

JOINT OPTIMIZATION OF EVENT DETECTORS AND EVIDENCE MERGER FOR CONTINUOUS PHONE RECOGNITION

Sabato Marco Siniscalchi, Øystein Birkenes, Magne H. Johnsen, and Torbjørn Svendsen

Department of Electronics and Telecommunications, NTNU, Trondheim, Norway

{marco77, birkenes, mhj, torbjorn}@iet.ntnu.no

Abstract

In the recent years, different data-driven methods have been proposed to detect articulatory features (AF) from short-term spectral representation. The main motivations for the AF based approach are as follows. First, the AFs in general can more accurately and parsimoniously characterize the acoustic variability associated with conversational speech. Further, while not explored in this work, AFs are more language universal than phones, and therefore they can generalize better and are easier to adapt to new languages. For use in phone based systems the AF scores are input to an evidence merger which produces phone posteriors as outputs.

Several classifiers are usually built, and each classifier is trained for detecting a single articulatory feature (describing manner and/or place). We believe that joint optimization of all the classifiers and the subsequent phone evidence merger may be beneficial for the classification performance. This work is a preliminary study on this direction, and it is validated on the continuous phone recognition task. A bank of articulatory detectors, designed using hidden Markov models (HMMs), learns the mapping from the MFCC space to the articulatory space. The detectors' outputs are then combined by the evidence merger. The AF based phone posteriors is integrated into an existing ASR engine and applied to N-best rescoring. Experimental results show promising performance on the TIMIT corpus.

Index Terms: system combination, detection-based automatic speech recognition, n-best rescoring, discriminative training.

1. Introduction

Recently, there has been a resurgence of interest in event-based detection [1]. Speech events are a collection of fundamental speech properties such as acoustic landmarks and distinctive features that, if properly detected, can be used to characterize all speech sounds. These features can be used in standard speech recognition algorithms to try to improve the overall performance of a conventional automatic speech recognition (ASR) system. In order to use distinctive features to build an ASR system, it is first required to define some acoustic correlates of each distinctive features (e.g. [2]), which describe how the acoustic signal relates to the features. There exist two main techniques to measure the acoustic correlates of distinctive features: (1) using signal processing algorithms (e.g. [3]), and (2) data-driven learning machines (e.g. [4, 5]). This work concerns with the latter approach along with the integration of articulatory motivated information into an existing ASR engine.

Several data-driven paradigms have been proposed to extract articulatory motivated information for speech recognition

(e.g. [4], [6], [7]). For example, in [4] a bank of artificial neural networks (ANNs) was built to classify articulatory features. The posterior probabilities output by each ANN were concatenated and integrated by a higher-level ANN trained to generate phone posteriors. The higher-level ANN is then used in a hybrid ANN/HMM system for continuous number recognition. In [6], and [7] a set of binary value classifiers (detectors) was designed to detect the presence or absence of distinctive features directly from Mel-Frequency Cepstrum Coefficients (MFCCs). These detectors were built using ANNs and HMMs, respectively. The detectors' outputs were concatenated and passed to an higher-level ANN which provides phone posteriors to be used in a rescoring framework. In the above studies, the articulatory classifiers are used to indicate whether a particular feature exists in the frame being analyzed. Then a non-linear discriminative function (i.e. the higher-level ANN) combines the articulatory classifiers outputs and generates evidence at the phone level. We may think of the higher-level ANN as an *evidence merger*. All of the classifiers were treated as independent systems, and the training of the single articulatory classifiers was carried out separately. Moreover, the higher-level ANN and the articulatory classifiers were trained separately. We believe that joint optimization of the speech event detectors and the evidence merger using a global criterion function may lead to more reliable evidences and better recognition performance.

