DISCRIMINATIVE TEMPLATE EXTRACTION FOR DIRECT MODELING

Shankar Shivappa *

University of California, San Diego
Dept. of Electrical and Computer Eng.
La Jolla, CA 92093, USA
sshivappa@ucsd.edu

Patrick Nguyen and Geoffrey Zweig

Microsoft Research
One Microsoft Way, Redmond, WA 98052, USA
{panguyen,gzweig}@microsoft.com

ABSTRACT

This paper addresses the problem of developing appropriate features for use in direct modeling approaches to speech recognition, such as those based on Maximum Entropy models or Segmental Conditional Random Fields. We propose a feature based on the detection of word-level templates which are discriminatively chosen based on a mutual information criterion. The templates for a word are derived directly from the MFCC feature vectors, based on self-similarity across examples. No pronunciation dictionary is used, and the resulting templates match closely to in-class examples and distantly to out-of-class examples. We utilize template detection events as input to a segmental CRF speech recognizer. We evaluate the entire scheme on a voice search task. The results show that the use of discriminative template based word detector streams improves the speech recognizer's performance over the baseline HMM results.

Index Terms— Discriminative Templates, Segmental Conditional Random Fields, Speech Recognition

1. INTRODUCTION

Direct modeling for speech recognition has received considerable attention in recent years. Direct models directly estimate the posterior distribution $P(w|x)$ of a sentence hypothesis w given the observation sequence x, unlike generative models which estimate the posterior probabilities indirectly by estimating $P(w)P(x|w)$. Among the key advantages of direct modeling approaches are that they are inherently discriminative, and provide a coherent framework to integrate a large number of possibly redundant features. The promise of such methods is that by adding sufficiently many informative features, we will eventually be able to recover the underlying word sequence.

In speech recognition, early examples of the direct modeling approach include maximum entropy Markov models (MEMMs)[1] and conditional random fields (CRFs) [2][3]. This past work is based on log-linear models, but still uses features defined at the conventional frame level. In later work, we have extended the direct modeling approach to use *segment* level features, both at the word and utterance level. In [4], we propose a flat direct model (FDM) which operates at the utterance level and uses word-level template-matching features. Two new classes of features based on phone and multi-phone detection were introduced in [5]. Our FDM approach was then generalized to continuous speech recognition with a Segmental CRF (SCARF) based speech recognizer described in [6] [7]. In the SCARF framework, a log-linear model is applied for each word in turn as decoding proceeds, and the features are based on the detection of units such as phones, phone-classes, or multi-phones [5].

*The author performed the work while at Microsoft Research

In this paper, we extend our previous work by presenting a method for extracting *discriminative* templates, and using them as the basis of detector streams in the SCARF framework. These templates are at the word level, and each one is explicitly designed to match closely to in-class examples and distantly to out-of-class examples. The use of discriminatively trained templates in speech recognition has of course been explored very early on [8]. However, this early work is restricted to whole-utterance models, and does not address the use of word units; thus it is difficult to generalize to a large vocabulary continuous speech recognition (LVCSR) task. In recent template-based work such as that of [9], the authors explore the use of a template based scheme in LVCSR, but do not address the issue of discriminative training.

The rest of the paper is organized as follows. In Section 2, we describe the process of extracting discriminative templates at the word level. In Section 3, we describe the detection process which uses the extracted templates to detect the presence/absence of words in an utterance. In Section 4, we describe the experimental setup to use the template based detector stream in a SCARF based speech recognition model and present the results on a voice search application. Finally in Section 5, we offer some concluding remarks.

2. FINDING DISCRIMINATIVE TEMPLATES

In this section we describe the steps involved in extracting discriminative acoustic templates. While we focus on word-level templates, extension to sub-word and multi-word units is also possible. Our approach builds on previous work [10] in which we presented a procedure to identify common audio portions between repeated utterances, based on the MFCC feature vectors alone. This is a maximum likelihood approach in which each frame in the second utterance is either explained as a noisy copy of some matched frames in the first utterance, or as having been drawn from a background model. This is illustrated in Figure 1. Both utterances are normalized to have zero mean and unit variance; the background model is thus a single Gaussian with those parameters. Matched frames are explained as being drawn from a Gaussian whose mean is that of the matched frames in the first utterance; the variance is the same as for the background model. Dynamic programming is then used to find the optimal segmentation such that the likelihood of explaining all the frames of the second utterance under this model is maximized. For details of the matching process, please refer to [10]. In the current framework, we extend this procedure to extract acoustic templates from a training pool of utterances that contain the word of interest. The key is that we obtain the matching of the frames in the first utterance to the frames in the second utterance. We now show how to exploit this matching process to measure the mutual information between frames and words, and thus to extract templates that have the best discrim-

inative properties in the training set. We stress that these templates are for arbitrary portions of utterances, and moreover the discriminative training process operates directly on the feature vectors.

