A STUDY ON KNOWLEDGE SOURCE INTEGRATION FOR CANDIDATE RESCORING IN AUTOMATIC SPEECH RECOGNITION

Jinyu Li, Yu Tsao and Chin-Hui Lee

School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0250, USA
{gtg781p, gtg690n}@mail.gatech.edu, chl@ece.gatech.edu

ABSTRACT

We propose a rescoring framework for speech recognition that incorporates acoustic phonetic knowledge sources. The scores corresponding to all knowledge sources are generated from a collection of neural network based classifiers. Rescoring is then performed by combining different knowledge scores and uses them to reorder candidate strings provided by state-of-the-art HMM-based speech recognizers. We report on continuous phone recognition experiments using the TIMIT database. Our results indicate that classifying manners and places of articulation provides additional information in rescoring, and achieving improved accuracies over our best baseline speech recognizers using both context-independent and context-dependent phone models. The same technique can also be extended to lattice rescoring and large vocabulary continuous speech recognition.

1. INTRODUCTION

With the increasing usage of data-driven learning frameworks, such as hidden Markov model (HMM) (e.g. [1]) and artificial neural network (ANN) (e.g. [2]), we have witnessed a fast technology progress in automatic speech recognition (ASR) in recent years. However, effort in integrating additional knowledge sources into state-of-the-art HMM based systems has only resulted in very limited successes. In order to improve ASR performance, we usually rely on collecting more data to train more detailed models. It is believed that diagnostic information provided by acoustic phonetic knowledge is potentially beneficial to ASR. In this study we explore ways to incorporate such knowledge sources into ASR design.

One way to integrate knowledge sources into ASR is to extract "knowledge-based" front-end features. In [3] and [4], such features are used to train new phone HMMs. In this paper we propose the generation of "*knowledge scores*" and use them to rescore *N*-Best candidate lists provided by the conventional HMM-based systems with a given set of models. Plenty of knowledge sources can be used (e.g. [4]). As proposed in this paper, we use articulatory knowledge that is related to human speech production. Such features are known to be robust to speech variations.

Two groups of techniques to score speech vectors can be implemented. The first (e.g. [3], [4], [5]) uses MFCCs [6] as input features. The algorithm usually adopts neural network to simulate the posterior probabilities of the feature, given the speech vector. Another group (e.g. [7], [8], [9]) uses knowledge-based features to detect *acoustic landmarks*. Support vector machine based classifiers for manner of articulation have also been designed using a set of 13 knowledge-based features under a probabilistic framework [10]. In the current study, we use neural network to classify speech frames into attribute categories and use the classification scores to feed into a score combination algorithm to obtain corresponding phone scores.

N-Best lists and word lattices [11] are usually used in multi-pass search to rescore candidate theories based on more detailed acoustic and language models. We propose an *N*-Best rescoring algorithm by combining phone scores obtained from different knowledge sources. It works well for both coarse and detailed models.

We evaluate the proposed algorithm on continuous phone recognition using the TIMIT database [12]. Our experimental results indicate that classifying manners and places of articulation provides additional information in rescoring, and resulting in improved accuracies over our best baseline speech recognizers using both context-independent and context-dependent phone models. The same technique can also be extended to lattice rescoring and large vocabulary continuous speech recognition.

2. KNOWLEDGE SCORING

To provide a set of scores to measure a goodness-of-fit between a speech frame and an individual knowledge source in order to rescore multiple theories, we first compute speech parameters related to the particular knowledge source. Then we design a classifier to evaluate a corresponding *knowledge score* used for classification. To avoid confusion with the term "feature" commonly used in the front end of conventional ASR systems, we will call these knowledge based features as "attributes".

