N-BEST RESCORING FOR SPEECH RECOGNITION USING PENALIZED LOGISTIC REGRESSION MACHINES WITH GARBAGE CLASS

Øystein Birkenes[ab], Tomoko Matsui[a], Kunio Tanabe[c], and Tor André Myrvoll[ab]

[a]The Institute of Statistical Mathematics, Tokyo, Japan
[b]Department of Electronics and Telecommunications, NTNU, Trondheim, Norway
[c]Waseda University, Tokyo, Japan
{birkenes,myrvoll}@iet.ntnu.no, tmatsui@ism.ac.jp, tanabe.kunio@waseda.jp

ABSTRACT

State-of-the-art pattern recognition approaches like neural networks or kernel methods have only had limited success in speech recognition. The difficulties often encountered include the varying lengths of speech signals as well as how to deal with sequences of labels (e.g., digit strings) and unknown segmentation. In this paper we present a combined hidden Markov model (HMM) and penalized logistic regression machine (PLRM) approach to continuous speech recognition that can cope with both of these difficulties. The key ingredients of our approach are N-best rescoring and PLRM with garbage class. Experiments on the Aurora2 connected digits database show significant increase in recognition accuracy relative to a purely HMM-based system.

Index Terms— Speech Recognition, N-Best Rescoring, PLRM, Garbage Class, Aurora2

1. INTRODUCTION

Although classification approaches like neural networks or kernel methods achieve state-of-the-art performance in many pattern recognition tasks, they have had limited success in continuous speech recognition. There are at least two reasons for this. First, in continuous speech recognition, the goal is to predict a sequence of labels (e.g., digit strings or phoneme strings) without knowing the segment boundaries for the labels. Most existing neural networks or kernel methods classify samples using single labels only. Second, speech signals have varying lengths, while the majority of known neural networks or kernels operate on vectors. The hidden Markov model (HMM) framework deals effectively with both of these issues, but suffers from incorrect model assumptions.

In [1], the authors presented a hybrid HMM/SVM approach for continuous speech recognition with the support vector machine (SVM) using the N-best rescoring paradigm. Their method addressed the above issues and slightly outperformed the conventional HMM approach, but the method had several weaknesses. Since the problem of segments (phones) with varying lengths was solved by discarding all but a fixed number of feature vectors, much information in the speech signals were lost. Moreover, in the rescoring of the N-best lists, sentences with deletion and insertion errors could not be corrected.

Recently, a combined HMM/PLRM approach was proposed [2] for continuous speech recognition. As with [1], the procedure taken was to rescore N-best lists, but unlike [1], a penalized logistic regression machine (PLRM) [3, 4, 5] was used instead of SVM to obtain conditional probabilities of segment labels, without discarding any of the feature vectors. The PLRM directly models the conditional probabilities of segment labels, and is in that sense more suited for N-best rescoring than SVM. However, also this approach could only correct substitution errors, and not deletion and insertion errors.

In this paper, we present a combined HMM/PLRM approach that overcomes the problems with insertion and deletion errors reported in [1] and [2]. We do this by introducing a garbage class in PLRM. Experiments on the Aurora2 connected digits database demonstrates the power of this approach.

The paper is organized as follows. In the next section we review PLRM and introduce a garbage class with PLRM. In Sec. 3 we explain how PLRM can be used to obtain conditional probabilities of words given a segment, and in Sec. 4 we use these probabilities to rescore sentence hypotheses in the N-best lists. Section 5 describes experiments performed on the Aurora2 connected digits database, and finally, Sec. 6 contains the conclusions and a discussion on future work.

2. THE PENALIZED LOGISTIC REGRESSION MACHINE

Let $(x, y) \in \mathcal{X} \times \mathcal{Y}$ be a random pair drawn according to an unknown probability distribution $p(x, y)$. In classification, the label set $\mathcal{Y}$ is finite, and the goal is to find a mapping $h : \mathcal{X} \to \mathcal{Y}$ that gives good prediction on any feature $x \in \mathcal{X}$. Let K denote the number of classes and let each class be represented by an integer in the set $\mathcal{Y} = \{1, \ldots, K\}$.

