

TOWARDS BOTTOM-UP CONTINUOUS PHONE RECOGNITION

Sabato Marco Siniscalchi¹, Torbjørn Svendsen¹, and Chin-Hui Lee²

¹Department of Electronics and Telecommunications
Norwegian University of Science and Technology, Trondheim, Norway

²School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332 USA

{marco77, torbjorn}@iet.ntnu.no, chl@ece.gatech.edu

ABSTRACT

We present a novel approach to designing bottom-up automatic speech recognition (ASR) systems. The key component of the proposed approach is a bank of articulatory attribute detectors implemented using a set of feed-forward artificial neural networks (ANNs). Each detector computes a score describing an activation level of the specified speech attributes that the current frame exhibits. These cues are first combined by an event merger that provides some evidence about the presence of a higher level feature which is then verified by an evidence verifier to produce hypotheses at the phone or word level. We evaluate several configurations of our proposed system on a continuous phone recognition task. Experimental results on the TIMIT database show that the system achieves a phone error rate of 25% which is superior to results obtained with either hidden Markov model (HMM) or conditional random field (CRF) based recognizers. We believe the system's inherent flexibility and the ease of adding new detectors may provide further improvements.

Index Terms— Knowledge based system, Speech recognition, Detectors, Feedforward neural networks

1. INTRODUCTION

In the field of automatic speech recognition (ASR), the prevailing modeling paradigm is knowledge-ignorant in the sense that data-driven techniques are employed. Better performance can be achieved mainly by collecting more and more data. As a result, a large body of literature in speech and language sciences is mostly unused in the modern ASR systems.

In recent years, many speech researchers have shown that proper integration of knowledge sources into state-of-the-art ASR systems effectively improves the the recognition accuracy. For example, in [1] phonologically inspired features are generated and used to train a set of hidden Markov models (HMMs) to build a LVCSR system. The ROVER algorithm is then used to merge this system with a conventional baseline system, and a relative error reduction of about 20% over the baseline system is reported. In [2], the authors show that a 50% error rate reduction over state-of-the-art ASR performance can be achieved by reordering competing hypotheses generated by a two-stage alphabet decoder according to the voice onset time (VOT) measurement.

Automatic Speech Attribute Transcription (ASAT) was recently proposed as a candidate framework for developing next generation ASR techniques and systems [3]. ASAT is a bottom-up, knowledge-rich modeling paradigm in which three main components are out-

lined: (a) a bank of feature detectors that can produce consistent detection results, even in adverse condition; (b) an event merger that combines low level events (articulatory and/or acoustic phonetic events) into higher level evidence, such as phones or words; and (c) an evidence verifier that generate a lattice of event hypotheses that can be optionally sent back to the event merger. Preliminary experiments reported in [4] and [5] have shown that statistical significant error reduction can be achieved by rescored competing hypotheses of a conventional ASR system according to the detector outputs. Nonetheless, these approaches are far from the original ASAT paradigm, and can be seen as feasibility studies that prove the validity of the detector-based approach.

In this paper, we propose a full bottom-up continuous phone recognition system which aspires to be a first step towards a better fulfillment of the ASAT vision. Therefore, the three blocks outlined above, that is, a bank of speech event detectors, an event merger, and an evidence verifier are implemented. The event detectors are trained on mel-bank energies derived features, and are forced to generate phonological posterior probabilities. The event merger is trained using the detectors output and generates phone posterior probability. Finally the evidence merger delivers either a single phone string hypothesis or a lattice of these hypotheses. The core of this architecture is the bank of speech detectors which make the system highly flexible. The number and topology of the detectors can be changed: each detector can be built using a different design methodology, and a different set of features can be used for each detector. DET curves and log-likelihood ratio (LLR) plots are tools that can guide the feature selection phase.

An important aspect of detection-based ASR systems is that they give the opportunity to better understand the flaws in the recognizer. For example, if the /p/ sound is systematically confused with the /b/ sound, this may show that the voicing detector needs to be improved. Moreover, the detection-based approaches inherently provide a platform in which expert knowledge of linguistic and acoustic phonetics can be methodically incorporated into the system.

