On the Effectiveness of Statistical Modeling based Template Matching Approach for Continuous Speech Recognition

Xie Sun, Xin Chen, and Yunxin Zhao

Department of Computer Science
University of Missouri, Columbia, MO 65211 USA

xs8pb@mail.missouri.edu, xck82@mail.missouri.edu, zhaoy@missouri.edu

Abstract

In this work, we validate the effectiveness of our recently proposed integrated template matching and statistical modeling approach on four baseline systems with increasing phone recognition accuracies in the range of 73% to 78% for the TIMIT task. The four baselines were generated using the methods of 1) Discriminative Training (DT) of Minimum Phone Error (MPE), 2) MFCC concatenated with ensemble Multiple Layer Perceptron (MFCC+EMLP) features, 3) DT combined with the MFCC+EMLP features, and 4) data sampling based ensemble acoustic models integrated with DT and MFCC+EMLP features. Experimental results obtained from template matching based rescoring on the phone lattices generated by the baseline models have shown that our template matching approach has produced consistent and significant improvements over the four baselines, and the highest recognition accuracy was 79.55% obtained from rescoring the phone lattices produced by the ensemble acoustic model baseline.

Index Terms: Template Matching, Discriminative Training, Multiple Layer Perceptron, Ensemble Acoustic Models

1. Introduction

Hidden Markov models (HMMs) have been the most successful approach in automatic speech recognition (ASR). However, the assumption that within each HMM state the observations are independent makes HMM ineffective for modeling the fine details of speech temporal evolutions that are important in characterizing nonstationary speech sounds. Attempts have been made to deal with this weakness in the HMM paradigm. One approach is segmental HMMs [1], where each state generates jointly a sequence of frames. In another approach, the continuity property of the trajectories described by the sequence vectors was exploited where dynamic features were used for smoothing the trajectory given by the mean values of the state sequence [2]. Templates offer another way to describe trajectories. A template consists of a sequence of feature vectors representing a speech segment. Template-based approaches don't need to make any explicit assumption about the data. They have attracted new efforts in recent years and the reported results are promising [3, 4, 5].

In the previous work [6], we proposed a framework of using Gaussian mixture models (GMMs) to label frame vectors to represent templates. The method was evaluated on three HMM baselines with different mixture sizes in GMMs for the TIMIT task with positive results. However, these three baselines had relatively low recognition accuracies, ranging from 70% to 72.5%. Since the templates were constructed using the baseline GMMs, we expect the template matching performance to be further improved when better baselines are generated. In our current work, we generated four different baselines for the TIMIT task by using the following four methods: 1) Discriminative Training (DT) of Minimum Phone Error (MPE), 2) MFCC concatenated with ensemble Multiple Layer Perceptron (MFCC+EMLP) features, 3) DT combined with the MFCC+EMLP features, and 4) data sampling based ensemble acoustic models integrated with DT and MFCC+EMLP features, where on the TIMIT full test set the phone recognition accuracies of the four baseline systems were 73.25%, 75.66%, 76.51% and 77.97%, respectively. We demonstrate that the statistical modeling base template matching approach can make the consistent and significant improvement over each baseline, and the best phone accuracy performance we obtained was 79.55% which is among the highly competitive results reported in the literature [7, 8] on the TIMIT task. We have investigated performing template matching on lattices with different sizes and we provide a discussion on how the different sizes of lattices affect recognition accuracies in the context of the baseline model quality.

In Section 2, the statistical modeling based template matching method is reviewed. In Section 3, we demonstrate the experimental results based on the TIMIT baselines generated by the four methods, and in Section 4, we discuss the experimental results. Finally, in Section 5, we give our conclusion and discuss future work.