This work was partly funded by the Scandinavia-Japan Sasakawa Foundation (SJSF) and the Research Council of Norway through the BRAGE project, which is a part of the language technology programme KUNSTI.

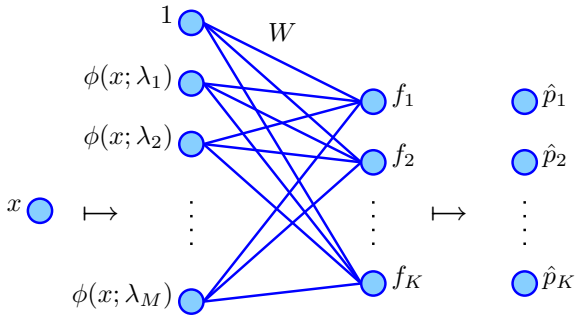

Fig. 1. The nonlinear logistic regression model in PLRM.

The penalized logistic regression machine (PLRM) [3, 4, 5] makes an estimate $\hat{p}(y|x)$ of the conditional probability distribution $p(y|x)$ based on a set of training examples $\mathcal{D} = \{(x^{(l)}, y^{(l)})\}_{l=1}^{L}$. Deterministic prediction on a given feature x is

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} \hat{p}(y|x). \tag{1}$$

The conditional distribution for each class k is modeled using a nonlinear logistic regression model with parameter matrix W with rows w_k, i.e.,

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

where $\boldsymbol{\phi}(x;\Lambda) = [1, \phi(x;\lambda_1), \ldots, \phi(x;\lambda_M)]^T$ is a vector of M regressors augmented by the scalar 1, with the set $\Lambda = \{\lambda_1, \ldots, \lambda_M\}$ denoting their hyperparameter vectors. The nonlinear logistic regression model in PLRM is illustrated in Fig. 1, where for simplicity we write $f_k = w_k^T \boldsymbol{\phi}(x;\Lambda)$.

Given a set of training data $\mathcal{D} = \{(x^{(l)}, y^{(l)})\}_{l=1}^{L}$, the parameter matrix W is estimated by minimizing

$$\mathcal{P}_\delta^{\log}(W;\mathcal{D}) = -\sum_{l=1}^{L} \log \hat{p}_{y^{(l)}} + \frac{\delta}{2} \operatorname{trace} \Gamma W \Sigma W^T, \tag{3}$$

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 kth diagonal element is the fraction of training samples with the kth class label, and $\Sigma = (1/L)\Phi\Phi^T$, where Φ is an $(M+1) \times L$ matrix with columns $\boldsymbol{\phi}(x^{(l)}, \Lambda)$. For the convex minimization of (3), see [5].

2.1. PLRM with Garbage Class

In some applications, the classifier will be presented with features x that do not correspond to any of the classes in the label set $\mathcal{Y}$. In this situation, the classifier should return a small probability for every class in $\mathcal{Y}$. However, this is made impossible by the fact that the total probability should sum to 1, that is, $\sum_{y \in \mathcal{Y}} p(y|x) = 1$. The solution to this problem is to introduce a new class with label $y = K+1 \in \mathcal{Y}_0 = \mathcal{Y} \cup \{K+1\}$, often referred to as a garbage class, that should get high conditional probability given features that are unlikely for the classes in $\mathcal{Y}$, and low probability otherwise.

In order to train such a garbage class, a set of features labeled with the garbage label, or garbage features, are needed. For applications with a low-dimensional feature set $\mathcal{X}$, these garbage features can be drawn from a uniform distribution over $\mathcal{X}$. For many practical applications however, $\mathcal{X}$ has a very high dimensionality, so an unreasonable high number of samples must be drawn from the uniform distribution in order to achieve reasonable performance. In such cases, prior knowledge of the nature or the generation of the possible garbage features is of great value. We will see in the next section how we can use N-best lists to generate garbage features for speech recognition.