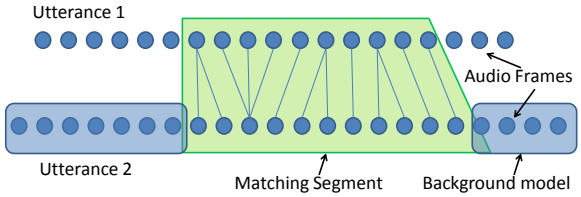

Fig. 1. The maximum likelihood approach to finding common audio segments between two utterances. Each frame of utterance 2 is explained either as a noisy copy of the matched frames in utterance 1, or as coming from a background model.

2.1. A simple non-discriminative approach

Consider the task of extracting templates for the word w_i. Let a set of utterances $X_i = \{x_{1i}, x_{2i} \dots x_{Ni}\}$ be drawn from the training set such that they contain the word w_i. We now describe a simple algorithm to extract templates from this training set:

- Take each utterance x_{ji} in X_i and match it with every other utterance in X_i.
- For each frame in x_{ji} count the number times it matched with a frame in the second utterance.
- If the frame receiving the maximum number of votes gets more than t_1 votes, it is chosen as a template center.
- The set of frames neighboring the template center that received more than t_2 votes are chosen as the template T_j.

We illustrate these steps in Figure 2.

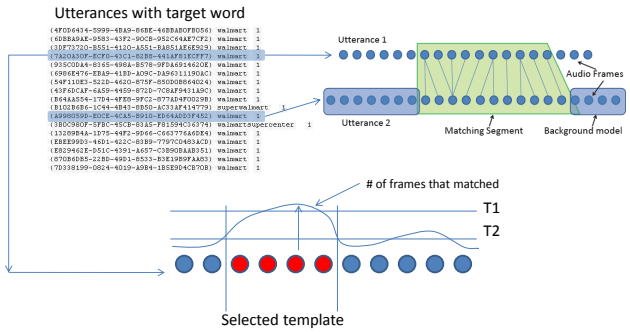

Fig. 2. A simple approach to extract templates based on picking frames that match well with other utterances in the training set.

This approach lets us extract those frames of the utterances that matched well with most of the other utterances and hence are good templates for representing word w_i. Note that this process does not require a prior segmentation of the utterance into words - because all the examples contain a particular word, those frames belonging to instances of that word will tend to accumulate the most matches. This is still the case when two utterances occasionally share another word. In the case that all utterances with the target word also occur with another word (e.g. target "Francisco" always occurring with "San"), the composite (e.g. "San Francisco") is expected to be extracted.

While this approach often produces reasonable sounding examples of a word, there is no reason to believe that the templates thus chosen are the best templates for distinguishing word w_i from other words in the vocabulary. The goal of the discriminative template extraction is to pick templates that match closely with word w_i while not matching well with other words in the vocabulary.

2.2. Mutual information based approach

In order to extract discriminative templates for word w_i, we will seek frames that are not only frequently matched when we have in-class utterances, but frames that are *infrequently* matched when we have out-of-class utterances. Let us again consider the set of utterances $X_i = \{x_{1i}, x_{2i} \dots x_{Ni}\}$ to be drawn from the training set such that that they contain the word w_i. We call this the set of in-class utterances, each of which with label $l = 1$. We also select a set $Y_i = \{y_{1i}, y_{2i} \dots y_{Ni}\}$ of utterances that do *not* contain the word w_i. These out-of-class utterances have the label $l = 0$. In order to select competitors that have portions which are confusable with a target word, we use n-best lists. First, we find words that occur on the n-best lists of utterances containing target word w_i. Then, we select utterances which have these competitor words in their transcripts, but not w_i. A random selection of utterances without w_i may also be used. The algorithm for finding discriminative templates is described below.