2. Statistical modeling based template matching

We choose the template unit as context-dependent phone segments (triphone context). To construct the templates, we first carry out forced alignments of training speech data with their transcriptions to obtain phone boundaries which define the context-dependent phone templates. We then use the GMM codebook which consists of the GMMs $\{m_1, m_2, \ldots, m_N\}$ from the phonetic decision tree (PDT) tied triphone states in the baseline HMMs to label the template frames. To do so, we compute the likelihood scores of a frame x_t of a phone template by all GMMs and the GMMs that give the top n likelihood scores, $\mathrm{p}(x_t|m_{1(t)}) \geq p(x_t|m_{2(t)}) \geq \cdots \geq p(x_t|m_{n(t)}) \geq \cdots$, are used to label x_t. Each GMM index is also associated with a weight that is proportional to the likelihood score $p(x_t|m_{i(t)})$. A template frame is therefore represented as:

$$x_t \rightarrow \left\{ \begin{bmatrix} m_{1(t)} \\ \vdots \\ m_{n(t)} \end{bmatrix} \begin{bmatrix} w_{1(t)} \\ \vdots \\ w_{n(t)} \end{bmatrix} \right\}, \text{ with } \sum_{i=1}^{n} w_{i(t)} \text{ normalized to 1.}$$

For two sequences of feature vectors m and n, we use (1) to calculate their distance:

$$D(m,n) = \frac{\min \sum_{k=1}^{N} d(\emptyset_m(k), \emptyset_n(k))}{N} L \quad (1)$$

where d is the local distance of any two frame vectors in the two sequences, $\emptyset_m$ and $\emptyset_n$ are two functions that map m and n to the common time axis, N is the warping path length, and L is the length of the test speech sequence. Here we adopt a symmetric constraint defined by (2):

$$D_{i,j} = d_{i,j} + \min\{D_{i-1,j},\ D_{i-1,j-1},\ D_{i,j-1}\} \quad (2)$$

where $d_{i,j}$ is the local distance for frame vectors i and j, and $D_{i,j}$ is the cumulative distance from the first i frame vectors in m to the first j frame vectors in n.

The local distance in the template matching is based on a log likelihood ratio of the GMMs that are used to represent each frame. Suppose we need to compute the local distance

between a frame vector f labeled by two GMMs (m_1, m_2) of a test phone segment with a frame vector k labeled by two GMMs (n_1, n_2) of a phone template. The log likelihood ratio (LLR) measure for calculating the local distance between f and k is:

$$d(f,k) = \left|\log \frac{w_{11}p(f|m_1)+w_{12}p(f|m_2)}{w_{21}p(f|n_1)+w_{22}p(f|n_2)}\right| \quad (3)$$

where

$$w_{11} = \frac{p(f|m_1)}{p(f|m_1)+p(f|m_2)}, \quad w_{12} = 1 - w_{11}$$

$$w_{21} = \frac{p(k|n_1)}{p(k|n_1)+p(k|n_2)}, \quad w_{22} = 1 - w_{21} \quad (4)$$

Note that the numerator in the ratio of Eq.(3) is not a conventional likelihood since the weights of the GMMs are computed directly from the likelihood scores of the current frame vector, but the LLR measure thus defined satisfies $d(f,f) = 0$, a desired property for a dissimilarity measure. However, the LLR measure is not symmetric and does not satisfy the triangular inequality. For more details about the LLR measure, please refer to [6].

We use the phonetic decision tree (PDT) based triphone tying for template clustering. The three tying structures defined by the three emitting states of the corresponding phone HMMs are kept, and the multiple tying results are jointly used in template matching. Our template matching method is implemented for lattice rescoring. In matching a test speech segment with a triphone unit, we select the n-best templates from the corresponding tied triphone cluster that are closest to the test segment and use their average score as the match score. The architecture of the overall template matching method is described in Fig. 1.

