Hidden Conditional Random Fields for Phone Recognition

Yun-Hsuan Sung [1] and Dan Jurafsky [2]

[1]*Electrical Engineering, Stanford University*
yhsung@stanford.edu

[2]*Linguistics, Stanford University*
jurafsky@stanford.edu

***Abstract*—We apply Hidden Conditional Random Fields (HCRFs) to the task of TIMIT phone recognition. HCRFs are discriminatively trained sequence models that augment conditional random fields with hidden states that are capable of representing subphones and mixture components. We extend HCRFs, which had previously only been applied to phone classification with known boundaries, to recognize continuous phone sequences. We use an N-best inference algorithm in both learning (to approximate all competitor phone sequences) and decoding (to marginalize over hidden states). Our monophone HCRFs achieve 28.3% phone error rate, outperforming maximum likelihood trained HMMs by 3.6%, maximum mutual information trained HMMs by 2.5%, and minimum phone error trained HMMs by 2.2%. We show that this win is partially due to HCRFs' ability to simultaneously optimize discriminative language models and acoustic models, a powerful property that has important implications for speech recognition.**

I. Introduction

Phone recognition is the task of mapping speech to phones without relying on a lexicon for phone-to-word mapping. It is widely used for investigating new acoustic models, and also plays a role in other tasks like real-time lip-syncing, in which lip shape is synchronized with the speech signal [10].

Currently, the most widely used probability models for acoustic modeling are Hidden Markov Models (HMMs). Researchers have trained HMMs both generatively by Maximum Likelihood (ML) and discriminatively by Maximum Mutual Information (MMI) [4] and Minimum Phone Error (MPE) [11]. Still, they make strong independence assumptions and suffer when facing non-independent features.

The conditional random field (CRF) [5] forms another family of sequence labeling models that is attractive as a potential replacement for HMMs. CRFs don't have strong independence assumptions and have the ability to incorporate a rich set of overlapping and non-independent features. In addition, CRFs are trained discriminatively by maximizing the conditional likelihood of the labels given the observations.

Recently, there has been increasing interest in applying CRFs to acoustic modeling. [8] used a hybrid MLP-CRF system for phone recognition. They used MLPs to extract phone posterior probabilities with phonological features as observations. They then trained a CRF to find the most probable phone sequences given the observations.

The MLP-CRF hybrid approach is powerful but requires independently training separate MLP classifiers in addition to the CRF. An alternative approach combines all the machine learning into a single model – the **hidden conditional random field** (**HCRF**) [2]. The conditional random field is augmented with hidden states that can model subphones and mixture components (much like a traditional HMM) with MFCCs rather than phonological features as the observations. [2] showed that HCRFs outperform both generatively and discriminatively trained HMMs on the task of phone classification. [15], [16] explored augmentations to this model such as discriminative methods for speaker adaptation in HCRFs. However, all of these previous HCRF models focused on the phone classification task, assuming known phone boundaries.

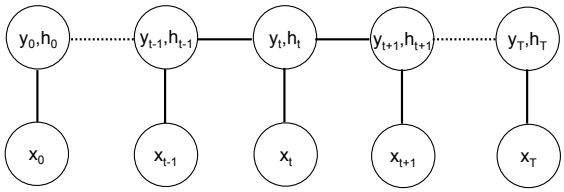

Fig. 1. Hidden Conditional Random Fields. x's are observations, y's are labels, and h's are hidden variables

In this paper, we extend HCRFs to the task of phone recognition – a task that does not assume that the phone boundaries are known in advance. We use the standard vector of 39 MFCCs as observations and supply various feature functions as inputs to the HCRFs. The hidden variables (subphone state sequences and mixture components in each state) are marginalized in learning and in inference.

We introduce HCRFs in section II, feature functions in section III, and learning and inference in section IV. We report our experiment results in sections V and discuss them in section VI.

II. Hidden Conditional Random Fields

An HCRF is a markov random field conditioned on designated evidence variables in which some of the variables are unobserved during training. The linear-chain HCRF used in speech recognition is a conditional distribution $p(Y|X)$ with a sequential structure (see Figure 1).