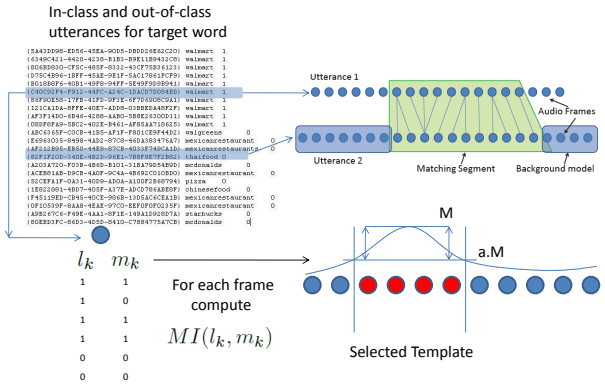

Fig. 3. A mutual information based approach to extract discriminative templates using in-class and out-of-class utterances in the training set.

- Match each utterance x_{ji} in X_i with every other utterance $z_k \in X_i \backslash x_{ij} \cup Y_i$.
- For each frame in x_{ij}, define the variable $m_k = 1$ if the frame matches some frame in utterance z_k. $m_k = 0$ otherwise. Let l_k denote the in-class/out-of-class label of z_k. Count the number of entries in the four distinct cases $(l_k, m_k) = \{(0,0), (0,1), (1,0), (1,1)\}$.
- Compute the mutual information between the variable m_k and labels l_k as in Equation 1.

$$MI(l_k, m_k) = \sum_{\substack{l_k \in \{0,1\} \\ m_k \in \{0,1\}}} P(l_k, m_k) log \left[\frac{P(l_k, m_k)}{P(l_k)P(m_k)} \right] \quad (1)$$

- The frame with the maximum mutual information is chosen as the template center.
- All contiguous frames in the neighborhood of the template center, having at least a fraction a times the maximum mutual information are included in the template.

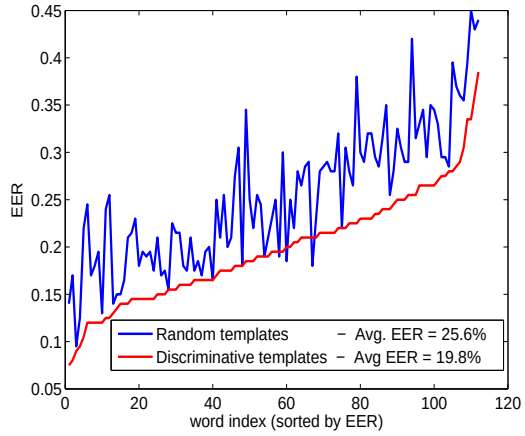

Fig. 4. Equal error rates for different words - comparing the discriminative templates and random templates.

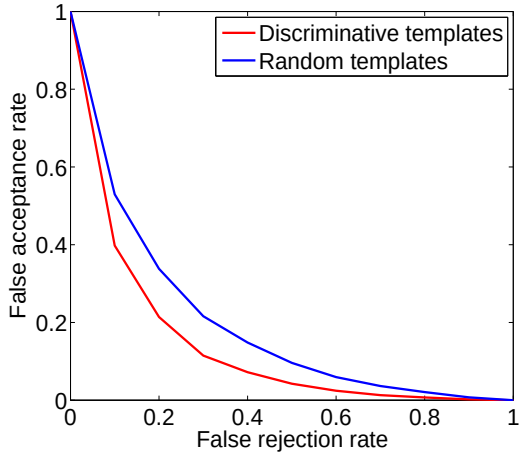

Fig. 5. Performance of the discriminative templates at different operating points, compared to that of the random templates.

$P(l_k)$, $P(m_k)$ and $P(l_k.m_k)$ can be estimated by counting the entries for$(l_k, m_k) = \{(0,0),(0,1),(1,0),(1,1)\}$. We illustrate these steps in Figure 3.

This selection process gives us the frames that have the maximum discriminative ability between the word w_i and the other words. Empirically, we have found that the selection process is also more robust to the actual value of the thresholds used, allowing us to use fixed values for all the words. This is due to the fact that the mutual information measure varies more smoothly over neighboring frames than the vote distribution.