3. PROBABILISTIC PREDICTION OF SPEECH SEGMENTS

Let $x = (x_1, \ldots, x_T)$ be a sequence of T feature vectors extracted from a speech segment, and let $y \in \mathcal{Y}$ be the class label of x, where $\mathcal{Y}$ is the vocabulary of subword labels (e.g., phonemes or digits), possibly augmented with a garbage label. In order to use PLRM for probabilistic prediction of speech segments, we need to define a mapping $\boldsymbol{\phi}$ which maps a segment x into a vector $\boldsymbol{\phi}(x;\Lambda)$. For this we use a set of M hidden Markov models (HMMs); one HMM for each of the subwords in the vocabulary, respectively, and possibly one HMM for the garbage class. We choose

$$x \mapsto \boldsymbol{\phi}(x;\Lambda) = [1, \phi(x;\lambda_1), \ldots, \phi(x;\lambda_M)]^T, \tag{4}$$

where $\phi(x;\lambda_m)$ is the frame-normalized log-likelihood of the mth HMM with parameter vector λ_m. To be more specific with our choice of nonlinear mapping, let $\lambda = (\pi, A, \eta)$ denote the parameters of an HMM; $\pi = [\pi_i]$ is the vector of initial state probabilities, $A = [a_{i,j}]$ is the transition probability matrix, and $\eta = \{\eta_i\}$ is the collection of the parameters of the state-conditional pdfs. Then[1]

$$\phi(x;\lambda) = \frac{1}{T} \log \max_q \pi_{q_1} p(x_1;\eta_{q_1}) \prod_{t=2}^{T} a_{q_{t-1},q_t} p(x_t;\eta_{q_t}). \tag{5}$$

where $q = (q_1, \ldots, q_T)$ is a state sequence.

To gain additional discriminative power, it was proposed in [6] to treat $\Lambda = \{\lambda_1, \ldots, \lambda_M\}$ as a parameter of the model (2) instead of just a hyperparameter of the nonlinear mapping $\boldsymbol{\phi}$. Thus, the parameters of the model are W and Λ, and we are interested in finding the pair (W^*, Λ^*) that minimizes the criterion in (3), i.e.,

$$(W^*, \Lambda^*) = \arg\min_{(W,\Lambda)} \mathcal{P}_\delta^{\log}(W, \Lambda; \mathcal{D}). \tag{6}$$

[1] This is actually an approximation to the frame-normalized log-likelihood of an HMM. Nevertheless, since this is a common approximation in the speech literature, we refer to this simply as the frame-normalized log-likelihood.

Although the function in (3) 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 [5]. As for the minimization with respect to Λ, there are many possibilities, two of which are the steepest descent method in [6], or the Rprop method [7] used in the experiments in this paper.

3.1. PLRM training for N-best Rescoring

An N-best list [8] is a list of the N most likely sentence hypotheses of a given utterance, and can be efficiently generated from a set of HMMs. The sentence hypotheses are ordered by their HMM likelihood, and each hypothesis is accompanied by a segmentation, which is the most likely segment boundaries given the sentence.

The role of PLRM in our N-best rescoring approach is to provide conditional probabilities of subwords given a segment. We choose to train a PLRM with garbage class, since many of the segments in the N-best lists do not contain a complete utterance of a subword. Some segments, for example, contain only a part of an utterance of a subword, or even an utterance of several subwords together. Hence, as mentioned in the previous section, two sets of training data have to be used; correct segments with the correct subword label, and incorrect segments with the garbage label.

For many training databases for speech, we do not know the segment boundaries for the data, only the orthographic transcription. Then, the most straightforward thing to do would be to estimate the segment boundaries. For this, we will make use of a set of subword HMMs to perform Viterbi forced alignment segmentation. Thus, from a pair (z, s), where z is a sequence of feature vectors of a sentence s with L_s subwords, we obtain a set $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(L_s)}, y^{(L_s)})\}$ of subword labeled segments. Doing this for all the pairs (z, s) in the training database gives a set

$$\mathcal{D}_{\text{correct}} = \{(x^{(l)}, y^{(l)})\}_{l=1,\ldots,L_{\text{correct}}} \tag{7}$$

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

$$\mathcal{D}_{\text{garbage}} = \{(x^{(l)}, K+1)\}_{l=L_{\text{correct}}+1,\ldots,L} \tag{8}$$

of all garbage-labeled segments.