In the most general framework shown in Figure 1, For any given speech attribute, A_i, FE_i stands for a feature extraction module that converts a speech signal, $x(t)$, into a sequence of speech parameter vectors, $\boldsymbol{Y}_i$. SC_i is an attribute scoring module that computes knowledge score, $\boldsymbol{KS}_i$. $\boldsymbol{KS}_i$ can be interpreted as a goodness-of-fit score between $\boldsymbol{Y}_i$ and A_i. In some cases, these scores simulate the *a posteriori* probabilities, $p(A_i|x(t))$, of an attribute given the speech signal. Attribute-specific feature extraction modules can be designed to evaluate sound-specific parameters, and then fed into various scoring routines. In this study, we use the 10-msec MFCC vectors for all FEs. We will explore other speech parameterization techniques in the future. There are also many techniques for designing the classification modules. To simplify our study, we use ANNs for all scoring modules. Different from those in [3] and [5], which use neural network to classify the entire manner or place attribute groups, we design one classifier for every single attribute. It is highly plausible that the optimal speech parameters and corresponding classifier or detector can be different for each speech attribute. Our design enables us to incorporate new speech parameters to design new attribute classifiers and detectors in future studies.

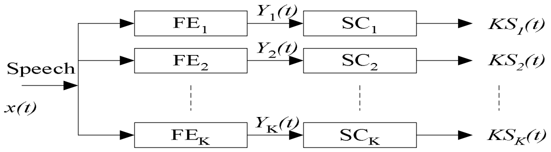

Figure 1 Knowledge extraction module

For every attribute in Figure 1, we use a feed-forward neural network to perform feature extraction. All the networks share the same structure, no parameter tuning was performed. The input to the networks has 9 frames (frame rate is 10 msec in the current system) of 12MFCCs + energy, giving a total of 117 input nodes. The hidden layer has 100 nodes. The output layer has only one node, and its value is 1 if the desired attribute is present at the center frame, and 0 otherwise. The silence attribute appears in both groups, only one classifier is needed for it. So we designed a collection of 15 neural networks, one for the each of the 15 attributes of the manner and place of articulation listed in Table 1. This list was adopted from [3]. We will incorporate other knowledge sources to better discriminate phones in future research.

Attribute Group	Attributes
Manner	vowel, stop, fricative, approximant, nasal, silence
Place	low, mid, high, dental, labial, coronal, palatal, velar, glottal, silence

Table 1 Manner and place of articulation attributes

3. N-BEST RESCORING

ROVER [13] is a convenient tool to combine ASR systems and it sometimes leads to a significant performance improvement when these systems use complementary features and exhibit different error patterns (e.g. [4]). For strings in an N-Best list obtained from a single system, like in our current study, we believe ROVER will not help much. Instead we use knowledge sources in a two-stage ASR system to improve performance as shown in Figure 2. In the first stage, the N-Best list is generated by using the baseline HMM based speech recognizer. The knowledge scores needed in the second stage are computed by passing speech through the knowledge extraction module described in Figure 1. These knowledge scores are then fused with some combination weights to get phone scores. We then combine phone scores into string scores and rescore the *N*-Best candidate strings to obtain the final recognized results. The rescoring algorithm is described as follows:

Step1. For the *m*-th frame of the speech signal ($m \in [1, M]$, *M* is the total number of frames in the test utterance), get the individual knowledge scores, $\boldsymbol{KS}_{i,m}$, of the *i*-th attribute ($i \in [1, K]$, *K* is the total number of attributes) with the extraction module in Figure 1.

Step2. Given the *m*-th frame and the *n*-th candidate string ($n \in [1, N]$, *N* is the total number of candidates), use the set of knowledge scores, $\boldsymbol{KS}_{i,m}$, to come up with a phone score, $\boldsymbol{PS}_{n,m}$, as described in Figure 3.

Step3. For every *n*-th candidate string weight $\boldsymbol{PS}_{n,m}$ of all the frames to obtain the total phone score for candidate n as: $PS_n = \sum_{m=1}^{M} w_m PS_{n,m}$. The weight w_m can be chosen according to the confidence of speech frame classification. We chose all the frame weights to be 1/*M* for simplicity here.

Step4. For every candidate, compute the final string score as: $S_n = \alpha_K PS_n + \alpha_L L_n$, where L_n is the log likelihood computed from the baseline HMM recognizer for candidate n . Here, we didn't tune the weights, and simply chose $\alpha_K = \alpha_L = 0.5$. Better result may be achieved with different weights. Finally, we select the candidate with the largest S_n as the final recognition result.