The purpose of the current study is to look at an alternative approach to training the articulatory classifiers and the evidence merger, and the focus is on the phone recognition task. We propose the penalized logistic regression machine (PLRM) [8] as a good candidate for this purpose for several reasons. First, PLRM is a statistically well founded classification approach that has recently been proposed for a variety of speech applications with promising performance [9, 10]. Second, defining and optimizing a criterion function that is jointly dependent upon the detector parameters and the merger parameters is straightforward inside the PLRM framework. Finally, PLRM can be discriminatively trained to directly model the conditional probabilities of speech events, which can be used during a rescoring step to integrate articulatory motivated knowledge into a conventional speech decoder. We evaluate our approach on a continuous phone recognition task using the TIMIT speech corpus [11]. We implement 15 speech detectors by using HMMs as in [7]. Differently from the above approaches, the evidence merger used to combine and integrate the detector's output is a linear discriminant function. After performing rescoring of the N-best lists generated by an external phone decoder, we observed that: (1) by jointly optimizing the parameters of the two systems we achieve better phone recognition accuracy than when the two system are optimized separately; and (2) by increasing the number of competing hypotheses in the N-best list,

This work is supported by SIRKUS project, financed by the Research Council of Norway.

the rescoring performance increases accordingly.

The paper is organized as follows. In the next section we give a presentation of the overall recognition system, and review both the speech detectors and the PLRM. Also, we outline the rescoring procedure. Section 3 describes the experimental setup and the results are presented. Section 4 contains the conclusions.

2. System Overview

The overall phone recognition system consists of two main parts: (1) a phone recognizer that provides the N-best list, and (2) a module that provides phone posterior probabilities needed during the rescoring step. The former is a phone recognizer designed using the HTK toolkit¹. The latter is the combination of a bank of speech phonetic feature detectors and linear discriminant functions re-cast into a PLRM. The overall system is a two-stage continuous phone decoder (Figure 1). The bank of speech attribute detectors and the PLRM are presented in the following section. More emphasis will be placed on the explanation of the PLRM since it is relatively new to the speech recognition community.

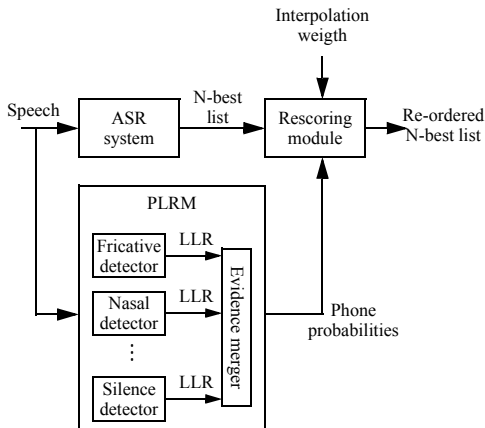


Figure 1: Two-stage phone decoder.

2.1. Phonetic Feature Detectors

The bank of speech detectors is implemented with HMMs. For a given speech segment, each detector provides evidence for one of the 15 broad classes of manner and place of articulation, namely fricative, vowel, stop, nasal, semi-vowel, low, mid, high, labial, coronal, dental, velar, glottal, retroflex, and silence. Log-likelihood ratio (LLR) between a target HMM and a competing HMM is used as the measure of goodness-of-fit between the input and the output of each detector, as in [7]. If $p(x; \lambda_m)$ denotes the likelihood of the m th target model with parameters λ_m and $p(x; \lambda_m^a)$ denotes the likelihood of its competing model with parameters λ_m^a , the output of the m th detector for an input segment x is computed as

$$LLR_m = \log \frac{p(x; \lambda_m)}{p(x; \lambda_m^a)}. \quad (1)$$

2.2. Probabilistic Prediction of Speech Segments using PLRM

In the following, we present the parametric form of the model assumed in PLRM, define the criterion function that is to be minimized in order to estimate the model parameters, and briefly

explain the optimization algorithm used to minimize the criterion function.

