A Study on Lattice Rescoring with Knowledge Scores for Automatic Speech Recognition

Sabato Marco Siniscalchi, Jinyu Li, and Chin-Hui Lee

School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0250, USA
{marco, jinyuli, chl}@ece.gatech.edu

Abstract

We study lattice rescoring with knowledge scores for automatic speech recognition. Frame-based log likelihood ratio is adopted as a score measure of the goodness-of-fit between a speech segment and the knowledge sources. We evaluate our approach in two different applications: phone recognition, and connected digit continuous recognition. By incorporating knowledge scores obtained from 15 attribute detectors for place and manner of articulation, we reduced phone error rate from 40.52% to 35.16% using monophone models. The error rate can be further reduced to 33.42% for triphone models. The same lattice rescoring algorithm is extended to connected digit recognition using the TIDIGITS database, and without using any digit-specific training data. We observed the digit error rate can be effectively reduced to 4.03% from 4.54% which was obtained with the conventional Viterbi decoding algorithm with no knowledge scores.

Index Terms: detection-based automatic speech recognition, lattice rescoring, domain-independent speech recognition.

1. Introduction

A state-of-the-art automatic speech recognition (ASR) system is often designed using data-driven methods, such as hidden Markov model (HMM) [1]. Its performance is usually improved by collecting more and more training data. The integration of additional knowledge sources is considered to be beneficial to ASR robustness [2], [3], [4], and [5]. An Automatic Speech Attribute Transcription (ASAT) paradigm for speech recognition has recently been proposed as a new way for integrating knowledge sources into HMM-based systems [2]. This framework is based upon detection of low level speech events, and integration of knowledge sources into ASR is accomplished by extracting knowledge-based front-end features, e.g. manner and place of articulation. As a first attempt, frame-based event detectors were realized with feed forward artificial neural networks (ANNs) [6]. The output of the ANNs represents the knowledge scores, and they were used to rescore phone-based *n*-best candidate lists provided by conventional HMM-based systems. A problem with the ANN-based scores is that they are likely to fluctuate [7], resulting in extra detected speech segments. This work extends the *n*-best phone lists rescoring algorithm and frame-based ANN knowledge scores in [6], and herein we address the two issues by using segment-based detectors and embedding alternative hypotheses into lattice structures. In addition, we validate the generality of the detector-based approach by performing word lattice rescoring.

We implement the HMM-based segment detectors as in [7], and propose knowledge scores based upon log likelihood ratio (LLR). Those scores can be computed at a segment, or frame level. We observed at the frame level a better separation between competing models. Phone-level scores are then obtained as a non-linear mapping of LLR scores into the phone space. These phone scores are used to rescore lattices of alternative hypotheses, which give more detailed information than *n*-best lists.

We evaluate our approach on two different task. The first is a continuous phone recognition using the TIMIT database [8]. By performing knowledge based rescoring we achieve, in terms of phone error rate, a 13.23% improvement over our best context-independent (CI) baseline, and a 7.5% improvement over our best context-dependent (CD) baseline. In the second task, we build a domain-independent connected digit recognition by training on the TIMIT database, and testing on the TIDIGITS database [9]. Using the same rescoring algorithm, we reduce digit error rate to 4.03% from 4.54% which is obtained with the conventional Viterbi decoding algorithm.

The rest of the paper is organized as follows. We first present knowledge scores computation in Section 2. Lattice rescoring is described in Section 3. Experimental results are then presented in Section 4. Finally, we discuss our findings in Section 5.

2. Knowledge-based scores

We use articulatory information as knowledge source. The main reason for this choice is because these features are related to human speech production, and they have shown their robustness to noise and cross-speaker variation [3]. Furthermore, standard acoustic features, such as mel-frequency cepstrum coefficients (MFCCs) [10], and articulatory features, when combined, have proved to be very useful for robust automatic speech recognition task [11]. ANNs are often adopted to map MFCCs into articulatory information, a reason is because their output can be interpreted as a good approximation of the a posteriori probability of observing an articulatory attribute for the given input, for instance as in [5], and [6]. Indeed, in this work we build segment-based detectors which are more reliable in spotting segments of speech. This assumption is supported by the findings reported in [7]. We build a bank of 15 detectors realized with HMMs [7] which map a segment of speech into one of the articulatory classes. We also propose frame-level LLR as knowledge scores to measure the goodness-of-fit between a speech segment and the corresponding knowledge sources. LLR has already proved its usefulness in rejecting unlikely hypotheses in several speech tasks. For example, in [12], LLR is used in the

