A Study on Detection Based Automatic Speech Recognition

Chengyuan Ma, Yu Tsao and Chin-Hui Lee

School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA
{cyma, yutsao, chl}@ece.gatech.edu

10.21437/Interspeech.2006-104

Abstract

We propose a new approach to automatic speech recognition based on word detection and knowledge-based verification. Given an utterance, we first design a collection of word detectors, one for each lexical item in the vocabulary. Some pruning strategies are used to eliminate unlikely word candidates. Then these detected words are combined into word strings. The proposed approach is different from the conventional maximum a posteriori decoding method, and it is a critical component in building a bottom-up, detection-based speech recognition system in which knowledge in acoustics, speech and language can easily be incorporated into pruning unlikely word hypotheses and rescoring. The proposed approach was evaluated on a connected digit task using phone models trained from the TIMIT corpus. When compared with state-of-the-art connected digit recognition algorithms, we found the proposed detection based framework works well even no digit samples were used for training the detectors and recognizers. Other knowledge based constraints, such as manner and place of articulation detectors, can be incorporated into this detection-based approach to improve the robustness and performance of the overall system.
Index Terms: speech recognition, detection-based.

1. Introduction

Research on automatic speech recognition (ASR) has witnessed dramatic progress and great success in the last several decades. More improvements have been obtained in the field of speech and language modeling due to the extensive use of statistical learning techniques, more and more speech and language data collections. However, some challenging problems still exist within the prevailing ASR framework. One of them is the robustness in adverse conditions. The acoustic mismatch between the training and testing will cause the ASR performance to drop a lot. Meanwhile, linguistic mismatches, such as out of vocabulary and out of grammar events will cause misrecognition. One reason for these limitations is that current ASR framework is a top-down, data-driven black box. That is, it provides very little diagnostic information for error correction and further improvement. Furthermore, ASR robustness issues are often caused by ignoring the detail knowledge in acoustics, speech, language and their interactions. One way to incorporate knowledge sources into ASR system designs is through bottom-up detection of fundamental speech unit followed by knowledge integration [1]. Some attempts were conducted to find robust distinctive feature which are invariant to speaker and speaking environments [2] [3]. Meanwhile, many knowledge supplemental modeling techniques have been investigated to incorporate available knowledge sources into state-of-the-art hidden Markov model (HMM) based ASR system. But it's difficult to incorporate many knowledge sources into a single search network as required by the maximum *a posteriori* (MAP) decoding paradigm.

When compared with human speech recognition (HSR), state-of-the-art ASR systems usually have a much larger error rate even in clean environment. There is strong evidence that human speech recognition starts at a bottom-up analysis [4]. Then multiple knowledge sources are integrated into the recognition process. To realize such a knowledge-driven ASR framework, a new detection-based, knowledge-rich speech recognition paradigm has been proposed [5]. It implies a new approach to solving the robustness problem and also takes advantages of the rich literatures in phonetics, acoustics and linguistics. Conventional data-driven statistical learning algorithms for ASR can also be further extended by incorporating diverse knowledge sources. The detection-based ASR paradigm is flexible in integrating many different kinds of knowledge sources. Because knowledge about the speech is explicitly built into the ASR system , the error correction and improvement can be made in a directed and meaningful manner.

In this study, we demonstrate one implementation of this detection-based, knowledge-rich ASR framework. Our proposed framework of the detection-based ASR is shown in Figure. 1. It consists of three parts: (1) word detectors design; (2) knowledge guided word hypothesis verification and false alarm pruning ; and (3) combining word hypothesis into word string. The system is realized for connected digit recognition that the small size vocabulary task facilitate many ways of integrating knowledge sources in ASR design.

By comparing with the state-of-the-art connected digit recognition algorithms, we found the proposed detection based framework works well even no task-dependent samples were used for training the detectors and recognizers. Other knowledge based constraints, such as place of articulation detectors, can be easily incorporated into this detection-based approach to improve the robustness and performance of the overall system.