Assume that we are given a sequence of observations $X = (x_0, x_1, ..., x_T)$ and we would like to infer a sequence of corresponding labels $Y = (y_0, y_1, ..., y_T)$; conventional CRFs

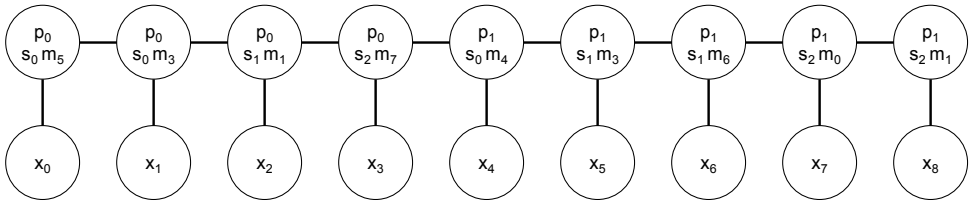

Fig. 2. An instance of a Viterbi labeling from an HCRF for phone recognition, showing a phone sequence (p_0, p_1) composed of a state sequence $s_0, s_0, s_1, s_2, s_0, s_1, s_1, s_2, s_2$ together with mixture components. s's and m's are hidden variables and are marginalized out in learning and inference.

model the conditional probability distribution function as:

$$p(Y|X) = \frac{1}{Z(X)} \exp\{\sum_t \lambda^T F(Y, X, t)\} \quad (1)$$

where F is the feature vector, a function of the label sequence Y and the input observation sequence X. λ is the parameter vector whose k^{th} element weights the k^{th} element in the feature vector F. The normalizing constant Z is called the *partition function* and is defined as:

$$Z(X) = \sum_{Y'} \exp\{\sum_t \lambda^T F(Y', X, t)\} \quad (2)$$

Summing over all possible instances of Y, this normalizing partition function contributes most of the computational expense within learning.

The cleanest way to apply CRFs to phone recognition would be to have the sequence of labels $Y = (y_0, y_1, ..., y_T)$ correspond to a series of phones and the sequence of observations $X = (x_0, x_1, ..., x_T)$ to a series of MFCC vectors. However, two aspects of variation in phone realization cripple this model.

First, the spectral and energy characteristics of a phone vary dramatically as it is uttered. Following conventional HMM systems, we capture this non-homogeneity by modeling each phone as a sequence of 3 sub-phones (*states*). Thus our model can use different parameters to describe the characteristics of the beginning, middle, and end of each phone.

Second, the acoustic realization of a phone differs widely across contexts. To accommodate such variation it is common in speech recognition systems to use both context-dependent phones and to use multiple mixture components. In our current experiments we are using only monophone CRFs, but introduce multiple components within each state to help address the problem of variation. Each component has different parameters to capture the different characteristic of the variable phone.

In summary, to capture surface variation, we introduce two kinds of hidden variables into the standard CRF model in the HCRF model. These hidden variables come from the states (subphones) s of each phone and the mixture components m for each state. It will be convenient to talk about both these variables as a single hidden variable h, which is the pair of states and components $h_t = (s_t, m_t)$. Figure 2 shows one instance of hidden variables, states $S = (s_0, s_1, ..., s_T)$ and components $M = (m_0, m_1, ...m_T)$. In both inference and learning tasks, these hidden variables are marginalized out.

For a sequence of hidden variables $H = (h_0, h_1, ..., h_T)$. HCRFs model the conditional distribution function as:

$$p(Y|X) = \frac{1}{Z(X)} \sum_H \exp\{\sum_t \lambda^T F(Y, H, X, t)\} \quad (3)$$

where the feature vector F is a function of the label sequence Y, the hidden variable sequence H, and the input observation sequence X. Thus the difference between (1) and (3) is that (3) marginalizes out the hidden variables. That is, to compute the most probable phone sequence, one would sum over all sequences of states (subphones) and mixture components. In this study, we marginalize these state and component variables in both inference and learning.

The constant partition function Z becomes:

$$Z(X) = \sum_{Y'} \sum_H \exp\{\sum_t \lambda^T F(Y', H, X, t)\} \quad (4)$$

Note that our models extend the original HCRF introduced in [2] and [15], [16], in which the label y was a single phone value rather than a phone sequence Y.