2.3. Template ranking

In the previous section, we described a procedure for finding discriminative templates; in this section, we describe a method for selecting a small number of examples of each word for use in decoding. Given a set of candidate templates $T_i = \{t_{1i}, t_{2i} \ldots t_{Ni}\}$ for the word w_i, we proceed to rank them according to a measure of goodness that lets us pick those templates that can best separate the in-class utterances from the out-of-class utterances. To do this, we again turn to the principle of mutual information as explained below. We will use the fact that under the model described in [10], there is a difference in likelihood between explaining all the frames in an utterance z_k with the background model, and the likelihood when matches to the frames in a template t are allowed.

Number of templates	1	2	5	10	20	50
Avg. EER (%)	45	34	28	25	24	22

Table 1. Effect of using different number of templates in the detection process.

- Select a new set of in-class and out-of-class utterances X_i and Y_i for each word w_i as before. This is a tuning set.
- For each template t_{ji}, force it to match with each in-class and out-of-class utterance $z_k \in X_i \cup Y_i$. Let c_k denote the gain in likelihood from matching and l_k denote the label of z_k as before.
- Compute the mutual information between the variables c and l as follows.

$$MI(c,l) = \sum_k log\left(\frac{P(c_k|l_k)}{P(c_k)}\right)$$

$P(c)$, $P(c|l = 0)$ and $P(c|l = 1)$ are all modeled as Gaussian distributions and their mean and variances are estimated from the samples c_k. The templates are ranked according to the mutual information and the top n templates are chosen as the acoustic templates for word w_i.

3. TEMPLATE BASED DETECTOR STREAMS

Having defined a method to identify and rank discriminative templates, we now proceed to the problem of using the templates to detect the existence of a word in a test utterance. Specifically, given an utterance, the output will be a sequence of detected words and the audio frame indices where they were detected. Our detection process is based on the following thresholding scheme.

- Select a third set of in-class and out-of-class utterances X_i and Y_i for each word w_i.
- For each template t_{ji}, force it to match (as described in Section 2) with each in-class and out-of-class utterance $z_k \in X_i \cup Y_i$. Let $C_k = \max_j(c_j)$ be the maximum gain in likelihood gotten by using a template of word w_i to explain the frames of the utterance
- Compute the histogram of C_k for the in-class and out-of-class examples and choose a threshold for a given false acceptance rate or false rejection rate. Each word gets its own threshold.

Having fixed the thresholds for a given FA/FR rate, we proceed to detect the word in a test utterance as follows. Match the templates for a word with the utterance and find the maximum gain in likelihood of matching among the templates of w_i. If this gain is higher than the threshold for w_i, we detect the word w_i in the test utterance. This detector stream is incorporated in the SCARF speech recognition framework.

4. EXPERIMENTAL SETUP AND RESULTS

We evaluate the efficacy of our template extraction scheme on a voice search task. We extract templates for the most frequent 1000 words in the Windows Live Search for Mobile (WLS4M) dataset [11]. This data consists of recordings of users asking for business

SCARF features	Baseline HMM results	Oracle results on n-best	Cheat all words	Cheat top 1000 words	Discriminative templates
Existence	85.9	93.0	90.8	90.1	86.0
Expectation	85.9	93.0	91.2	91.1	86.4
Both	85.9	93.0	91.2	91.2	86.6

Table 2. Sentence accuracies using the template detector stream in the SCARF speech recognizer.

listings, for example, "Walmart Superstore" or "Honda Dealership." In this section, we first evaluate the performance of the templates in isolation from a speech recognition task, and then present the results in an actual recognition setup. For the initial experiments presented here, we used a small training set consisting of 45k utterances. The test set consists of instances of the most frequent business requests, and has 3623 utterances. In order to obtain the templates, we used 200 in-class and 200 out-of-class training examples at each stage of the template extraction process.

In our first performance measure, the discriminatively extracted templates are compared with templates that are randomly selected occurrences of word w_i (the randomly selected occurrences are based on a forced alignment of the transcription). One measure of template performance is the equal error rate (EER) between false detections and false rejections for word occurrences in the test set. This is illustrated in Figure 4 for a set of randomly selected words. We see that the equal error rate produced by our discriminative approach is systematically lower than when randomly selected word examples are used as templates, on a word-by-word basis. Averaged over all the 1000 words, the equal error rate of the discriminative templates is 5.8% better than the randomly selected templates. In Figure 5, we present the FA/FR rates for random and discriminative templates, averaged over all words. We see that the discriminative templates are uniformly better at all operating points. For a given FRR, the discriminative templates always have lower FAR than the random templates. In Table 1, we show the sensitivity of the detection process to the number of top templates retained for each word, averaged over all words. We see that beyond 10 templates, the gain in adding more templates per word is not significant. We thus restrict ourselves to 10 templates in the rest of the analysis.