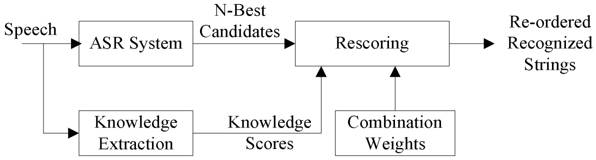

Figure 2 Two-stage ASR with knowledge sources

In Step2 above, a combination module is used to map the knowledge scores, ***KS***, to phone scores, ***PS***. In our current work, simulated attribute probabilities are used as ***KS*** and are fused together in order to simulate the *a posteriori* probabilities of phones, *p(*Ph$_j$ |*x(t))* ($j \in [1, P]$, P is the total number of phones) as shown in Figure 3. In our implementation for the combination module, we use a feed-forward ANN with 100 hidden nodes. Given the *m*-th frame and the *n*-th candidate string, we obtain the phone identity, Ph$_j$, from the corresponding transcription. Take the logarithm of *p(*Ph$_j$ |*x(t))* as the phone score, $\boldsymbol{PS}_{n,m}$, for the candidate frame.

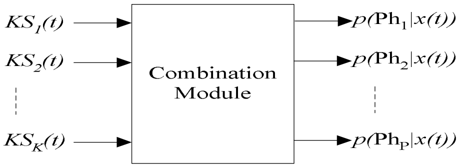

Figure 3 Combination module for rescoring

4. EXPERIMENTS

The TIMIT database is used in all the following experiments. Excluding utterances used for speaker adaptation (SA), there are a total of 3696 and 192 utterances in the training and core testing sets, respectively. To train the HMMs, we used the entire training set. To train ANN-based classifiers and combination network, we randomly selected 3504 utterances as the training set, and the remaining 192 utterances serve as a validation set.

4.1 Baseline Phone Recognition System

Starting from the 48 context-independent (CI) phones defined in [14], we merged the phones “cl”, “vcl” and “epi” into the phone “sil”, and reduced the TIMIT phone set to a set of 45 CI phones. We then trained one set of monophone models and another collection of triphone HMMs by using HTK [15]. The input features are 12MFCCs + energy, and their first and second order time derivatives. For the CI-phone based recognizer, every monophone has 3 states, and every state has 16 mixture Gaussian components. For the context-dependent (CD) phone based recognizer, we obtained a total of 995 shared states, each has 8 Gaussian mixture components. Only acoustic models we employed in continuous phone recognition, no language models were used. The performance is measured by the phone accuracy rate (Acc). The Acc for the monophone and triphone based recognizers are 59.48% and 63.87%, respectively. These baseline results are similar to those reported in [16].

4.2 ANN-Based Attribute Classifiers

To train the proposed ANN-based attribute classifiers, NETLAB [17] was used. We evaluated them with frame error rate. For every frame, we chose the class with the largest output value as the result of frame classification for the manner or place attribute group. Confusion matrix of the manner attributes is listed in Table 2. The *(i, j)*-th element of the confusion matrix indicates the classification rate of the *i*-th attribute being categorized into the *j*-th class. For example, we can see the approximant attribute gives the lowest classification rate of 56.5%. In addition, 32.3% of the approximant frames were wrongly classified as the vowel attribute. These frame level classification problems can often be alleviated by a better definition of attributes. We will also explore segment-based models and scores, like HMMs, to improve the performance. In this study, we are mostly interested in obtaining frame-based knowledge scores. The overall frame error rates of manner and place attribute group are 17.9% and 26.8%, respectively. The frame error rates for the place attributes are plotted in Figure 4. Our results are again comparable with those reported in [5], showing a slightly worse performance for the manner attributes and a slightly better performance for the place attributes. Attribute bigram models and duration models, used in [5] to reduce the classification error rates, were not used in our experiments.

As shown in Table 2 and Figure 4, large differences exist among various attribute classes. The silence attribute achieved the best frame classification accuracy of 92.9%. The vowel, fricative and retroflex attributes have error rates of less than 20%, while approximant, mid, dental and glottal attributes give error rates of more than 40%. One possible explanation is that the MFCC based spectral features, currently used as the input features for all ANNs, work well to classify some attributes, but fail in other cases where temporal features may be more discriminative. This motivates us to explore other speech parameters to design attribute classifiers. In the framework shown as Figure 1, we can flexibly change the input parameters of individual classifiers, without affecting other classifiers.