III. Feature Functions

The observations X extracted from the speech signal are 39-dimensional MFCCs. In speech recognition MFCCs are referred to as *features*. But in graphical models like HCRFs we reserve the word *features* or *feature functions* to mean the weighted functions of the input and output in the model. To distinguish 'MFCC features' from 'feature functions', we will refer to MFCCs as *observations* instead of features.

In this study, we used two different kinds of feature functions (See Figure 3). Transition feature functions have the hidden variables and labels at time $t-1$ and t as arguments. State feature functions have the observations and the hidden variables, labels at time t as arguments.

The transition feature functions we use in this study are bigram and state transition functions.

$$f^{(Bi)}_{yy'} = \delta(y_{t-1} = y, y_t = y') \qquad \forall y, y'$$
$$f^{(Tr)}_{yss'} = \delta(y_t = y, s_{t-1} = s, s_t = s') \qquad \forall y, s, s'$$

where $\delta(.)$ is the indicator function, the y's are phone labels, and the s's are the (subphone) states in each phone. The bigram features capture the transition between each phone pair, i.e., the state transition between the end of each phone to the start of the next phone. The state transition captures the state transition within each phone.

The state feature functions are the component occurrence, the first moment (observation value itself), and the second

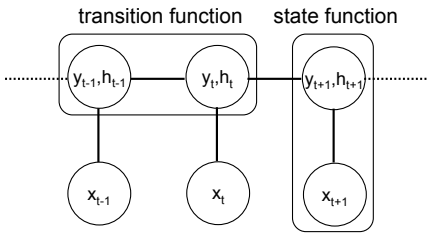

Fig. 3. Transition feature functions and state feature functions.

moment (squared values).

$$f_{s,m}^{(Occ)} = \delta(s_t = s, m_t = m) \qquad \forall s, m$$
$$f_{s,m}^{(M1)} = \delta(s_t = s, m_t = m)x_t \qquad \forall s, m$$
$$f_{s,m}^{(M2)} = \delta(s_t = s, m_t = m)x_t^2 \qquad \forall s, m$$

where s's stand for states in each phone. m's stand for components in each state. The component occurrence is the indicator function for each specific component. The first moment feature is used for the value of the observations themselves. The second moment feature is used for the square values of observations. The second moment feature functions are important because in most cases, the models need to emphasize some regions in the real value axis which can't be done only via the first moment features alone.

IV. Algorithms

The two main tasks in phone recognition are learning and inference (decoding). In both decoding and learning, we marginalize out the hidden variables (since we want the most likely phone sequence, not the most likely state sequence), using an N-best inference algorithm.

A. Decoding: N-best Inference

We begin with the task of decoding, producing a sequence of phones Y from a sequence of input observations X. In this phone recognition task, we don't actually want the Viterbi path through the HCRF. This is because the single best path includes a single particular sequence of states and mixtures.

Thus we need to find the best phone sequence, summing over states and mixture components. Rather than do this in a single pass, we do this by first generating the N-best sequences, and then run the forward algorithm over these to generate the total probability of the phone sequences.

In learning, we also need to find the top N probable phone sequences. The HCRF, like other discriminative models, e.g. MMI HMMs, needs to normalize the probability of the best phone sequence by the sum of all other sequences (the partition function). Because it is exponentially expensive to find all possible phone sequences, we use only the top N most probable phone sequences as an approximation.

The application to learning will be discussed in section IV-B; we begin here with the N-best decoding algorithm.

The exact sentence-dependent algorithm is an efficient algorithm for finding the N most likely phone sequences [1]. Its forward phase maintains for each cell of the search (each state i at each time point t) all possible phone sequences leading into it. Any two state sequences with the same phone sequences are merged into one hypothesis. This algorithm guarantees finding the exact N best phone sequences but is very time consuming. Instead, we use an approximate algorithm called phone-dependent N-best [13].