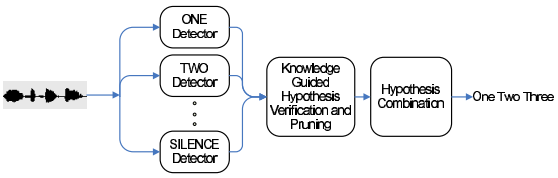

Figure 1: *Framework of Detection-based ASR System.*

2. Word detector design

Many existing techniques (e.g., artificial neural network (ANN), support vector machine (SVM) and HMM) and many knowledge sources can be used for designing detectors at different stages, e.g., word, sub-word and attribute levels. For connected digit recognition system, all the detectors are on the word level. We have a separate detector for each lexical item in the vocabulary. One of the basic principles for designing detectors is to detect as many candidates as possible to avoid candidates missing. That is, we expect to have many false alarms while keeping the missed detection rate as close to zero as possible. In this implementation, HMM modeling techniques are used for detector design. For each digit, a set of monophone models are trained from the training set. The key issue for HMM based detector design is how to choose an appropriate grammar network. A simple and intuitive example for detecting a word is shown in Figure. 2. For each target word, it will compete with its corresponding anti-model and a silence model when decoding. The drawback of this design is that it will result in many missed detection errors.

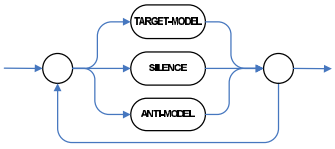

Figure 2: *Simple Network of Digit Detector.*

A more complicated and elaborate network for word detector is shown in Figure. 3. Now for each target word, we introduce its cohort models which are the most competitive word models and a silence model as the filler to absorb all the other events except for the target word. With this network, less misses will occur. This is a very general detector design. One practical issue is how to select the cohorts for each target word. As a extreme example, for each target digit, we can choose all the other digits as its cohorts.

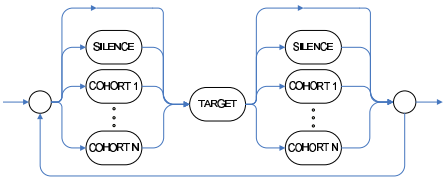

Figure 3: *General Network of Digit Detector.*

Figure. 4 shows an example of the output of the 11 digit detectors. The first and second panel are the waveform of the test utterance $31o2$ and its corresponding spectrogram. The following 11 panels are the detector outputs with the level on the Y-axis for each panel indicating a confidence measure for detecting these words. For example, the bottom panel has three segments above the X-axis. It means that the "oh" detector tells us these segments are digit "oh". Actually, only the second segment is really a digit "oh". The first and the third one are false alarms.

3. Word verification and pruning

It's obvious that these detectors generate a lot of false alarms just as we expect. To improve the recognition performance and reduce

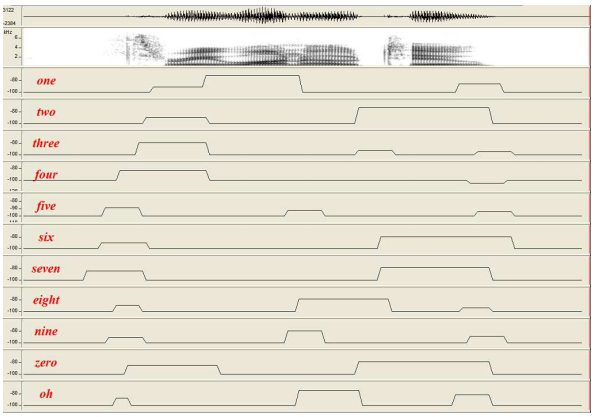

Figure 4: *Hypotheses Generated by 11 Detectors.*

the computational complexity of the recognition process, it is desirable to verify these digit hypotheses and prune some of the false alarms. Word verification is formulated as a statistical hypothesis testing problem [6]. The likelihood ratio or generalized likelihood ratio is a good testing statistic for verification. One practical issue is to determine the threshold to accept the detector outputs or reject them. Knowledge guided hypothesis verification and pruning is at the core of the detection-based ASR paradigm. All knowledge sources available from acoustic, phonetic and linguistic research can be exploited for false alarm elimination. In the following, three pruning strategies will be presented. We expect more will come out from the community.