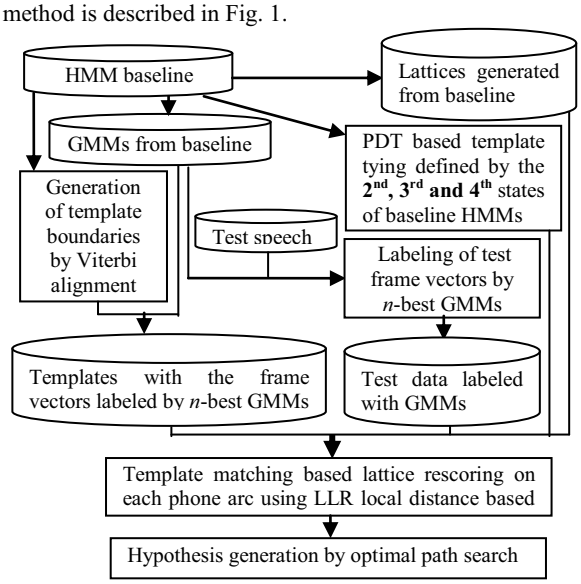

Fig.1: *Block diagram of the template matching approach*

3. Experiments

We performed phone recognition experiments on the TIMIT task. The training set had 3696 sentences from 462 speakers and the full test set included 1344 sentences spoken by 168 speakers. The commonly adopted phone set of 39 phones and a phone bi-gram language model (LM) were used. All HMMs included 16-component GMMs and crossword triphone models were used in decoding search. The total number of phone templates was 152715 and 5 GMM indices were used to label a frame vector. All templates were used to match a test phone segment, but only the top 3% of them were selected to calculate an averaged matching score for the segment. Phone lattices were generated for each test sentence by using HTK [9] with the baseline models. For each baseline approach, three lattice sets were generated by using the tokens n=*2, 3*, and *4* [9], with the average numbers of nodes per lattice in the order of 250, 850 and 1800, and the average numbers of arcs in the order of 450, 2350 and 6250, respectively, representing small, medium, and large lattices for the current TIMIT task. For convenience of subsequent discussions, we'll refer to the three lattice size scales by the three token sizes. Since phone lattices provided hypothesized phone boundaries, template matching was directly performed on the phone arcs in each lattice.

3.1 Template matching based on the DT baseline

Discriminatively trained models have been shown to significantly improve error rates, as they have more power to better differentiate between confusable sounds. Minimum Phone Error (MPE) based DT has been shown to improve HMM baseline for TIMIT in [11]. In this current work, MPE was used to train a HMM baseline, where 39-dimensional features defined by 13 MFCCs and their first and second time derivatives were used. We first used the maximum likelihood criteria to train a basic HMM baseline which had the recognition accuracy 71.86%. We then used the MPE criterion to train the discriminative models on top of the basic baseline HMMs. The discriminative models were obtained with 4 iterations of DT. The DT trained HMM baseline had the recognition accuracy 73.25% and the accuracy gained from the MPE training over the basic HMM baseline was 1.39% absolute. We used the discriminative HMMs as one of our new baselines, which were used to generate the phone lattices for rescoring. We extracted 1189 GMMs from the DT baseline model to label the frame vectors of templates.

In Fig. 2, we compare phone recognition performances by using the MPE based HMM baseline and template matching rescored results on three lattice sets which were generated with tokens n=*2, 3,* and *4*. Template matching made the largest improvement of 1.49% absolute over the MPE baseline on the smallest lattice size (token n=*2*). When the lattice size increased to n=*4*, even though template matching still made a 1.02% improvement over the baseline, compared with the case n=*2*, the gain decreased.

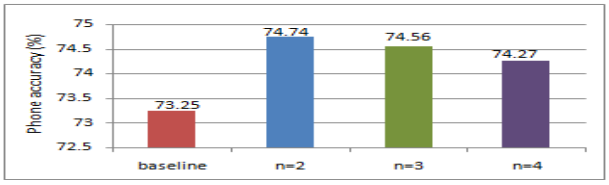

Fig. 2: *A comparison of phone accuracy (%) for MPE trained HMM baseline and template matching based lattice rescoring on three lattice sets with tokens n=2, 3, and 4*

3.2 Template matching based on the baseline generated by MFCC+ EMLP features

The concatenation of MLP features with traditional MFCC features has been proven effective in different tasks [10]. In [11], MFCC plus ensemble MLP (EMLP) features were used on TIMIT task and a significant improvement was obtained over a HMM baseline. In the current work, the MFCC+EMLP features were also used to generate HMMs as our baseline. 10-fold cross validation (CV) data sampling was used to build the MLP ensemble to generate the EMLP features. For each speech frame, PCA was used to reduce the EMLP feature dimension from 39 to 15, and then the reduced EMLP features were concatenated with the original 39 MFCC-based features to form a 54-dimensional feature vector. For more detail about the generation of the EMLP features, please refer to [11]. The

MFCC+EMLP feature based HMM baseline had the phone recognition accuracy of 75.66%, from which 1678 GMMs were extracted to generate templates. The MFCC+EMLP baseline models were used to generate the phone lattices for rescoring.