Experimental results on a continuous phone recognition task show that our system outperforms conventional HMM-based ASR systems, and compare favorably against the CRF-based system trained on either phonological or phone classes features. The TIMIT corpus [6] was used for training and testing purposes in all the experiments.

The paper is organized as follows: in the next section we describe the overall system, and provide details for each of the three components. In Section 3 we give an overview of the experimental setup and discuss the results. Our conclusion and discussion about future follow in Section 4.

2. SYSTEM OVERVIEW

Figure 1 shows a block diagram of our detector-based system which, as aforementioned, consists of three main blocks, (1) a bank of speech event detectors, (2) an event merger, and (3) an evidence verifier. More details about each block will be provided in the following sections. Emphasis will be placed on the explanation of the bank of event detectors since it is the core of our architecture. In Figure 1, we show also a set of front-ends since a different speech parametric representation may be eventually used for each detector. In this paper, we use the same set of speech features for all the detectors. Moreover, the evidence verifier may output either the best decoded hypothesis or a lattice of hypotheses. The latter could be used for further refinement steps, such as lattice rescoring. In this work, the evidence verifier provides only the first best hypotheses.

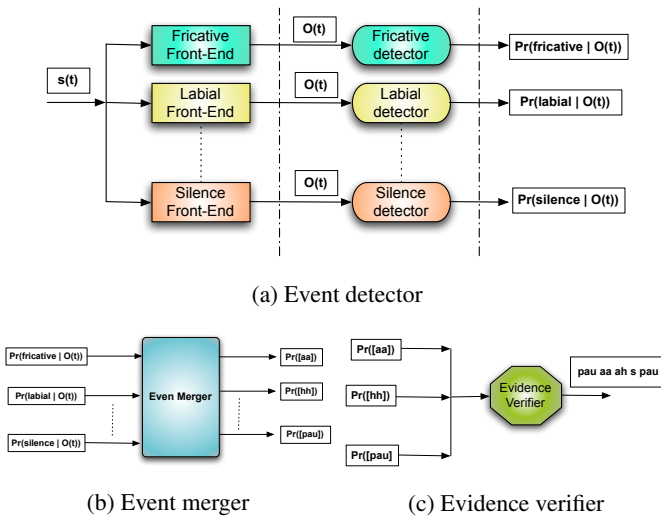


Fig. 1. Overall system.

2.1. Speech Event Detectors

The goal of each detector is to analyze the speech signal and produce a confidence score or a posterior probability that pertains to some acoustic-phonetic attribute. As pointed out in [7], event detection is a challenging task, and it is usually more complicated than conventional signal detection in many aspects. For example, speech events do not follow a well-defined theory, and show a wide variation in their duration that can range from few milliseconds to seconds. Event detection is also complicated by the fact that speech is usually collected in different acoustic conditions and over a large population of speakers so that speech events often exhibit a great deal of variation. In addition, detailed timing in event detection is a critical component since we often need to combine these detected events to form higher level evidence. Therefore, we can no longer afford having highly variable segment boundaries.

Both frame and segment based detectors can be used for speech event detection. Frame based detectors may be built using ANNs; whereas, segment based detectors can be implemented with HMMs. An obvious advantage in using segment based detector is that they are more reliable in spotting segments of speech as shown in [7], but the detection curves are not synchronized in time making the event merger task much more difficult. If we want all the detection curves

to be synchronized in time, we need segmentation information. Although this information could be provided for the training utterances, it is not available for testing utterances. To circumvent these issues, we use frame based detectors, we build each detector using a feed-forward ANN with one hidden layer and 500 hidden nodes. The ANN is trained by classical back-propagation algorithm with cross entropy error function. The softmax activation function is used in the output layer, and the ANN produces the posterior probability that the speech event happened during the frame being processed.

Table 1 lists the set of phonological features that we use in our experiments. This set of 22 features is a "wise" combination of the Sound Pattern of English (SPE) feature defined by Chomsky and Halle [8], and the manner and place of articulation features proposed in [4].

List of speech attribute
fricative glide liquid nasal stop vowel
coronal dental glottal high labial low mid retroflex velar
anterior back continuant round tense voiced silence

Table 1. Speech attributes.