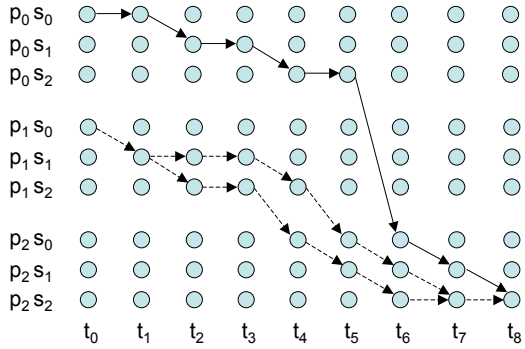

Fig. 4. The phone-dependent N-best algorithm. The two dashed paths from p_1 to p_2 are merged because they both have the same previous phone $p1$ despite having different state sequences.

The phone-dependent N-best algorithm assumes that the start time for a phone depends only on the preceding phone and not the whole preceding phone sequence (like the markov-one assumption used in a bigram language model). This requires storing a single path for each possible previous phone in each cell (state i and time t) of the search (see Figure 4). State sequences with the same previous phone are merged into one hypothesis. The computation becomes more efficient without losing too much accuracy [13].

In decoding, we applied the phone-dependent N-best algorithm by using HCRF models to get a N-best list. The forward algorithm is then used to rescore the conditional probability of each phone sequence in the N-best list. The decoder then returns the sequence with the highest probability, following equation (5) where L is the N-best list.

$$\tilde{Y} = \arg\max_{Y \in L} p(Y|X) \tag{5}$$

We found that $N = 10$ gives better performance than $N = 1$, i.e., finding the most probable phone sequence directly.

B. Learning

When training HCRFs, we want to maximize the conditional probability of the label sequence Y given the observation sequence X. To simplify calculation, we maximize the log-conditional distribution instead of equation (3) directly. The objective function for optimization becomes

$$\log p(Y|X) = \log \sum_H \exp\{\sum_t \lambda^T F(Y, H, X, t)\} - \log \sum_{Y'} \sum_H \exp\{\sum_t \lambda^T F(Y', H, X, t)\} \tag{6}$$

Equation (6) requires summing over all possible phone sequences in the second term. We approximated this equation by using the phone-dependent N-best algorithm to find the N

best phone sequences. (A lattice would be a natural extension.) Equation (6) then becomes

$$\log p(Y|X) \approx \log \sum_{H} \exp\{\sum_{t} \lambda^T F(Y, H, X, t)\} - \log \sum_{Y' \in L} \sum_{H} \exp\{\sum_{t} \lambda^T F(Y', H, X, t)\} \quad (7)$$

where L is the N-best list.

The learning problem is formulated as an unconstrained optimization problem. We use Stochastic Gradient Descent (SGD) for optimization, following [2] and [15]. The corresponding gradient with respect to λ_k used in SGD can be derived as follows

$$\frac{\partial \log p(Y|X;\lambda)}{\partial \lambda_k} = \sum_{H} f_k(Y, H, X, t) p(H|Y, X) - \sum_{Y' \in L} \sum_{H} f_k(Y', H, X, t) p(Y', H|X) = E_{H|Y,X}[f_k(Y, H, X, t)] - E_{Y',H|X}[f_k(Y', H, X, t)] \quad (8)$$

When a local maximum is reached, the gradient equals zero. As equation (8) shows, the expectation of features under the distribution of hidden variables given the label and observation variables is equal to the expectation of features under the distribution of hidden and label given observation variables. [17] gives the corresponding derivation for CRF training. In CRFs, the empirical count of the features is equal to the expectation of features given the model distribution when the maximum is achieved. We can get the same result if we remove the hidden variables H from equation (8).

For initialization, we followed [2], [15] in using HMM parameters as a starting point. Extending these previous models (which used ML HMMs), in the current work we trained MPE HMMs and transformed the parameters to give initializations for the transition, component occurrence, first moment, and second moment parameters in the corresponding HCRFs.

V. Experiments

A. Corpus

We used the TIMIT acoustic-phonetic continuous speech corpus [6] as our data set in this study. We mapped the 61 TIMIT phones into 48 phones for model training. The phone set was further collapsed from 48 phones to 39 phones for evaluation, replicating the method in [7] for comparison.

The training set in the TIMIT corpus contains 462 speakers and 3696 utterances. We used the core test set defined in TIMIT as our main test set (24 speakers and 192 utterances). The randomly selected 50 speakers (400 utterances) in the remaining test set are used as a development set for tuning parameters for all four models. All speaker dependent utterances are removed from training, development, and test sets.