%	vowel	frica -tive	stop	nasal	approxi -mant	silen -ce
vowel	**89.0**	1.5	1.5	1.8	6.0	0.2
fricative	3.8	**85.2**	6.8	1.2	1.3	1.7
stop	7.6	11.0	**72.5**	2.9	2.1	3.9
nasal	11.2	2.5	4.8	**77.5**	3.2	0.8
approxi -mant	32.3	2.9	3.7	3.2	**56.5**	1.4
silence	1.1	1.2	3.2	0.7	0.9	**92.9**

Table 2 Confusion matrix for manner attributes

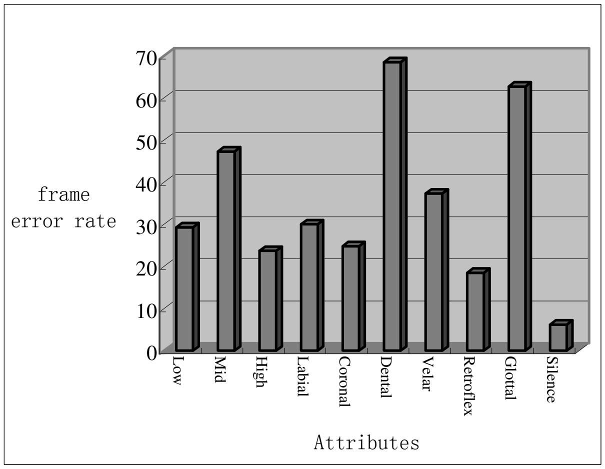

Figure 4 Frame error rate of place attributes

4.3 Rescoring Performance

In order to achieve an improved performance over the baseline systems, we need the *N*-Best lists from both the monophone and triphone based recognizers. We experimented with both the 24-best and 100-best lists for rescoring. If the candidate that best matches the reference phone transcription of the test utterance is chosen, we can get the upper bound accuracies of our rescoring algorithm shown in Table 3. It is clear that with more candidates and better models, the upper bound accuracies always improve.

	24-Best list	100-Best list
Monophone	64.41%	66.43%
Triphone	69.08%	71.10%

Table 3 Upper bound accuracies for *N*-best rescoring

Since the strings that achieve the upper bound accuracies are not known, we picked the candidate with the largest score as the recognized string. Table 4 lists the result of rescoring. Although the absolute Acc rate improvement is rather small, it is very important to realize that the N-Best list only provides us a very small room for performance improvement, as limited by the performance upper bounds listed in Table 3. Even if our classifiers are perfect and the knowledge scores are capable of discriminating among all attributes, we can only attain these upper bounds. In the current situation, with imperfect classifiers and limited knowledge sources, we only reach an Acc rate between the baseline and the upper bound achievable by the given N-Best list. Therefore it is more meaningful to measure a relative Acc improvement shown in the bottom row of Table 4.

$$\text{Relative Acc Improvment} = \frac{\text{Rescoring Result - Baseline}}{\text{NBest List Upper Bound - Baseline}}$$

Acc	CI Phone 24-Best	CI Phone 100-Best	CD Phone 24-Best	CD Phone 100-Best
Upper Bound	64.41%	66.43%	69.08%	71.10%
Baseline	59.48%	59.48%	63.87%	63.87%
Rescore	60.80%	61.13%	64.55%	64.72%
Relative	26.8%	23.7%	13.0%	11.8%

Table 4 Relative performance improvement

It is interesting to note that the relative Acc improvement obtained by our rescoring algorithm was over 20% for monophone based systems. This Acc improvement was reduced to about 10% when triphone based systems are evaluated. It seems that the performance improvement for the triphone based recognizer is not as much as monophone based recognizer, because the attribute classifiers help less when detailed models are used. It is imperative to design more accurate classifiers to integrate more useful knowledge sources into our rescoring algorithms. It is also noted that the absolute Acc improvement of the 100-best list is better than that of the 24-best list. That is to say that with more candidates, we can achieve a better accuracy. We therefore expect to get better error rate reduction when word or phone lattices are used in rescoring, because they have the ability to provide more varieties of candidates than the *N*-Best list.