Let $x = (x_1, \dots, x_T)$ be a sequence of T feature vectors extracted from a speech segment, and let $y \in \mathcal{Y}$ be the phone label of x , where \mathcal{Y} is the set of K phones (in the experiments we use a set of $K = 39$ phones as explained later). PLRM estimates the conditional probability of a phone label y given a segment x using a nonlinear logistic regression model of the form

$$\hat{p}_k = \hat{p}(y = k|x, W, \Lambda) = \frac{\exp w_k^T \phi(x; \Lambda)}{\sum_{i=1}^K \exp w_i^T \phi(x; \Lambda)}, \quad (2)$$

where ϕ is a nonlinear transformation that maps a feature vector sequence x into an M -dimensional vector $\phi(x; \Lambda)$ parameterized by Λ , and w_k are $M + 1$ -dimensional weight vectors. For convenience, the weight vectors are taken to be rows of the $K \times M + 1$ -dimensional parameter matrix W . In this work, we perform the mapping ϕ with the set of $M = 15$ speech detectors introduced in the previous section in the following way:

$$x \mapsto \phi(x; \Lambda) = [1, \frac{1}{T}LLR_1, \dots, \frac{1}{T}LLR_M]^T. \quad (3)$$

The parameter of the above mapping is the set of all the HMM parameters of the detectors, i.e. $\Lambda = \{\lambda_1, \dots, \lambda_M, \lambda_1^a, \dots, \lambda_M^a\}$.

We want to jointly optimize the detector parameters and the merger parameters W . We do this by using a set of training data $\mathcal{D} = (x^{(l)}, y^{(l)})_1^L$, consisting of pairs of speech segments and phone labels, in order to estimate the pair (W^*, Λ^*) that minimizes a global criterion function. We choose the criterion function [9]

$$\mathcal{P}_\delta^{\log}(W; \Lambda; \mathcal{D}) = - \sum_{l=1}^L \log \hat{p}_{y^{(l)}} + \frac{\delta}{2} \text{trace } \Gamma W \Sigma W^T, \quad (4)$$

where the first term is the negative log of the logistic regression likelihood, and the second term is a penalty term weighted by a hyperparameter $\delta > 0$. The matrix Γ is a $K \times K$ diagonal matrix whose k th diagonal element is the fraction of training samples with the k th class label, and $\Sigma = (1/L)\Phi\Phi^T$, where Φ is an $(M+1) \times L$ matrix with columns $\phi(x^{(l)}, \Lambda)$. Although the function in (4) is convex with respect to W , it is not guaranteed to be convex with respect to Λ . A local minimum can be obtained by using a coordinate descent approach with coordinates W and Λ . For the convex minimization with respect to W , we use the method in [12]. As for the minimization with respect to Λ , we use the Rprop method [13].

2.3. Generating speech segments for PLRM training

The role of PLRM in our N-best rescoring approach is to provide conditional probabilities of a phone given a speech segment. In many cases, a speech segment in an N-best list does not correspond to a complete phone. Some segments, for example, contain only a part of a phone, or even several phones together. If such incorrect segments were not taken into account during the PLRM training phase, the PLRM would be likely to generate unreliable probability estimates. For this reason, we choose to train the PLRM with a garbage class as one of the outputs, which purpose is to provide high probability for incorrect segments and low probability otherwise. In order to train the garbage class, we need to provide the training algorithm with typical examples of incorrect segments generated by N-best lists. Hence, for the training of PLRM with a garbage class, two sets of training data have to be used; correct segments with the correct phone label, and incorrect segments with the garbage label.

¹HTK toolkit, <http://htk.eng.cam.ac.uk/>

To generate the correct segments to be used in training, we need to know the phone boundaries. These boundaries were generated by forced alignment on each sentence in the training set with an existing decoder (see Sec. 3) Thus from a pair of (z, s) , where z is a sequence of feature vectors of a sentence s with L_s phones, we obtain a set $\{(x^{(1)}, y^{(1)}), \dots, (x^{(L_s)}, y^{(L_s)})\}$ of phone-labeled segments. Doing this for all the pairs (z, s) in the training material gives a set

$$\mathcal{D}_{\text{correct}} = \{(x^{(l)}, y^{(l)})\}_{l=1, \dots, L_{\text{correct}}} \quad (5)$$

of all correctly labeled segments.