B. Methods

We extracted the standard 12 MFCC features and log energy with their deltas and double deltas to form 39-dimensional observations. The window size and hopping time are 25ms and 10ms, respectively. We applied a Hamming window with pre-emphasis coefficient 0.97. The number of filterbank channels is 40 and the number of cepstral filters is 12.

TABLE I
PHONE ERROR RATES ON TIMIT CORE TEST SET OF GENERATIVELY AND DISCRIMINATIVELY TRAINED HMMS AND HCRFS.

Comps	ML HMMs	MMI HMMs	MPE HMMs	HCRFs
8	35.9%	33.3%	32.1%	29.4%
16	33.5%	32.1%	31.2%	28.7%
32	31.6%	30.8%	30.5%	**28.3%**
64	31.1%	30.9%	31.0%	29.1%

TABLE II
PHONE ERROR RATES ON TIMIT CORE TEST SET OF ML INITIALIZED AND MPE INITIALIZED HCRFS.

Comps	ML-initialized	MPE-initialized
4	31.2%	30.5%
8	30.6%	29.4%
16	29.7%	28.7%
32	29.0%	28.3%

We first trained ML HMMs and used them to train MMI and MPE HMMs with the HTK toolkit. I-smoothing 100 and a learning rate parameter 2 are used in MMI and MPE HMM training. A bigram phone language model was trained by maximum likelihood for all HMM systems. (A trigram phone language model would be a natural extension.) Finally, the MPE HMMs and the ML phone language model are used as initialization for HCRF training.

In HCRF learning, 10-best phone sequences are found by the phone-dependent N-best algorithm for each utterance. In each of the 300 to 3,000 passes, 10 different utterances are randomly selected from the training data to update all parameters. The random selection makes parameter update more frequent than using the whole training data and makes the training converge faster. Parameters are averaged over all passes to reduce variance. The final models are selected based on the development set.

C. Results

In our first study, we compare the generatively (ML) and discriminatively (MMI & MPE) trained HMMs with HCRFs. All results are presented in accuracy, the edit distance between the reference and the hypothesis (including insertion, deletion, and substitution errors). As Table I shows, MMI and MPE HMMs are consistently better than ML HMMs for all numbers of components. Our HCRFs outperform MMI and MPE HMMs further. The best result is 28.3% with 32 components. All three discriminative training models degrade due to overfitting with 64 components.

In our second study, we compare different initialization methods for HCRF training (see Table II). Because the log conditional probability is not convex, finding better local optima is important for HCRF training. Starting from MPE HMMs consistently gives about a 1% absolute improvement over starting from ML HMMs.

TABLE III
DELETION, SUBSTITUTION, AND INSERTION ERRORS FOR ALL FOUR MODELS WITH 32 COMPONENTS

Error	ML HMMs	MMI HMMs	MPE HMMs	HCRFs
Deletion	9.5%	8.1%	8.8%	7.2%
Substitution	19.1%	18.2%	17.8%	17.5%
Insertion	3.0%	4.5%	3.9%	3.6%
Total	31.6%	30.8%	30.5%	28.3%

D. Error Analysis

In Table I, ML HMMs continue to improve performance when the number of components increases up to 64. However, MMI HMMs, MPE HMMs, and HCRFs degrade as the number of component increases from 32 to 64. This is not surprising given the limited TIMIT training data, since discriminative training methods in general require more training data than generative training methods [9].

We show insertion, deletion, and substitution errors in Table III. The numbers are tuned by minimizing the phone error rate by using a held-out development data set. All three discriminatively trained models, MMI HMMs, MPE HMMs, and HCRFs, have lower deletion and substitution errors but higher insertion errors than generatively trained ML HMMs. Of the three discriminatively trained models, HCRFs have the lowest errors on all three different kinds of errors. The discriminative methods tend to have more balanced deletion and insertion errors, because they try to distinguish the corret phone sequences from other sequences in the N-best list.

