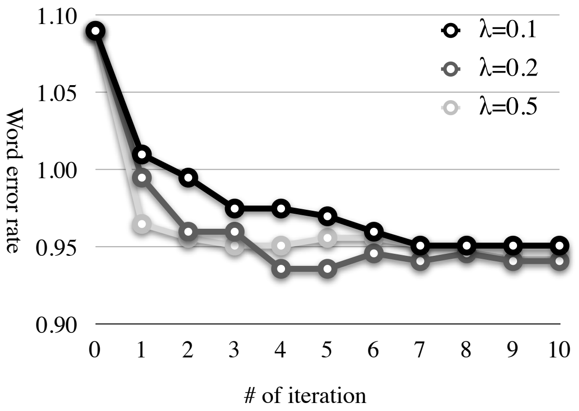

Figure 6: Word error rate for Japanese connected digits recognition.

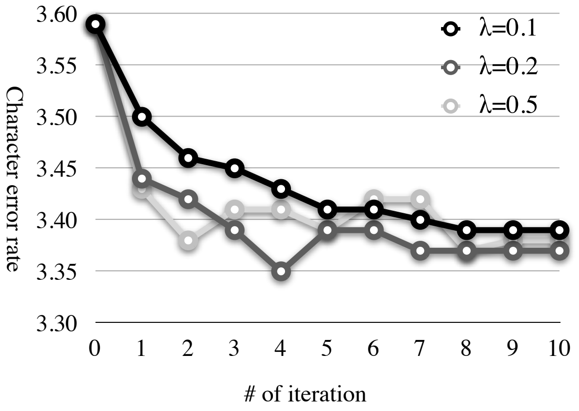

Figure 7: Character error rate for Japanese LVCSR.

manner. For variance, we used the common variance for each monophone [10]. We used the square root of Bhattacharyya distance as f-divergence to calculate edge length. The number of mixtures of Gaussians for SEM was set to 16.

4.2. Results

Fig. 6 and Fig. 7 show the Word Error Rate (WER) and the Character Error Rate (CER) for connected digits recognition and LVCSR, respectively. Since Japanese word segmentation is ambiguous, we used the CER instead of the WER for the evaluation of LVCSR. We used the lowest-CER candidates and the highest-CER ones to train averaged perceptron for LVCSR task, so that it was trained to reduce CER. The horizontal axis shows the number of iterative training of the averaged perceptron T. WER at $T = 0$ presents the result by the baseline system. λ is a learning rate for the average perceptron algorithm. The proposed method outperformed the baseline in any case of λ and T. As for connected digits recognition, when $\lambda = 0.2$ and $T = 4$ or 5, the lowest WER 0.94%, relative 14.1% improvement from the baseline WER 1.09%, was achieved. In this condition, all of the number of substitutions, insertions, and deletions are reduced from 67, 140, and 14 to 64, 113, and 13, respectively. As for LVCSR, when $\lambda = 0.2$ and $T = 4$, the lowest CER 3.35%, relative 6.69% improvement from the baseline CER 3.59%, was achieved. In this condition, all of the number of substitutions, insertions, and deletions are reduced from 422, 56, and 64 to 401, 44, and 61, respectively.

5. Conclusion

We propose in this paper a discriminative reranking for LVCSR leveraging an invariant structure. The proposed method is the first trial to apply an invariant structure to an LVCSR task. Experimental results show that a relative CER improvement of 6.69% over our baseline LVCSR system was achieved.

Our future work includes feature engineering of discriminative reranking. There are many useful features for discriminative reranking such as word n-gram counts. Because linguistic features like n-gram counts and edge-based scores offer different kinds of information, simultaneous use of both features in discriminative reranking might improve the performance even more.

6. References

[1] M.Collins, "Discriminative reranking for natural language parsing," *Proc. ICML*, pp.175–182, 2000.

[2] L. Shen, *et.al.*, "Discriminative reranking for machine translation," *Proc. HLT-NAACL*, pp.177–184, 2004.

[3] B. Roark, *et.al.*, "Discriminative n-gram language modeling," *Computer Speech and Language*, vol. 21, no. 2, pp.373–392, 2007.

[4] M. J. F. Gales *et.al.*, "Structured discriminative models for speech recognition," *IEEE Trans.* (submitted) http://mi.eng.cam.ac.uk/∼mjfg/segdisc_2012.pdf

[5] E. Arisoy, *et.al.*, "Feature combination approaches for discriminative language models," *Proc. Interspeech*, pp.617–620, 2011.

[6] T. Oba, *et.al.*, "Round-robin duel discriminative language models in one-pass decoding with on-the-fly error correction," *Proc ICASSP*, pp. 5588–5591, 2011.

[7] M. Lehr and I. Shafran, "Discriminatively estimated joint acoustic, duration and language model for speech recognition," *Proc. ICASSP*, pp.5542–5545, 2010.

[8] G. Zweig, *et.al.*, "Speech recognition with segmental conditional random fields: a summary of the JHU CLSP 2010 summer workshop," *Proc. ICASSP*, pp.5044–5047, 2011.

[9] N. Minematsu, "Yet another acoustic representation of speech sounds," *Proc. ICASSP*, pp.585–588, 2004.

[10] M. Suzuki, *et.al.*, "Continuous digits recognition leveraging invariant structure," *Proc. INTERSPEECH,* pp.993–996, 2011.

[11] Y. Qiao and N. Minematsu "A study on invariance of f-divergence and its application to speech recognition," *IEEE Trans. on Signal Processing*, vol 58, no.7, pp.3884–3890, 2010.

[12] S. Chen, *et.al.*, "An empirical study of smoothing techniques for language modeling", *Computer Speech & Language,* vol. 13, no. 4, pp. 359–393, 1999.

[13] S. Chen, *et.al.*, "Advances in speech transcription at IBM under the DARPA EARS program," *IEEE Trans. on Speech and Audio Processing,* vol. 14, no. 5, pp.1596-1608, 2006.

[14] T. Oba, *et.al.*, "Efficient discriminative training of error corrective models using high-WER competitors," *IEICE Technical Report,* SP2007-185-214, pp. 99–104, 2008.

[15] K. Crammer, *et.al.*, "Scalable large-margin online learning for structured classification," *NIPS Workshop on Learning With Structured Outputs,* 2005.