3.1. Temporal information based pruning

For example, phone dependent duration constraint is one simple pruning strategy. The duration constraints can be used to eliminate those short segments in the detection result. The statistics of phone duration can be obtained from the training set. For example, the duration of the word "one" (/w/-/ah/-/n/) should be greater than 150 ms.

3.2. Attributes model based pruning

Another method is to use models of the manner and place attributes to generate the attribute sequence for each detected segment. Each manner attribute is modeled with a HMM. Then for each detected segment, it can be decoded as a sequence of manner attributes. If correctly decoded, each word has its own attribute sequence pattern. Any obvious deviation from the desired pattern can be pruned by some rules. For example, among all the outputs of detector "one", some of them are actually from speech for "nine". So we can prune those segments whose manner attribute sequence doesn't contain glides. This kind of model based pruning techniques have shown their effectiveness in our evaluation experiments.

3.3. Signal based pruning

Model based pruning can easily be implemented and used. However, we still need to train these manner attribute models from some training set. Inevitably, the robustness problem still exists. So it's desirable to have some robust pruning strategies. Signal

feature based pruning is one of them. For example, from research in acoustics, we know that the energy of a nasal sound /n/ is often concentrated on the low frequency region (below 400 HZ), while the fricative sound /f/ has a relatively flat spectrum and energy distribution in high frequency region. So the percentage of low frequency energy in the total energy is useful and robust in distinguishing the nasal and fricative sound. Also the formants position of vowels and other spectral features can be used to distinguish certain pair of sounds.

4. Hypotheses combination

After hypothesis verification and false alarm pruning, we investigate hypothesis combination strategies using outputs from all detectors to generate a word string efficiently and accurately.

4.1. Hypothesis lattice conversion

The weighted directed graph (WDG) is one of the methods that can be used to combine the detector output into a digit string. The hypothesis combination can be formulated as a search problem on a weighted directed graph G, which is a pair (V, E), where V is a set of vertices, and E is a set of edges between the ordered vertices $E = \{(u, v)|u, v \in V\}$. Meanwhile, there is a weight $W_{u,v}$ associated with each edge.

The following procedure can be used to convert the hypothesis lattice into a directed graph.

1. Constructing the node set, V, which consists of all the detected digit boundaries. For instance, for one detected segment (T_a, T_b), both T_a and T_b will be elements of V.
2. Ranking all the detected boundaries in a time line and adding a edge for each pair of adjacent nodes in the graph in order to guarantee the existence of a path from the start node to the end node.
3. For each detected segment, adding a edge from its start node to its end node.
4. Adding reversal edge to those nodes which are very close to each other (e.g. within 20 ms) or merging these nodes into one node, due to the potential overlap in the detected boundaries.

4.2. Search in the weighted directed graph

Given the constructed directed graph, the weight we choose should be consistent with our search criterion. For example, when the search is based on the maximum likelihood criterion, the log-likelihood can be used as the weight. Of course, we can put other score metrics to each edge under a certain criterion.

Finding the best path in a WDG is a well studied problem in computer science and operation research. So finding the best matched string over the detector output lattice is equivalent to finding a path with the maximal weight. The well-known Dijkstra's algorithm can be used to find the best matched path. To further improve the recognition performance by rescoring with other detectors' results, the KSP (K-shortest path) algorithm [7] can be used to find the *K*-best digit strings. Figure. 5 is the WDG converted from Figure. 4. Each node in the graph is a detected digit boundary. The number in the node is the time stamp (in 10 ms). Each edge represent a detected digit or a silence edge. The number beside each edge is the frame average log-likelihood. And the red edges are the best path we obtained for the utterance $31o2$.