The full training data used to train the PLRM is therefore

$$\mathcal{D} = \mathcal{D}_{\text{correct}} \cup \mathcal{D}_{\text{garbage}}. \tag{9}$$

4. N-BEST RESCORING WITH PLRM

In the previous section we explained how PLRM can be used to obtain the conditional probability of a subword given a segment. In this section we will see how we can use these probability estimates in continuous speech recognition by rescoring and reordering sentence hypotheses of an N-best list.

For a given sentence hypothesis $\hat{s} = (\hat{y}^{(1)}, \ldots, \hat{y}^{(L_{\hat{s}})})$ with corresponding segmentation $z = (x^{(1)}, \ldots, x^{(L_{\hat{s}})})$, 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:

$$\hat{p}_{\hat{s}} = \left(\prod_{l=1}^{L_{\hat{s}}} \hat{p}_{\hat{y}^{(l)}} \right)^{1/L_{\hat{s}}}. \tag{10}$$

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 (10) can be interpolated with the HMM likelihood. 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 conditional probabilities can then be written as

$$S_{\hat{s}} = (1-\alpha) \log \hat{p}_{\hat{s}} + \alpha \log \hat{p}(\hat{s}|z) \tag{11}$$

$$\propto (1-\alpha) \log \hat{p}_{\hat{s}} + \alpha (\log \hat{p}(z|\hat{s}) + \log \hat{p}(\hat{s})), \tag{12}$$

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

$$S_{\hat{s}} \propto (1-\alpha) \log \hat{p}_{\hat{s}} + \alpha \log \hat{p}(z|\hat{s}). \tag{13}$$

It should be noted that the score in the right side above, unlike the score in (10), cannot be interpreted as a probability, except for $\alpha = 0$, when the two scores are the same.

5. EXPERIMENTS

Experiments were conducted on the Aurora2 connected digits database. This is a database of utterances, from different speakers, of digit strings with lengths 1–7 digits. Training of HMMs and the PLRM were done using the 8440 utterances in the clean condition training set. As a test set, we chose the clean data of test set A, which consists of 4004 utterances. For the generation of the N-best lists, we used the set of HMMs that were defined in the training script distributed with the database. For PLRM training and rescoring, we had $K = 12$ classes including digit classes and silence, in addition to one garbage glass. For each class, we used an HMM with 16 states and 3 mixtures per state.

In the training of the PLRM we updated only the means of the HMMs while keeping the other HMM parameters fixed.

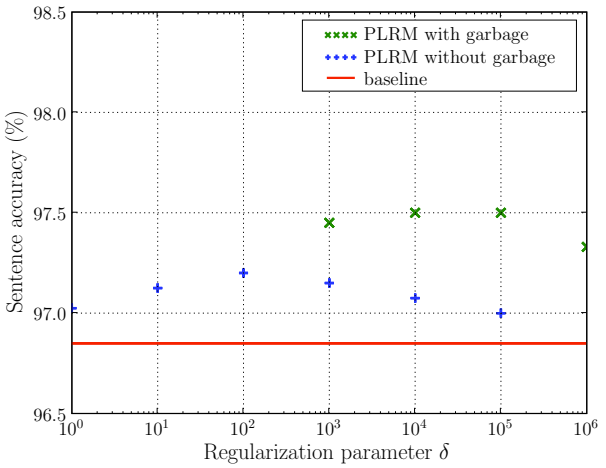

Fig. 2. Sentence accuracy on the test set for various δ.

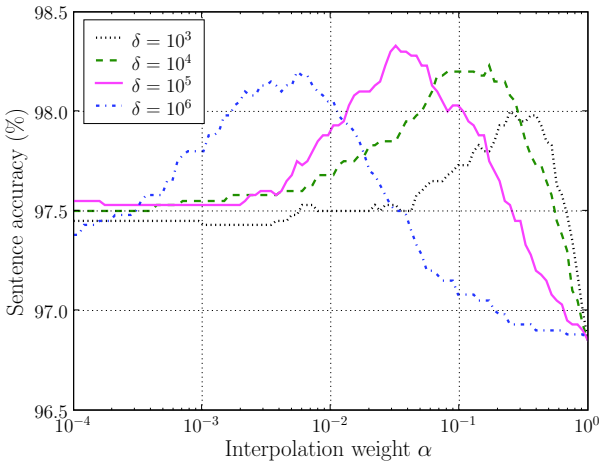

Fig. 3. Sentence accuracy using interpolated scores.