Extracting garbage segments to be used in the training of PLRM is more difficult. In the rescoring phase, segments that differ somehow from the true unknown segments should give small probability to any class in the vocabulary, and therefore high probability to the garbage class. In order to achieve this, we generate an N-best list for each training utterance, and compare all segments within the list with the corresponding forced alignment generated segments. The forced alignment segmentation is used here since the true segment boundaries are not known. The segments from the N-best list that have at least ϵ (number of) frames not in common with any of the forced alignment segments, are used as garbage segments for training. This gives us a set

$$\mathcal{D}_{\text{garbage}} = \{(x^{(l)}, K + 1)\}_{l=L_{\text{correct}}+1, \dots, L} \quad (6)$$

of all garbage-labeled segments.

The full training data used to train the PLRM is therefore

$$\mathcal{D} = \mathcal{D}_{\text{correct}} \cup \mathcal{D}_{\text{garbage}}. \quad (7)$$

2.4. Rescoring Procedure

For a given sentence hypothesis $\hat{s} = (\hat{y}^{(1)}, \dots, \hat{y}^{(L_{\hat{s}})})$ with corresponding segmentation $z = (x^{(1)}, \dots, x^{(L_{\hat{s}})})$, we can use PLRM to compute the conditional probabilities $\hat{p}_{\hat{y}^{(l)}} = \hat{p}(y = \hat{y}^{(l)} | x^{(l)})$. A score for the sentence can then be taken as the following geometric mean:

$$\hat{p}_{\hat{s}} = \left(\prod_{l=1}^{L_{\hat{s}}} \hat{p}_{\hat{y}^{(l)}} \right)^{1/L_{\hat{s}}}. \quad (8)$$

When all hypotheses in the N-best list have been rescored, they can be reordered in descending order based on their score.

Alternatively, the score obtained from (8) can be interpolated with the sentence HMM likelihood from the external phone recognizer. Let $\hat{p}(\hat{s}|z)$ denote the posterior sentence probability that can in theory be obtained from the sentence HMM likelihood $\hat{p}(z|\hat{s})$. The log of the weighted geometric mean with weight $0 \leq \alpha \leq 1$ between the two sentence scores can then be written as

$$S_{\hat{s}} = (1 - \alpha) \log \hat{p}_{\hat{s}} + \alpha \log \hat{p}(\hat{s}|z) \quad (9)$$

$$\propto (1 - \alpha) \log \hat{p}_{\hat{s}} + \alpha (\log \hat{p}(z|\hat{s}) + \log \hat{p}(\hat{s})), \quad (10)$$

since $\hat{p}(z)$ is constant for all hypotheses in an N-best list. Furthermore, if we assume that $\hat{p}(\hat{s})$ is constant we can write

$$S_{\hat{s}} \propto (1 - \alpha) \log \hat{p}_{\hat{s}} + \alpha \log \hat{p}(z|\hat{s}). \quad (11)$$

The right-hand side of the above equation is taken as the interpolated score. Note that if $\alpha = 0$, only the PLRM score is used for rescoring, while $\alpha = 1$, only the HMM score is used.

3. Experiments

All experiments were conducted using the TIMIT corpus [11], which is a high quality speech corpus labeled at both phone and word level. We mapped the 64 phonetic labels in the transcription into 39 phones [14], and ignored the glottal stop. As baseline system, and for the generation of the N-best list at a phone level, we used a HMM-based recognizer. The baseline is a context independent phone model system implemented with the HTK toolkit. Each phone model has as a 3-state HMM with a 16 component GMM per state, and is trained by Maximum Likelihood Estimation (MLE). The baseline phone accuracy, in percentage, is 65.61.