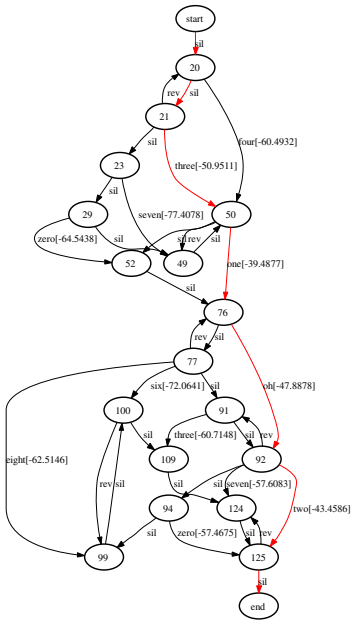

Figure 5: *Weighted Directed Graph.*

5. Experiment setup and result analysis

All the evaluation experiments are conducted on the TIDIGITS corpus [9]. The digit vocabulary is made of 11 digits, one to nine, plus oh and zero. The training set has 8623 digit strings and the test set has 8700 digit strings. A conventional procedure is used for front-end processing. 12-dimensional MFCC and the log-scaled energy were extracted for each 10-ms frame. Their first and second order derivatives are also computed for each frame. To conduct cross-corpus evaluation and reduce the channel effects, every element of the feature vector has been normalized with zero-mean and unit variance.

5.1. Whole word model in matched condition

In this experiment, the training set from the TIDIGITS corpus are used to train the whole-word HMM model for each digit. Each HMM has 12 states and use a simple left-to-right topology without state-skip. A state-of-the-art HMM based ASR system and a detection-based ASR system are built for comparison. The conventional HMM based ASR gave a word error rate about 0.48% and the detection-based ASR was slightly worse at 0.73%. So in the matched acoustic condition, the detection-based system can get comparable results as the conventional ASR system.

5.2. Monophone model in mismatched condition

Now we simulate a real ASR scenario. We purposely introduced a mismatched condition to illustrate the benefits of incorporating knowledge into the detection based ASR system. TIMIT [8] was used for mono-phone model training while the TIDIGITS was down-sampled from 20 KHz to 16 KHz and used for testing. Each mono-phone model is a 3-state left-to-right HMM. A conventional Viterbi-based ASR system and a detection-based ASR system were built for the experiment. The deletion, substitution and insertion errors of step-by-step knowledge-based pruning are

shown in Table 1.

The word error rate of the conventional ASR system is 4.54%. And for the detection-based ASR system without pruning, it is 6.37%. It's clear that the detection-based system has much more substitution and insertion errors.

Duration Pruning: When we took a close look at the recognition results of the detection-based ASR system, we found too many short segments were detected and recognized as words. So the phone-dependent duration constraints can be imposed on the detection results. After pruning with the duration constraints, the word error rate of the detection based ASR system was reduced to 5.03%. The insertion errors were reduced from 791 to 351, while the deletion errors increase from 167 to 227.

Manner Pruning: We also observed that some confusion pairs are very significant in the word confusion matrix. For example, five/nine (ground-truth/recognized result), five/four, one/nine, eight/three, seven/five, four/oh, etc. Some of these substitution errors can be reduced by manner model based pruning discussed in Section 3.2. The rules used for pruning can be learned from some development data by decision tree. The manner sequence pattern pruning method can generally be used to prune those clear confusions. The overall performance after manner model based pruning is 4.23%. We can see that the substitution errors were reduced from 860 to 620 and the insertion error were reduced from 351 to 302.

Signal Feature Pruning: Signal feature based pruning is often more meaningful and robust. The spectral features of nasal and fricative can be used in five/nine confusion pair. The substitution errors of five/nine were reduced from 51 to 11 by using the low frequency energy ratio and a voicing detector. As for the eight/three confusion pair, the spectrum before the segment /iy/ in three and segment /ey/ in eight are different due to the existence of the fricative /th/ and glides /r/. With a voicing detector and high frequency energy ratio in these two segments, we can reduce the substitution of eight/three from 56 to 24. Similar work can be done on other confusion pairs to reduce those hard confusions. Now the overall performance was improved to 3.74%. The substitution errors have been further reduced (from 620 to 524), while the deletion errors was increased a little (from 258 to 286).