For each of the coordinate descent iterations we used the Rprop method [7] with 100 iterations to update the HMM means Λ and the Newton method with 4 iterations to update W. After 30 coordinate descent iterations, the optimization was stopped due to time limitations.

A 5-best list was used to extract the garbage segments from the training set, with $\epsilon = 10$. In the rescoring phase, we also used a 5-best list. The list accuracy, i.e., the sentence accuracy obtained with a perfect rescoring method, was 99.18%.

Figure 2 shows the sentence accuracy on the test set for our approach (PLRM with garbage), compared with the approach taken in [2] (PLRM without garbage), and the Aurora2 default recognition system (baseline). We see that our approach gives the best accuracy for the four values of the regularization parameter δ (see Eq. (3)) we used in our experiments. For lower values of δ, we expect a somewhat lower sentence accuracy due to over-fitting. Very large δ values are expected to degrade the accuracy since the regression likelihood will be gradually negligible compared to the penalty term.

Figure 3 shows the effect of interpolating the HMM score with the PLRM score as in (13). Note that with $\alpha = 0$, only the PLRM score is used in the rescoring, and when $\alpha = 1$, only the HMM score is used. The large gain in performance when taking both scores into account can be explained by the observation that the HMM score and the PLRM score made very different sets of errors.

6. CONCLUSIONS AND FUTURE WORK

We have presented a combined HMM/PLRM approach for continuous speech recognition. Our approach copes with the sequence label problem with unknown segmentation by the use of an N-best list. Moreover, we use HMM likelihoods as input to PLRM and thereby address the problem of speech signals with varying lengths. With the use of a garbage class in PLRM our approach does a good job in correcting deletion and insertion errors, in addition to substitution errors. The experiments show that the approach works well for a wide range of δ values, and particularly well when the sentence score obtained from PLRM is combined with the sentence likelihood from HMM.

We have not discussed how to find the optimal values of the hyperparameter δ and the interpolation weight α. This is a topic for future research.

7. REFERENCES

[1] A. Ganapathiraju, J. Hamaker, and J. Picone, "Applications of support vector machines to speech recognition," *IEEE Trans. Signal Processing*, vol. 52, no. 8, pp. 2348–2355, Aug 2004.

[2] Ø. Birkenes, T. Matsui, K. Tanabe, and T. A. Myrvoll, "Continuous speech recognition with penalized logistic regression machines," in *Proc. IEEE Nordic Signal Processing Symposium*, Reykjavik, Iceland, June 2006.

[3] K. Tanabe, "Penalized logistic regression machines: New methods for statistical prediction 1," *ISM Cooperative Research Report 143*, pp. 163–194, March 2001.

[4] K. Tanabe, "Penalized logistic regression machines: New methods for statistical prediction 2," in *Proc. IBIS*, Tokyo, Aug 2001, pp. 71–76.

[5] K. Tanabe, "Penalized logistic regression machines and related linear numerical algebra," in *KOKYUROKU 1320, Institute for Mathematical Sciences*, Kyoto, 2003, pp. 239–250.

[6] Ø. Birkenes, T. Matsui, and K. Tanabe, "Isolated-word recognition with penalized logistic regression machines," in *ICASSP*, Tolouse, France, May 2006.

[7] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The RPROP algorithm," in *Proc. of the IEEE Intl. Conf. on Neural Networks*, San Francisco, CA, 1993, pp. 586–591.

[8] R. Schwartz and Y.L. Chow, "The N-best algorithm: an efficient and exact procedure for finding the N most likely sentence hypotheses," in *ICASSP*, Albuquerque, New Mexico, USA, April 1990, vol. 1, pp. 81–84.