verification stage to prune unlikely hypotheses. A one-pass decoder to produce multiple theories entirely based upon LLR was proposed in [13]. Additionally, in [14] a hybrid decoder based on a generalized confidence score was proposed, and LLR proved useful in rejecting low confidence local path. A LLR verification score is proposed in [15] to perform speech understanding based on key-phrase detection and verification. In this study we adopt a feed forward ANN to combine LLR scores into phone scores. The scores at the phone level are then used in the rescoring phase which will be discussed in section 3. In the following sections we present the procedure to compute LLR and knowledge-based scores.

2.1. Computing LLR scores

When segment-based detectors are implemented with HMMs, LLR can be computed at a segment level, $LLR^{(s)}$, or at a frame level, $LLR^{(f)}$. In the first case only the long term knowledge (broad level) is generated, and so much of the information is smeared out. This information manifests itself with only one score for the entire segment. In contrast, $LLR^{(f)}$, which carry short term knowledge (local level), provides a more detailed information which is conveyed to us as a sequence of scores. We think that this local information is beneficial to discriminate between target model and the corresponding competing model. To validate this assumption, we use the well-known generalized log likelihood ratio (GLLR) measure [16]. In Figure 1, $LLR^{(s)}$ and $LLR^{(f)}$ for the vowel class are compared. In the top panel the GLLR plot using segment-based scores is shown. The bottom panel shows the GLLR plot using frame-based scores. It is evident that the overlap region in the top panel is larger than that in the bottom panel, and so $LLR^{(s)}$ is more likely to generate false alarms, and false rejections than $LLR^{(f)}$. Our findings are in line with the results presented in [14], in which log likelihood is combined with LLR to address several decoding problems. It was found that the use of $LLR^{(f)}$ yields a higher recognition rate than the use of phone-level and word-level LLR.

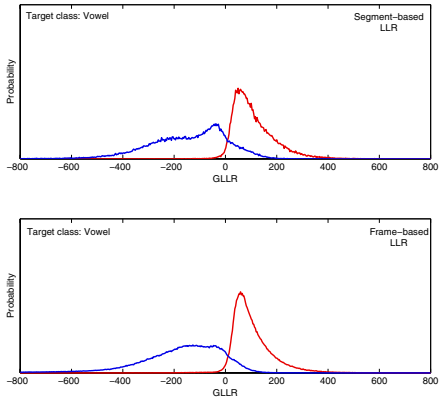

Figure 1: *LLR Model separation. The y-axes scale is linear with 0% at the base and 100% at the top.*

Usually $LLR^{(f)}$ can be computed in several way. Here we adopt the formulation presented in [14], i.e. the $LLR^{(f)}$, generated by an observation vector o_t in state i at time t, is defined as,

$$LLR_i^{(f)}(o_t) = log \frac{P(o_t|\lambda_i)}{P(o_t|\lambda_i^a)} \quad (1)$$

where λ_i is the target HMM model of the articulatory class ending in state i, and λ_i^a is its corresponding competing model. In this scheme two potential difficulties can arise. The first is the selection of the competing model for each articulatory class. Defining the competing model is not easy for certain classes such as the silence and superfluous sounds. In this study, we address the issue by training the competing model on data which do not correspond to the target model, for instance, all of the "non-nasal" data are used to generate an HMM for "non-nasal" unit. The second difficulty is the selection of the path in order to synchronize the log likelihood computation of the target model and of the competing model. Since we assume that target and competing models have the same HMM topology, a single optimal state sequence is decoded and assumed to be the same state sequence for the target model and its corresponding competing model, as in [14]. Furthermore, $LLR_i^{(f)}(o_t)$ scores for the given segment are computed along a well defined state sequence, so they are not independent one to the other. As a result, these scores do not fluctuate as the ANN-based scores presented [6], and they generate less number of wrong segments.

Figure 2 compares segment and frame detectors for the fricative manner. For the segment detector both $LLR^{(s)}$ and $LLR^{(f)}$ detection curves are presented. The $LLR^{(s)}$ is computed by averaging over the number of frames in the segment. The top panel shows the spectrogram. The middle panel shows the reference segments. The bottom panel shows the detection curves. In order to report all the scores to the same range of values, we apply a sigmoid limiter to the LLR scores as in [14]. The detected segment, achieved by the segment detectors in the bottom panel, is more similar to the reference segments. For the ANN-based scores, we observe extra segment due to noisy scores if we set a threshold to 0.5 (shown as the dashed line). Another property that makes LLR more appealing than ANN-based scores is its proved ability at pruning unlikely hypotheses during the decoding, for instance [14], and [15].