5. CONCLUSION

We have proposed a string rescoring approach to ASR by incorporating knowledge sources into computing knowledge scores and reordering *N*-best candidate strings. Based on classifying manner and place of articulation, the corresponding attribute scores provide additional discrimination information, and therefore improve the overall phone recognition accuracy. Different rescoring strategies have been evaluated. The best performance improvement was obtained by using a neural network to combine outputs of the attribute classifiers.

Many research issues are worth pursuing. Our classifiers are all based on 10-msec MFCCs. If we can design classifiers with corresponding optimal input features, including both short time temporal features, such as voice onset time, and long term features, such as pitch contours, we expect to improve the classification rate, and therefore boost the overall system performance. In addition to manner and place of articulation, other knowledge sources can be incorporated to rescore the N-Best list. We also intend to explore other strategies for rescoring word and phone lattices. The graphical models [18] seem to be an ideal tool to model the complex interactions among different acoustic phonetic attributes. Furthermore, many new types of knowledge scores and confidence measures can also be combined to improve the effectiveness of rescoring.

Acknowledgement

Part of this effort was supported under the NSF SGER grant, IIS-03-96848.

REFERENCES

[1] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition”, *Proc. IEEE*, Vol.77, No.2, pp257-286, 1989.

[2] S. Haykin, *Neural Networks: a Comprehensive Foundation (2nd edition)*, Prentice Hall, 1998.

[3] K. Kirchhoff, “Combining articulatory and acoustic information for speech recognition in noisy and reverberant environments,” *Proc. ICSLP98*, Sydney, Australia, 1998.

[4] B. Launay, O. Siohan, A.C. Surendran, and C.-H. Lee, “Towards knowledge-based features for HMM based large vocabulary automatic speech recognition,” *Proc. ICASSP02*, Orlando, pp. 817-820, 2002.

[5] K. Hacioglu, B. Pellom, and W. Ward, “Parsing speech into articulatory events,” *Proc. ICASSP04*, Montreal, Canada, pp.925-928, 2004.

[6] S. Davis, and P. Mermelstein, “Comparison of parametric representations for monosyllable word recognition in continuously spoken sentences,” *IEEE Trans. on Acoust., Speech and Signal Process.*, Vol. 28, No. 4, pp. 357-366, 1980.

[7] A. M. Ali and J. V. Spiegel, “Acoustic-phonetic features for the automatic classification of fricatives,” *J. Acoust. Soc. Am.*, Vol.109, No.5, pp.2217-2235, 2001.

[8] N. Bitar and C. Espy-Wilson, “The design of acoustic parameters for speaker-independent speech recognition,” *Proc. Eurospeech97*, Patras, Greece, pp. 1239-1242, 1997.

[9] K. Stevens, “Estimating distinctive features in speech,” *J. Acoust. Soc. Am.*, Vol.111, No.4, pp. 1872-1891, 2002.

[10] A. Juneja and C. Espy-Wilson, “Segmentation of continuous speech using acoustic-phonetic parameters and statistical learning,” *Proc. International Conference on Neural Information Processing*, Singapore, 2002.

[11] R. Schwartz, L. Nguyen, and J. Makhoul, “Multiple-Pass Search Strategies” in *Automatic Speech and Speaker Recognition*, C.H. Lee, F.K. Soong, and K.K. Paliwal, eds. Klewer Academic Publishers, pp. 429-456, 1996.

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

[13] J. G. Fiscus, “A post-processing system to yield reduced word error rates: recognizer output voting error reduction (ROVER)”, *Proc. ASRU*, pp. 347-352, 1997.

[14] K. F. Lee and H. W. Hon, “Speaker-independent phone recognition using hidden Markov models”, *IEEE Trans. on Acoust., Speech and Signal Process.*, Vol. 37, No. 11, pp. 1641-1648, 1989.

[15] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.2)*, Cambridge University, 2002.

[16] S. Young, “The general use of tying in phoneme-based HMM speech recognisers,” *Proc. ICASSP92*, pp.569-572, 1992.

[17] I. T. Nabney, *NETLAB: Algorithms for Pattern Recognition*, Springer, 2001.

[18] M. I. Jordan, *Learning in Graphical Models*, MIT Press, 1998.