Once we have selected the templates for the top 1000 words, we train a SCARF speech recognizer for the WLS4M task. We used the template detections in association with two types of features: existence and expectation. These features are defined in terms of a hypothesized segmentation of the observations into words. Each hypothesized word spans some block of observations. Existence features are of the form: Do a word and detector unit co-occur in a block? The expectation features model the Correct Accept/False Reject/False Accept of a detector unit in a word hypothesis. For example, if the phone "p" was expected in a block of observations which have been labeled with the word "red," the feature associated with the false accept of "p" would be true. For more details of the features used in the SCARF framework, please refer to [7]. The SCARF framework also uses a built-in HMM baseline feature so that we don't perform worse than the baseline.

In Table 2, we present the results from incorporating the template detector stream into the SCARF framework. The search strategy was guided by word occurrences in n-best lists produced by the baseline HMM decoding, so the oracle results on the n-best provide an upper bound on the possible improvement. In order to estimate the possible improvement by using template based detector streams in SCARF, we performed two cheating experiments. In the first experiment, we built a word detector that could detect all the words with 100% accuracy. In the second experiment, we assumed that our detector can detect only the top 1000 words accurately and does not detect any other word. We see that restricting ourselves to the top 1000 words does not affect our accuracy in this task. However, we see that these cheating experiment results still fall short of the oracle upper bound. This is due to the inconsistent annotation of homophones in the test data, e.g. Crown Plaza and Crowne Plaza. Having established the improvement possible in using the perfect template detector, we then proceed to evaluate our template detector stream on the test set. We see an improvement of 0.7% over the baseline.

5. CONCLUDING REMARKS

We have presented a discriminative acoustic template extraction scheme, and applied it to find examples of individual words. We use these templates in the SCARF based speech recognition model and show that we can improve over the baseline results. In the future one could explore the possibility of training templates at different levels (sub-word and multi-word) and for specific classes of words.

6. REFERENCES

[1] H-K. J. Kuo and Y. Gao, "Maximum Entropy Direct Models for Speech Recognition," in *Proc. of ASRU*, 2003.

[2] A. Gunawardana, M. Mahajan, A. Acero, and J. C. Platt, "Hidden Conditional Random Fields for Phone Classification," in *Interspeech*, 2005.

[3] J. Morris and E. Fosler-Lussier, "Discriminative Phonetic Recognition with Conditional Random Fields," in *HLT-NAACL*, 2006.

[4] G. Heigold, G. Zweig, X. Li, and P. Nguyen, "A Flat Direct Model for Speech Recognition," in *Proc. ICASSP*, 2009.

[5] G. Zweig and P. Nguyen, "Maximum Mutual Information Multiphone Units in Direct Modeling," in *Proc. Interspeech*, 2009.

[6] G. Zweig and P. Nguyen, "SCARF: A Segmental CRF Speech Recognition System," in *Microsoft research technical report*, 2009.

[7] G. Zweig and P. Nguyen, "A segmental CRF approach to large vocabulary continuous speech recognition," in *Proc. of ASRU*, 2009.

[8] P. C. Chang and B. H. Juang, "Discriminative template training for dynamic programming speech recognition," *ICASSP*, 1992.

[9] W. De Wachter, K. Demuynck, D. Van Compernolle, and P. Wambacq, "Data Driven Example Based Continuous Speech Recognition," in *Eurospeech*, 2003, pp. 1133–1136.

[10] G. Zweig, "New Methods for the Analysis of Repeated Utterances," in *Proc. Interspeech*, 2009.

[11] A. Acero, N. Bernstein, R. Chambers, Y. C. Ju, X. Li, J. Odell, P. Nguyen, O. Scholz, and G. Zweig, " Live Search for Mobile: Web Services by Voice on the Cellphone," in *Proc. ICASSP*, 2008.