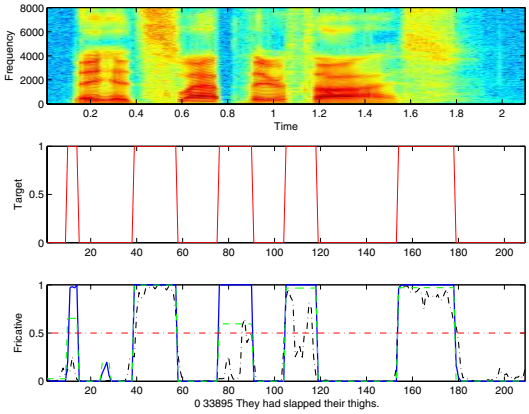

Figure 2: *Detection Curves for the fricative attribute. In the bottom panel, the dashdot, the solid, and the dashed lines represent ANN-based, $LLR^{(f)}$, and $LLR^{(s)}$ scores respectively*

2.2. Phone-Level scores

In order to come up with phone level scores which will then be used in the lattice rescoring phase, we use a non-linear function realized by a feed forward ANN with one hidden layer with 100 hidden nodes. We stop the training phase when the error rate on the validation set reaches a plateau. The ANN is fed with the $LLR_i^{(f)}(o_t)$ output of the 15 articulatory detectors, and it is

trained in a frame-wise way. Its output can be thought as an estimate of the a posteriori probability of the phone Ph_j at time t, $p_t(Ph_j|LLR_i(o_t))$ ($j = 1, ..., P$; where P is the total number of phones). The ANN output is a sequence of classes at frame level for each feature stream.

3. Lattice rescoring

A speech lattice generated by a set of HMMs can be defined as a graph, *G(N, A)*, with *N* nodes, and *A* arcs. The timing information is embedded into the nodes; while the arcs carry the symbol along with the scores information. The basic idea behind lattices is that they represent a great number of alternative theories in a compact way. As already stated, their advantage over *n*-best lists is that lattices provide a wider searching space, and they also avoid the drawback of representing many identical theories in competing strings, for the given spoken utterance. In order to illustrate the above concept, we compute upper bound accuracies of *n*-best lists and lattices generated by a continuous phone recognizer using CI, and CD models [6]. We compute the upper bound accuracies pretending to have perfect phone level scores ($p_t(Ph_j|LLR_i(o_t))$). The results are listed in Table 1, with *n* equal to 100. These results indicate that lattices give higher upper bound accuracies in both coarse and detailed models, as expected.

Table 1: *Upper Bound Accuracies.*

CI *100*-Best	CI lattice	CD *100*-Best	CD lattice
66.43%	87.86%	71.10 %	88.07%

Because we do not perform an exhaustive search and since we do not have perfect models, the lattices do not always contain the target string. Therefore, we consider the string with the highest score as the best approximation to the target string. This means that we have an upper-bound on accuracy that constrains the rescoring performance. Thus it is easy to understand that more room for improvement will allow us, in the future, to achieve higher performance if we develop better rescoring techniques.

In the next section, we present rescoring performances of phone and word lattices. We denote the rescored log likelihood value as S_n for the given arc, and compute it as,

$$S_n = w_{ps} * PS_n + w_l * L_n \quad (2)$$

where L_n is the log likelihood of the *n*-th arc; PS_n is a linear combination of $PS_{n,m}$ for each arc, with $PS_{n,m}$ being a non linear transformation of the score of the *m*-th frame for the *n*-th arc (logarithm operation on $p_t(Ph_j|LLR_i(o_t))$ discussed in Section 2.2). We do not perform weight tuning, and w_{ps}, and w_l are both set to be equal.

4. Experimental results

All the experiments were carried out by training on the TIMIT database, which is a high quality speech corpus labeled at both the phone and word levels. The training set is composed of 3696 utterances. The HMM-based detectors and the combining ANN were trained on a subset of 3504 randomly selected utterances, while the remaining 192 utterances were used as a validation set. HMM-based detectors for 15 speech attribute, namely fricative, vowel, stop, nasal, approximant, low, mid, high, labial, coronal, dental, velar, retroflex, glottal, and silence were obtained as in [7]. Therefore, a pair of CI target and competing models were trained for every event of interest. Each HMM has 3 states with 32 Gaussian mixture components per state.