In Fig. 3, we compare phone recognition performances by using the MFCC+EMLP feature based HMM baseline and template matching based lattice rescoring on the three lattice sets. Template matching made the improvements of 1.37%, 1.61%, and 1.49% absolute over the MFCC+EMLP feature baseline for the three lattice sets with tokens *n=2*, *3*, and *4*, respectively. It's worth noticing that this time template matching made the most improvement over the HMM baseline when the lattice token *n=3*.

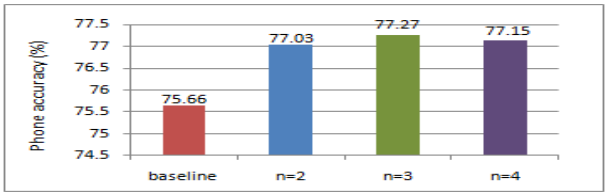

Fig. 3: *A comparison of phone accuracy (%) for the MFCC+EMLP feature based HMM baseline and template matching based lattice rescoring on three lattice sets with tokens n=2, 3, and 4*

3.3 Template matching based on the baseline generated by DT integrated with MFCC+EMLP features

We further combined the MPE based discriminative training with the MFCC+EMLP feature based HMMs. Using the MFCC+EMLP based HMMs as the initial models, MPE based discriminative training was then performed with 4 iterations to further optimize the model parameters. The newly trained MPE+MFCC+EMLP based models had the same number of HMM parameters as the MFCC+EMLP baseline HMMs, and the phone recognition accuracy of the new baseline was 76.51%. From the newly trained baseline HMMs, 1678 GMMs were extracted to generate templates, and three sets of phone lattices were generated.

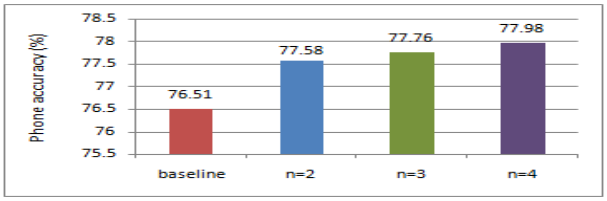

Fig. 4: *A comparison of phone accuracy (%) for MPE+MFCC+EMLP based HMM baseline and template matching based lattice rescoring on three lattice sets with tokens n=2, 3, and 4*

In Fig. 4, we compare phone recognition performances by using MPE+MFCC+EMLP based HMM baseline and the template matching based lattice rescoring on the three lattice sets. Template matching made the phone accuracy improvements of 1.07%, 1.25%, 1.47% absolute over the HMM baseline on the lattice sets with tokens *n=2*, *3*, and *4*, respectively. It's worth noting that this time template matching continued improving the baseline until the lattice token *n=4* (We also generated a lattice set with token *n=5* to evaluate template matching, but the performance gain became 1.31% which was smaller than the case of *n=4*).

3.4 Template matching based on the baseline generated by ensemble models integrated with DT and MFCC+EMLP features

Using data sampling approach to generate ensemble acoustic models was discussed in [11], where through data sampling, multiple training data sets were produced, and each sampled training data set was used to train one set of acoustic models in which each of them is called a base model. For an *N*-fold cross validation (CV) data sampling, a *(N-1)/N* fraction of training data is included in each sampled training set. Data sampling generated ensemble models usually make significant performance improvement over the single model trained using full training set, even though each individual base model in the ensemble models has lower performances [11]. In Section 4.3, we obtained the phone recognition accuracy of 76.51% for MPE+MFCC+EMLP based HMM baseline that was trained using the full training set. Here we applied a 10-fold CV data sampling to produce an ensemble of 10 base models. Each base model was trained by MPE+MFCC+EMLP. The phone recognition accuracy of the ensemble acoustic models was 77.97%, which made an improvement of 1.46% absolute over the baseline model in Section 4.3. We used the base models in the ensemble acoustic models to generate 10 lattice sets for each fixed token size *n*, and template matching was used individually on the lattices generated by each corresponding model of the 10 base models for rescoring. The average number of GMMs extracted from the 10 individual model sets to be used in template construction was 1558. In order to combine 10 different rescored lattices from these 10 individual base models, we first converted each lattice in the 10 lattice sets to the corresponding confusion network (CN) and then combined the 10 CNs to produce the final rescored recognition result [12].