2.2. Event Merger & Evidence Verifier

The event merger combines the event detectors outputs together with different weights and delivers evidences at a phone level. The event merger is implemented using a single feed-forward ANN with one hidden layer and 800 hidden nodes. The softmax activation function is used in the output layer. The ANN is trained using classical back-propagation algorithm with cross entropy error function and classify phone states (3 states for each phone). The training algorithm starts with labels derived from the TIMIT manual transcription and with uniform segmentation of phones into states.

The evidence verifier is a decoding network which consists of a set of context independent phone models layered in parallel and with uniform entrance probability. A 3 emitting states left-right HMM models a single phone, all transition probabilities are set equal to 0.5. The HMM state likelihood is the phone posterior probability of the event merger scaled by the prior phone probability. We assume equal prior probability for all phones. The Viterbi algorithm performed over the decoding network provide the decoded sentence.

3. EXPERIMENTS

In the next sections, first we describe the experimental setup, data corpus, and the software used to build the system. Then we present the results followed by some comments.

3.1. Experimental setup

Corpus: All the experiments were conducted using the TIMIT corpus, which is a high quality speech corpus labeled at both the phone level and the word level. In this paper, only a subset of the corpus (i.e. SI ad SX sentences only) has been used (3696 sentences during training and 1344 sentences during testing). A small development set (400 sentences) was split off from the training partition and was used to decide when to stop the ANNs training. The original 64 phonetic labels were mapped into 39 phones as prescribed in [9].

Baselines: For comparison purpose we built two systems based on Gaussian mixtures model (GMM) and HMM. In particular, the first

system uses monophone models with 3 states per model and 16 Gaussian mixture components per state. The second system uses tri-phone models with 3 states per model and 8 Gaussian mixture components per state. The input features are 12 Mel-Frequency Cepstrum Coefficients (MFCCs), plus logE, and their first and second order time derivatives. The system were trained on the complete training set, and evaluated on the 1344 sentences of testing set. We refer to these two systems as baseline. A 0-gram language model was used for all the experiments.

Features: Several parametric representations of the speech signal can be used in the proposed system. Among all possibilities, we consider and compare three different sets of parametrization: (1) 12th order MFCC features (along with log energy), plus velocity and acceleration coefficients, (2) a nine-frame window centered around the frame being processed of 12th order MFCCs (along with log energy) plus velocity and acceleration coefficients, and (3) energy trajectories in mel-frequency bands. In the latter case, 23 mel filters are used, for each mel filter a vector of 31 frames centered around the frame being processed is generated and then down-sampled to 11 coefficients by DCT. In this section, we refer to the three parametrization as *MFCC*, *9 - MFCC*, and *MBE*, respectively. For all experiments, the speech waveforms are analyzed with a window length of 25 ms and a step size of 10 ms.

Figure 2 shows the detection (DET) curves for the low attribute (left panel). Both *MFCC*-based detectors perform worse than *MBE*-based one. Thus, in all our experiments we will use *MBE* based features if it is not explicitly stated otherwise. It is also worth noting that if the detectors were built to generate posteriors at a state level (e.g., 3 states per speech event) the detection curves would be even better, as shown in right panel of Figure 2 for the nasal attribute. We can think of this three-state construction either as a detector based on sequential information (e.g. HMM), or as a deterministic information merging from state to attribute before phone merging, i.e., the detector will integrate multiple sequential information before giving a single (or multiple) detector output.

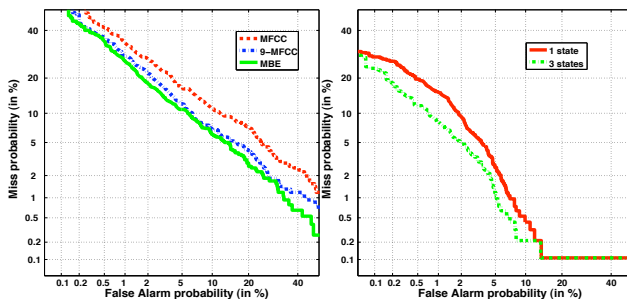


Fig. 2. In the left panel, the DET curves of the low event detectors shown. The dashed, the dashdot, and the solid lines represent the *MFCC*, *9 - MFCC*, and *MBE* features, respectively. In the right panel, the DET curves of the nasal detectors when trained with *MBE* based feature to generate event posteriors (solid curve) or state event posterior (dashed curve) are shown

Software: The GMM/HMM systems were built with the HTK speech toolkit [10]. All the ANNs were built using the ICSI QuickNet neural network software package¹. The Viterbi algorithm used to generate the recognized phone sequence was implemented using the STK

toolkit².