4.1. Phone lattice rescoring

HTK [17] is used to build the CI and CD baseline phone recognition systems, as in [6]. The HVite tool of HTK allows the generation of a lattice of hypotheses, for each given utterance. We generate two phone lattices, based on monophone and triphone models, respectively.

The performances for continuous phone recognition task are reported in Table 2 in terms of phone error rate. In the monophone case the relative improvement is 13.23%. Furthermore, in the triphone case the relative phone error rate improvement is 7.5%. When the baseline is more detailed, there is a decrease in relative performance improvement. For the sake of completeness, we also report lattice rescoring performances obtained using ANN-based scores, computed as in [6]. These performances are 38.6% and 35.14% for monophone and triphone models respectively. The results indicate that $LLR_i(o_t)$-based rescoring outperforms the ANN-based rescoring under all work condition.

Table 2: *Phone lattice rescoring performance*

Phone error rate	CI Phone	CD Phone
Baseline	40.52%	36.13%
Rescore	35.16%	33.42%

4.2. Word lattice rescoring

Now that we have proved that LLR-based knowledge scores are better than ANN-based ones and that lattices give more margin of improvement, we want to show that the ASAT paradigm based on detection of low level events can be easily plugged into the rescoring of more complicated lattices, such as word lattices. In this session, we provide evidence of detection-based approach rescoring word lattices.

We use the connected digit recognition task as our first attempt of word lattice rescoring with knowledge scores. We have to recognize only 11 words, yet the task is made more challenging by training on TIMIT and testing on TIDIGITS, that is, by simulating a domain-independent environment. From this cross-database setup many problems arise since the two databases have a different sampling rate, were built for different purposes (*task mismatch*), and the data were collected in different acoustical environments, for instance, different microphones and different sound rooms (*acoustical mismatch*). In the current setup TIDIGITS was designed for evaluating speaker independent recognition algorithm of connected digit strings, and the data were collected with a 20KHz sampling rate; on the other hand, TIMIT was built to incorporate sufficient variability to analyze the acoustic realization of phonetic segments in terms of contextual dependencies, syntactic effects, and speaker-specific factors, and the data were sampled at 16KHz.

We cope with the acoustic mismatch issues by performing some preliminary signal conditioning on data sets. In particular, we downsampled TIDIGITS utterances to 16KHz, and reduced the strong acoustic discrepancy by constraining the training and testing MFCCs features to vary in the same range of values. The last operation is implemented with a speaker-by-speaker zero mean and unit variance normalization of the MFCCs features. Moreover, mean normalization is performed only to the static coefficients, and variance normalization applies to both static and dynamic coefficients. We address the task mismatch issue by using an HMM-based system with CI phones models as described in section 4.1. Finally, we want to point out that in this first study, we

do not investigate other possible normalization approaches, such as utterance by utterance normalization, nor do we explore other methodology to better handle cross-database issues.

As we stated, the normalization process mitigates the acoustic discrepancies issue, yet it induces a mismatch between the input domain of the manner and place detectors and of the CI phone models. As a result, if we were to use directly the scores computed in Section 2.2 in the word lattice rescoring procedure, the performance would be very poor. To lessen this secondary effect, we retrain the detectors. This operation needs not be performed in the absence of such cross-database problems. The phone-level scores are computed as in Section 2.2. HVite is again used to generate the word lattices. Nevertheless, standard HVite provides only one level of model alignment, in this particular case at a word level. In order to perform rescoring at the phone level, we also generate phone level alignment for each arc of the lattice. Finally, since phone-level scores can be too sensitive to the change at the boundaries of consecutive phones, we define another score at the end of each word, W_n, to provide a smoothing effect. W_n is a non-linear combination of $PS_{n,m}$ at a word level. The new weighted rescoring formula is defined as follows,

$$S_n = w_{ps} * PS_n + w_{wrd} * W_n + w_l * L_n. \quad (3)$$

where the three weights are set to have a weighting power of 20%, 40%, and 40% respectively.

The rescoring performance, presented in Table 3, indicate a 11.23% word error rate (WER) improvement over the baseline. We also compute the WER upper bound, which is equal to 1.69%. It is clear that our detection-based approach strategy outperforms the conventional decoding scheme in all the proposed tasks. Finally, for the sake of completeness, we want to point out that if digit-specific database is used, TIDIGITS task usually results in better WER than the presented one. Nevertheless, we purposely introduced a mismatched condition to illustrate that our approach is task and domain independent.