In Fig. 5, we plot the recognition accuracies for the individual base models and the template matching rescored accuracy with the three lattice sets generated by the corresponding base model. We can see that template matching improved the accuracy of the individual models and on the lattice set with token *n=4*, template matching made the largest improvement. In Table 1, the phone recognition accuracies were averaged for the 10 base models and for the lattice rescoring results based on the 10 base models with the token size *n=2, 3*, and *4*, respectively. In Table 2, we show the phone recognition accuracy results for the baseline models of MPE+MFCC+EMLP that was trained using the full training set, the ensemble acoustic models, and the CN integration of lattice rescoring results on each of the three lattice sets. When token *n=4*, the lattice rescoring results based on the CN integration produced the best recognition accuracy of 79.55% which was a 1.58% absolute improvement over the ensemble acoustic model baseline.

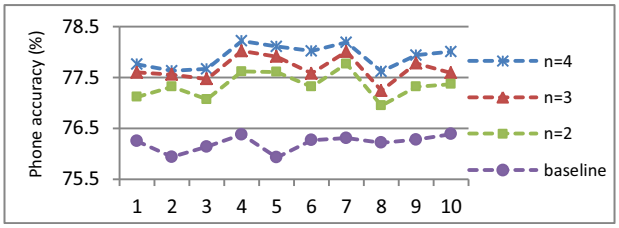

Fig. 5: *Recognition accuracies (%) for individual base models and template matching based rescored results on each base model with three lattice sets (token n=2, 3, and 4).*

Table 1. *Averaged recognition accuracies (%) for base models and lattice rescoring results on three lattice sets with tokens n=2, 3 and 4.*

Method	Base models	*n=2 (rescoring)*	*n=3 (rescoring)*	*n=4 (rescoring)*
Averaged accuracy	76.21	77.35	77.68	77.92

Table 2. *Recognition accuracies (%) for baseline models trained using the full training set, the ensemble acoustic models, and the CN integration of the lattice rescoring results on three lattice sets*

Method	Phone accuracy	
MPE+ MFCC+EMLP based single model	76.51	
MPE+MFCC+EMLP based ensemble models	77.97	
CN integration of template matching based lattice rescoring	*n =2*	78.89
	n=3	79.28
	n=4	**79.55**

4. Discussion

We summarize the accuracy performances of the different baselines and their best lattice rescoring results with the corresponding lattice size in Table 3. The template matching approach has made consistent and significant improvements over the HMM baselines with increasing recognition accuracies (For details of the significance test, please refer to [6]). In addition, lattice size determined by the number of tokens *n* for template matching increased when the quality of the baseline models improved. In Fig.6, we show the average rank of correct phone string references when they were inserted into the different lattice sets for the three methods of baseline generation. The MPE had the lowest phone baseline accuracy among these three methods, and template matching produced the best performance for the lattice set with token *n=2*. We also notice that when the token size for the lattices generated by MPE models increased from 2 to 4, the average rank of correct references significantly "decreased". For the method of MFCC+EMLP features, its baseline accuracy was better than MPE. When the lattice token increased from 2 to 3, the average rank of the correct references was stable. However, when the lattice token further increased from 3 to 4, the rank largely "decreased". In this case, template matching obtained the best recognition accuracy for the lattice size generated by token *n=3*. The method of MPE+MFCC+EMLE was the best baseline among these three methods, and template matching achieved the best recognition accuracy for the lattice size with token *n=4*. In this case, the average rank of the reference phone strings was almost unchanged when the lattice token increased from 2 to 4.

Table 3. *Summary of recognition accuracies (%) of the four baselines and the best lattice rescoring results with the corresponding lattice size*

Method	Baseline	Template Matching	# of tokens
MPE	73.25	74.74	2
MFCC+EMLP	75.66	77.27	3
MPE+MFCC+EMLP	76.51	77.96	4
MPE+MFCC+EMLP+ Ensemble models	77.97	79.55	4

In general, the phone recognition results of higher recognition accuracy should be closer to the correct references. When the lattice size increases, if the rank of correct references largely "decreased", then the possibility of picking up the correct hypotheses by rescoring becomes smaller, whereas if the rank of the correct references stays stable, then the new correct hypotheses provided by large lattices are more likely to be picked up by rescoring and better recognition accuracy can be achieved. Based on these observations and reasoning, we conclude that the template matching approach has a better capacity to pick up correct hypotheses from the large lattices when better baseline models were used. To support the analysis, in Table 4, we also provide the percentages of the inserted correct references being ranked the first in the TIMIT test data with the different lattice sets produced by the three token sizes.