3.2. Results & Discussion

The best way to make the reader appreciate the flexibility of our detector-based system is to show some of the several experiments that have been carried out in our laboratories. In our first experiment, a bank of 16 speech event detectors out of 22 was used, namely, fricative, approximant (glide and liquid), nasal, stop, vowel, coronal, dental, glottal, high, labial, mid, retroflex, velar, voiced, and silence. The detectors deliver posterior probabilities at a speech attribute level. Table 2 shows the phone accuracy of the baseline systems and our system as given by the HTK's HResults tool. Table 2 also lists the best CRF-based system performance as reported in [11]. In [11], the CRF model is used as mathematical tool to combine phone- and phonological-based classifiers. The reader is referred to [11]. With this configuration, our system achieves better performance than the two conventional baselines in term of phone accuracy, but worse than then CRF based systems [11]. Nonetheless, in all of the CRF-based configuration, the CRF model is fed with an higher dimensional input vector (from 44 up to 122). Thus, we increased the number of detector from 16 to 22. This operation required only to retrain the new detectors and the event merger. Since some redundancy was introduced Karhunen-Loeve (KL) transform is performed on the event merger input. Table 3 shows that the new performance compares favorably against the best results achieved with CRF-based system (see first two rows).

Model	Labels	Phone Accuracy
Baseline 1	monophones	61.39%
Baseline 2	triphones	66.57%
CRF (phone-based)	monophone	70.10%
CRF (phonological-based)	monophone	69.13%
16 Detectors	monophones	66.85%

Table 2. Baselines, CRF-based, and detector-based system performances in terms of phone accuracy.

Model	Phone Accuracy
22 Detectors	69.36 %
22 Detectors + KL	70.38 %
22 Detectors (three-state) + KL	72.31 %

Table 3. Detector-based system performance in terms of phone accuracy.

In the Section 3.1, it was shown that three-state detectors can achieve a better detection performance, so we wonder whether this improvement cascades through the following stages. We retain all the detectors and the event merger. The new performance is listed in the last row of Table 3 and confirms our hypothesis. This indicates that a positive or negative change in the detection performance may predict similar effect on recognizer performance. This can be a key point of our detection-based approach.

In all our previous experiments, we have adopted the scheme showed in Figure 3.a. Nonetheless, it has been shown that the use of different ANN for independently processing (1) groups of energies in different frequency bands (TRAPs - Figure 3.b) [12], (2) different temporal context with respect to the frame being processed (STC - Figure 3.c) [9], or (3) different blocks of spectral vector (TILES -

¹ICSI quicknet software package, <http://www.icsi.berkeley.edu/speech/qn.htm>

²STK toolkit, <http://www.fit.vutbr.cz/speech/sw/stk.html>

Figure 3.d) helps improve the recognition performance. Whenever several ANNs are employed to process the critical band based features, the detectors are actually acting as a merger that combines ANN outputs, as highlighted in Figure 3.

In the TRAPs-based system, two *broad-bands* were generated by combining 13 critical bands together (one ANN per broad band is used). In the STC-based system, for each critical band a window of 310ms centered around the frame being processed is considered and split in two halves: left-context and right-context (one ANN per context is used). The TILES-based system is a combination of the two above system (one ANN per tile is used). Table 4 shows the performance, in term of phone accuracy, of all the previous configurations when hand-labeled transcription is used to train the merger, i.e., no realignment followed by retraining is performed. The TRAPs-based system performs the worst, and this may be due to that we should have joined 3 or 5 bands together as shown in [9]. TILES-based system performs slightly worse than the STC-based system and this may be explained by the fact in the TILES-based system the ANNs are trained with lower dimensional feature vectors. Further experiments should be performed to verify these hypotheses, but they go beyond the goal of this paper.