Table 3: *Word lattice rescoring performance*

WER	CI Phone
Baseline	4.54%
Rescore	4.03%

5. Summary

We propose a lattice rescoring procedure based on knowledge scores generated with a bank of 15 detectors for manner and place of articulation. $LLR^{(f)}$s are used as scores, and it is shown that from frame-based information a higher separation between competing model is achieved. In addition, we show that $LLR^{(f)}$ scores do not fluctuate as much as ANN-based scores, and thus the number of wrong speech segment is reduced. The experimental results that detection-based approach outperforms conventional ASR systems in all of the presented task. Moreover, this is the first attempt to improve the WER of ASR systems in the ASAT project. We believe that our performance can be further improved by adjusting the parameters of the knowledge extracting module. Finally, we intend to explore other knowledge scores and rescoring strategies, such as non-linear combination of different scores, and study new detector architectures to combine multiple spatial and temporal events aiming at improving the ASR performance.

6. Acknowledgements

Part of this effort was supported under the NSF garnt grant, IIS-04-27113, and an IBM Faculty Award. The first author is indebted with his colleagues Chengyuan Ma, and Yu Tsao of the Georgia Institute of Technology for the insightful discussions on this topic.

7. References

[1] Rabiner, L. R., "A tutorial on hidden Markov models and selected applications in speech recognition", Proc. IEEE, vol. 77, 1989.

[2] Lee, C. H., "From knowledge-ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition", Proc. ICSLP, 2004.

[3] Kirchhoff, K., "Combining articulatory and acoustic information for speech recognition in noisy and reverberant environments", Proc. ICSLP, 1998.

[4] Launay, K., Siohan, O., Surendran, A.C., and Lee, C.-H., "Towards knowledge-based features for HMM based large vocabulary automatic speech recognition", Proc. ICASSP, 2002.

[5] Hacioglu, K., Pellom, B., and Ward, W., "Parsing speech into articulatory events", Proc. ICASSP, 2004.

[6] Li, J., Tsao, Y., and Lee, C.-H., "A study on knowledge source integration for rescoring in automatic speech recognition", Proc. ICASSP, 2005.

[7] Li, J., and Lee, C. H., "On Designing and Evaluating Speech Event Detectors", Proc. InterSpeech, 2005.

[8] Garofolo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G. , Pallett, D. S., and Dahlgren, N. L., "DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus", U.S. Dept. of Commerce, NIST, Gaithersburg, MD, February 1993.

[9] Leonard, R. G., "A database for speaker independent digit recognition," Proc. ICASSP, 1984.

[10] Davis, S., and Mermelstein, P., "Comparison of parametric representations for monosyllable word recognition in continuously spoken sentences", IEEE Trans. on Acoust., Speech and Signal Process., Vol. 28, 1980.

[11] Kirchhoff, K., "Robust Speech Recognition Using Articulatory Information," Ph.D. thesis, University of Bielefeld, 1999.

[12] Sukkar, R. A. and Wilpon, J. G., "A two pass classifier for utterance rejection in keyword spotting", Proc. ICASSP, 1993.

[13] Lee, C. H., "A unified statistical hypothesis testing approach to speaker verification and verbal information verification",Proc. COST Workshop Speech Technology Public Telephone Network: Where Are We Today?, Sept. 1997.

[14] Koo, M. W., Lee, C. H., and Juang, B. H., "Speech Recognition and Utterance Verification Based on a Generalized Confidence Score", IEEE Trans. Speech and Audio Proc., vol. 9, November 2001.

[15] Kawahara, T., Lee, C. H., and Juang, B. H., "Flexible Speech Understanding Based in Combined Key-Phrases Detection and Verification", IEEE Trans. Speech and Audio Proc., Vol. 6, November 1998.

[16] Tsao, Y., Li, J., and Lee, C. H., "A Study on Separation between Acoustic Model and Its Applications", Proc. InterSpeech, 2005.

[17] Young, S. Evermann, G., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., and Woodland, P., The HTK Book, Cambridge University, 2002.

[16] King, S., and Taylor, P., "Detection of Phonological features in Continuous Speech using Neural Network," Computer Speech and Language, v. 14, n. 4, Oct. 2000.