Both the PLRM module and the baseline system were trained using the standard TIMIT training set, which consists of 3696 sentences from the SI and SX material. Forced alignment segmentation of the training set was performed with the baseline system. For the extraction of the garbage segments, we used a 20-best list with $\epsilon = 3$, also generated by the baseline system. Each HMM in the bank of detectors has 3 states with 16 Gaussian mixture components per state. The initial values for the HMM detector parameters were generated using MLE. In the training of the PLRM, for each detector we updated only the target HMM models. In particular, we update only the means of the HMMs detectors while keeping the other HMM parameters fixed. For each of the coordinate descent iterations we used the Rprop method with 100 iterations to update the HMM means Λ and the Newton method with 4 iterations to update W . After 6 coordinate descent iterations, the optimization was stopped due to time limitations.

For the testing phase, we used the standard NIST 24 speaker core test set. Rescoring was done on both 20-best and 50-best lists, both generated by the baseline system. Table 1 lists the baseline performance and the upper bounds for the 20-best lists and the 50-best lists in terms of phone accuracy (Acc.).

Table 1: *Phone accuracy (Acc.) and upper bounds with 20-best lists and 50-best lists using the baseline decoder. δ updating*

	baseline	20 - best	50 - best
Acc. (in %)	65.61	70.30	71.83

3.1. Results & Discussion

Tables 2 and 3 summarize the experimental results for several values of the regularization parameter for the 20-best list and the 50-best list, respectively. Only the PLRM score is displayed, i.e., with the interpolation weight set to $\alpha = 0$. The results show that joint optimization of the event detectors with parameters Λ and evidence merger with parameter W is better than separate optimization.

By comparing the last rows of Tables 2 and 3 with the baseline performance, we can see that PLRM achieves better performance for all δ values. Although the absolute accuracy rate improvement is rather small, it is worth to notice that the N-Best lists, used for this preliminary study, allow only a small room for performance improvement, as limited by the performance upper bounds listed in Table 1. Even if the PLRM outputs perfect scores discriminating among all phones, these upper bound accuracies were the best possible attainable results. Since the PLRM is not an ideal system, we can only reach a performance between the baseline and the upper bound allowed by the N-Best list. Therefore, it is more meaningful to measure a relative Acc. improvement as

Table 2: *PLRM phone accuracy on the 20-best lists before and after joint optimization for several values of δ*

δ	10^{-1}	10^1
Disjoint Optimization	65.56	65.53
Joint Optimization	66.85	66.75

Table 3: *PLRM phone accuracy on the 50-best lists before and after joint optimization for several values of δ*

δ	10^{-1}	10^1
Disjoint Optimization	65.42	65.64
Joint Optimization	67.27	67.30

$$Rel.Impr. = \frac{PLRM Acc. - Baseline Acc.}{NBest Acc. - Baseline Acc.}, \quad (12)$$

where *Rel.Impr.* stands for the relative improvement, *PLRM Acc.* represents the accuracy with $\alpha = 0$, *Baseline Acc.* is the accuracy of the first-stage decoder, and *NBest Acc.* is the maximum achievable accuracy. Thus, the relative accuracy improvements for the 20-best and the 50-best list computed as in (12) are **26.4%** and **27.2%**, respectively. Further, an increment in the phone accuracy is observed when passing from 20-best lists to 50-best lists, so we expect the performance to improve by increasing the number of competing hypotheses.