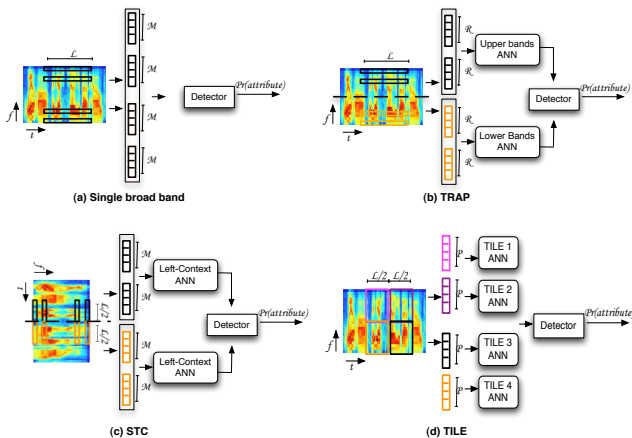


Fig. 3. Several possible schemes to process critical-band based features.

Model	Phone Accuracy
TRAPs	71.32%
STC-2	73.73%
TILES	73.23%
STC-2 (+realignment)	75.00%

Table 4. Phone accuracy for several critical band based features schemes.

Finally, by performing force-alignment and retraining the phone accuracy of STC-based system achieves the value of 74.82%. If the number of hidden nodes in the event merger is increased up to 1000, the phone accuracy becomes 75.00 %, as shown in the last row of Table 4. This last value is comparable, in terms of 95% confidence interval, to recently proposed high-accuracy phone recognizers (e.g.,[13] [9]).

4. CONCLUSIONS

We have presented a novel approach to design bottom-up, knowledge-rich ASR systems. We have validated our approach on a continuous phone recognition task using the TIMIT corpus. Experimental results have shown that our detector-based system outperform conventional systems, and it achieves a better phone accuracy than similarly trained CRF-based ASR systems. Moreover, our system reaches a performance as good as recently proposed high-accuracy phone recognizers, but it is by design more flexible. In fact, the number of detectors and their topology can be changed on the fly and only the event merger needs to be retrained. Finally, the parametric form of the speech signal can be chosen to better suit the specific of each single detectors. This aspect is currently under investigation in our laboratories along with experiments on speech recognition at a word level.

5. REFERENCES

- [1] B. Launay, O. Siohan, A. C. Surendran, and C.-H. Lee, “Towards knowledge-based features for hmm base large vocabulary automatic speech recognition,” in *ICASSP*, 2005.
- [2] P. Niyogi and P. Ramesh, “A detection framework for locating phonetic events,” in *ICSLP*, 1998.
- [3] H. Lee, C, “From knowledge -ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition,” in *ICSLP*, 2002.
- [4] J. Li, Y. Tsao, and C.-H. Lee, “A study on knowledge source integration for candidate rescoring in automatic speech recognition,” in *ICASSP*, 2005.
- [5] M. Siniscalchi, S., J. Li, and C.-H. Lee, “A study on lattice rescoring with knowledge scores for automatic speech recognition,” in *Interspeech*, 2006.
- [6] J. S. Garofolo, Lamel L. F., Fisher W. M., Ficus J. G., Pallett D. S., and Dahlgren N. L., “Darpa timit acoustic phonetic continuous speech corpus,” in *U.S. Dept. of Commerce, NIST*, 1993.
- [7] J. Li and C.-H. Lee, “On designing and evaluating speech event detectors,” in *InterSpeech*, 2005.
- [8] N. Chomsky and M. Halle, *The Sound Pattern of English*, MIT Press, 1991.
- [9] Schwarz P, Matějka P, and Černocký J., “Hierarchical structures of neural networks for phoneme recognition,” in *ICASSP*, 2006.
- [10] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, Povey D., V. Valtchev, and P. C. Woodland, *The HTK Book (for HTK Version 3.3)*, Cambridge University Press, Cambridge, UK, 2005.
- [11] J. Morris and E. Folser-Lussier, “Further experiments with detector-based conditional random fields in phonetic recognition,” in *ICASSP*, 2007.
- [12] H. Hermansky and S. Sharma, “Temporal patterns (traps) in asr of noisy speech,” in *ICASSP*, 1999.
- [13] L. Deng and D. Yu, “Use of differential cepstra as acoustic features in hidden trajectory modeling for phonetic recognition,” in *ICASSP*, 2007.