From our experiment results, this kind of signal feature based pruning is very promising. It should be noted that no digit model was used in digit detection and pruning. For reference only, if we use the digit-specific models for pruning, the word error rate of the detection-based ASR system is 2.15%. It's much better than the result of conventional state-of-the-art ASR system. It shows that even if the acoustic model for detector design is not perfect, we can still have very good recognition performance by word detection and appropriate pruning strategies. We want to point out that if digit-specific database is used with a new discriminative training algorithm, the string accuracy of TIDIGITS task is 99.33% [10].

Table 1: *ASR result.*

	Del.	Sub.	Ins.	Word Err. (%)
Detection W/O Pruning	167	864	791	6.37
W/ Duration Pruning	227	860	351	5.03
W/ Manner Pruning	258	620	302	4.23
W/ Feature Pruning	286	524	260	3.74
Digit-specific Pruning	370	118	126	2.15
Conventional ASR	469	617	211	4.54

6. Summary and future work

In this paper, we demonstrated one implementation of the detection-based, knowledge-rich speech recognition paradigm. Our experiment results show that by explicitly incorporating our knowledge about the speech and language into our detector design and pruning strategy, the performance of the detection-based ASR system can be improved step by step in a meaningful and directed manner. It's also noted that the performance improvement in the proposed system is additive. That is, a better module for a feature will not produce as much poorer result for the individual module and overall performance. The word verification and pruning strategies mentioned in this paper are still faraway from being perfect. We are expecting more reliable knowledge sources to be detected. In future studies, more knowledge sources will be incorporated into the framework for hypothesis pruning. In addition, some post-processing can be done on the N-best candidates. We are more interested in investigating the detection-based ASR system for LVCSR tasks.

7. Acknowledgement

This work was partially supported by the NSF ITR grant, IIS-04-27413.

8. References

[1] Lee, C.-H., "On Automatic Speech Recognition at the Dawn of the 21st Century," *IEICE Trans. Inf. & Syst.*, pp. 377–396, 2003.

[2] Liu, S. A., "Landmark Detection for Distinctive Feature-based Speech Recognition," *JASA*, pp. 3417–3430, 1996.

[3] Juneja, A. and Espy-Wilson, C., "Segmentation of Continuous Speech Using Acoustic-Phonetic Parameters and Statistical Learning," *Proc. ICONIP*, vol. 2, pp. 726–730, 2002.

[4] Allen, J. B., "How Do Humans Process and Recognize Speech?" *IEEE Trans. Speech and Audio Processing*, vol. 2, no. 4, pp. 567–577, 1994.

[5] Lee, C.-H., "From Knowledge-Ignorant to Knowledge-Rich Modeling: A New Speech Research Paradigm for Next Generation Automatic Speech Recognition," in *Proc. InterSpeech*, 2004.

[6] Sukkar, R. A. and Lee, C.-H., "Vocabulary Independent Discriminative Utterance Verification for Non-Keyword Rejection in Subword Based Speech Recognition," *IEEE Trans. Speech and Audio Processing*, vol. 4, no. 6, pp. 420–429, 1996.

[7] Epstein, D., "Finding the K-Shortest Paths," in *SIAM J. Computing*, pp. 652–673, 1998.

[8] Garofalo, 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 CD-ROM," Tech. Rep., U.S. Dept. of Commerce, NIST, Gaithersburg, MD, 1993.

[9] Leonard, R. G., "A Database for Speaker-Independent Digit Recognition," in *Proc. ICASSP*, 1984.

[10] Li, J., Yuan, M. and Lee, C.-H., "Soft Margin Estimation of Hidden Markov Model Parameters," submitted to *InterSpeech* 2006.