We noted that other studies have reported better performance than the proposed baseline system by using different configurations (e.g., [15]). We used the discriminatively trained system presented in [15], referred to as BUT, as new baseline system to validate our approach when a better baseline system is used. Due to time limitation, the PLRM was trained using the standard NIST 462 speaker training set, which represents a subset of the full TIMIT training material, the value of δ was chosen equal to 0.1, and 20-best list was used in the rescoring phase. The BUT system was trained on the whole TIMIT training set, and garbage segments were extracted from the 20-best list with $\epsilon = 3$. Table 4 shows the baseline performance, the 20-best list accuracy upper bound, the performance after joint optimization with $\alpha = 0$, and the relative improvement computed with (12). As shown in the last row of Table 4, the PLRM system outperforms the BUT baseline system. Therefore, the improvement holds even with a better baseline. We believe that a better PLRM performance can be achieved by (1) using the whole training material to train the PLRM system, (2) rescoring with an higher value of N , and (3) tuning the value of δ and ϵ .

Table 4: *PLRM phone accuracy on the 20-best lists generated with BUT.*

δ	10^{-1}
Baseline	74.42
Joint Optimization	75.21
Upper bound	79.14
Rel. Impr.	16.73

We also studied the effect of interpolating the HMM score with the PLRM score for the 20-best and 50-best lists, respectively, for $\delta \in \{10^{-1}, 10\}$, but we did not observe any improvement, which might mean that the two systems make the same set of errors. Nonetheless, a better choice of N and ϵ for extracting the garbage segments, a higher value of N in the rescoring phase, and the tuning of δ may lead to significant improvement during interpolation.

4. Conclusions

We have proposed to use PLRM to jointly optimize speech event detectors and evidence merger for the use in N -best rescoring. Initial experiments show promising results. We believe that the approach has potential for further improvement, and many research directions are worth pursuing. This applies to choice of articulatory classes, adaptation of more detector parameters, phone specific weights for rescoring combination, and better definition of garbage segments.

5. References

- [1] Lee, C.-H., "From knowledge-ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition", Proc. of ICSLP, 2004.
- [2] Stevens, K., "Acoustic correlates of some phonetic categories", JASA, vol. 68, pp. 836-842, 1980
- [3] N. N. Bitar, and C. Y. Espy-Wilson, "Knowledge-based parameters for HMM speech recognition", Proc. of ICASSP, 1996.
- [4] Kirchhoff, K., "Combining articulatory and acoustic information for speech recognition in noisy and reverberant environments", Proc. of ICSLP, 1998.
- [5] Launay, K., Siohan, O., Surendran, A.C., and Lee, C.-H., "Towards knowledge-based features for HMM based large vocabulary automatic speech recognition", Proc. of ICASSP, 2002.
- [6] Li, J., Tsao, Y., and Lee, C.-H., "A study on knowledge source integration for rescoring in automatic speech recognition", Proc. of ICASSP, 2005
- [7] Siniscalchi, S. M., Li, J., and Lee, C.-H., "A Study on Lattice Rescoring with Knowledge Scores for Automatic Speech Recognition", Proc. of ICSLP, 2006.
- [8] Tanabe, K., "Penalized Logistic Regression Machines: New methods for statistical prediction 2", IBIS, 2001.
- [9] Birkenes, Ø., Matsui, T., and Tanabe, K., "Isolated-word Recognition with Penalized Logistic Regression Machines", Proc. of ICASSP, 2006.
- [10] Birkenes Ø., Matsui, T, Tanabe, K, and Myrvoll, T. A., "N-best rescoring for speech recognition using penalized regression machines with garbage class", Proc. of ICASSP, 2006.
- [11] 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.
- [12] Tanabe, K, Penalized logistic regression machines and related linear numerical algebra, In KOKYUROKU 1320, Institute for Mathematical Sciences, Kyoto, 2003.
- [13] Riedmiller, M. and Braun, H., "A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP algorithm", Proc. of the IEEE Intl. Conf. on Neural Networks, 1993.
- [14] Lee, K. F., and Hon, H. W., "Speaker-independent phone recognition using hidden Markov models," IEEE Trans. on Acoust., Speech and Signal Process., Vol. 37, No. 11, 1989.
- [15] Schwarz, P., Matějka, P., and Černocký, J., "Hierarchical structures of neural networks for phoneme recognition", Proc. of ICASSP, 2006.