Computation and storage overheads resulting from the proposed template matching based lattice rescoring as well as a method of generating template representatives to reduce the overheads were discussed in [6].

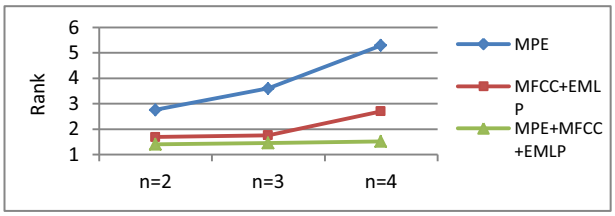

Fig. 6: *Average ranks of correct references in three lattice sets for the three methods of baseline generation*

Table 4. *Percentages of the correct references ranking the first in the TIMIT test data with three different lattice sizes*

method	*n=2*	*n=3*	*n=4*
MPE	84.2%	82.7%	78.6%
MFCC+EMLP	88.6%	88.2%	85.9%
MPE+MFCC+EMLP	89.8%	89.8%	89.5%

5. Conclusion

In this paper, we have validated the effectiveness of our statistical model based template matching approach on four different HMM baselines which were generated by MPE, MFCC+EMLP features, MPE+MFCC+EMLP, and MPE+MFCC+EMLP based ensemble acoustic model. The template matching approach used triphones instead of triphone states as a recognition unit so that it can capture better the dynamic information of speech than the traditional HMMs. Our experiments have shown positive gains in phone recognition accuracy on the TIMIT full test set, and our best performance of 79.55% is among the best reported results on the TIMIT continuous phoneme recognition task.

6. Acknowledgement

This work is supported in part by National Science Foundation under the grant award IIS – 0916639.

7. References

[1] Ostendorf, M., Digalakis, V. and Kimball, O.A., "From HMMs to segment models: a unified view of stochastic modeling for speech recognition," IEEE Trans. on SAP, Vol. 4, pp. 360-378, 1996.

[2] Gish, H. and Ng, K.., "Parametric trajectory models for speech recognition," Proc. of ICSLP, Vol. 1, pp. 466–469, 1996.

[3] De Wachter, M., Matton, M., Demuynck, K. and Wanbacq, P., "Template-based continuous speech recognition," IEEE Trans. On ASLP, Vol. 15, No.4, May 2007.

[4] Golipour, L. and O'Shaughnessy, D., "Phoneme classification and lattice rescoring based on a k-NN approach," in Proc. Interspeech, Sept. 2010, pp. 1954–1957.

[5] Demuynck, K., Seppi, D., Van Hamme, H., and Van Compernolle, D., "Progress in example based automatic speech recognition," in Proc. ICASSP, May 2011, pp. 4692-4695.

[6] Sun, X. and Zhao, Y., "Integrate template matching and statistical modeling for speech recognition," Proc. Interspeech, pp. 74-77, 2010.

[7] Sainath, T. N., Ramabhadran, B., and Picheny, M., "An Explortation of Large Vocabulary Tools for Small Vocabulary Phonetic Recognition," in *Proc. ASRU*, 2009.

[8] Mohamed, A., Dahl, G., and Hinton, G., "Acoustic Modeling using Deep Belief Networks," *IEEE Trans. on Audio, Speech and Language Processing.*, 2011.

[9] http://htk.eng.cam.ac.uk/.

[10] Zhu, Q., et al., "Using MLP features in SRI's conversational speech recognition system," in Proc. ICSLP, vol. 2, pp. 921–924,2005.

[11]Chen, X. and Zhao, Y., "Integrating MLP features and discriminative training in data sampling based ensemble acoustic modeling Data sampling based ensemble acoustic modeling," Proc. Interspeech, pp.1349-1352, 2010.

[12] Evermann, G. & Woodland, P. C. (2000*b*). Posterior probability decoding, confidence estimation and system combination. *Proceedings of the Speech Transcription Workshop*, College Park.