VI. DISCUSSION

A. MMI HMMs vs HCRFs

HMMs have been shown to be equivalent to HCRFs with carefully designed feature functions [3]. By equivalence, we mean that for each HCRF parameter set, there exists a parameter set for HMMs that gives the same conditional probability. And while HCRFs are trained by maximizing conditional probability, as are MMI HMMs, the performance of our HCRFs is better than that of MMI HMMs. An analysis of this improvement might indicate whether HCRF training could be used as an alternative to extended Baum Welch training in HMMs. We thus summarize two main reasons for better performance below.

First, optimizing the conditional probabilities of HCRFs is an unconstrained optimization problem. The problem is easy to solve and Stochastic Gradient Descent updates the parameters more frequently than batch mode methods. The Extended Baum Welch algorithm is used for MMI HMM training because it is a constrained optimization problem. Instead of optimizing the conditional probability directly, which is difficult, it tries to find an auxiliary function which is easy to optimize. This suggests that HCRF training might be a better method than Extended Baum Welch for HMM training.

Second, our HCRF training simultaneously optimizes the acoustic model and the (phone bigram) language model, unlike traditional ASR systems that train the acoustic and language models separately. This is because the language model is encoded as transition feature functions (shown in Figure 3), and optimized simultaneously with the state feature functions. This joint discriminative training improved the performance of our model, as shown in Table IV. Training acoustic parameters directly results in a phone error rate of 29.3% (i.e., a 1.2% error rate reduction when compared to 30.5% achieved by the initial MPE HMMs). Training acoustic and language parameters simultaneously achieves 28.3%, i.e., an additional 1.0% reduction.

TABLE IV
SIMULTANEOUSLY TRAINING THE ACOUSTIC AND LANGUAGE MODEL PARAMETERS IMPROVES PERFORMANCE OVER JUST TRAINING THE ACOUSTIC MODEL PARAMETERS.

Error	Initial (MPE HMM)	HCRF Acoustic only	HCRF Acoustic & LM
Deletion	8.8%	8.8%	7.2%
Substitution	17.8%	17.9%	17.5%
Insertion	3.9%	2.6%	3.6%
Total	30.5%	29.3%	28.3%

We hypothesize that the discriminatively trained language model improves recognition by learning to distinguish phones that are particularly confusable in the baseline acoustic model. To test this hypothesis, we extracted from the phone confusion matrix the pairs of phones that were most often confused by our initial (MPE HMM) models (i.e., the models before HCRF/discriminative language model training). The top 10 most confused phone pairs are shown in Table V; the set is not surprising, as pairs like cl/vcl, er/r, m/n, and ih/ix/ax are acoustically quite similar.

We then investigated whether the language model probabilities were adjusted by our discriminative model in order to help distinguish these difficult phones. We examined the transition probability for every pair of phones in the initial (MPE HMM) model, and saw how the probability changed in the final HCRF model. Table VI shows the phone transitions whose probability increased the most after discriminative training. As our hypothesis predicts, all the phone transitions that gain a lot of probability mass after discriminative language model training involve at least one phone in those confusing phone pairs (cl/vcl, er/r, m/n, ih/ix/ax).

B. Large-margin HMMs vs HCRFs

Our best result (28.3%) is also competitive with the large-margin HMMs (28.2%) in [14] for the same task. Large-margin HMMs combine the idea of margin-based training with HMMs, and were shown to be better than MMI and MCE trained HMMs. We think the large margin idea can be also applied to our HCRFs to improve performance further.

C. The MLP-CRF hybrid vs HCRFs

Another advantage of HCRFs over HMMs that we don't explore in this paper is the ease of adding new parameters and corresponding feature functions in HCRFs. For example, [8] show how to add phonological features as features functions in a (non-hidden) CRF. Since phonologically features generally change within each phone, incorporating the [8] phonological features into HCRFs (which allow hidden state changes

TABLE V
THE 10 PAIRS OF PHONES MOST LIKELY TO BE CONFUSED IN OUR BASELINE MPE-HMM SYSTEM.

Pairs	Counts	Pairs	Counts
cl/vcl	1266	**ih/ix**	967
ax/ix	833	s/z	700
er/r	501	eh/ih	437
m/n	386	ix/iy	379
ae/eh	360	d/t	274

TABLE VI
THE 20 PHONE TRANSITIONS WHOSE PROBABILITY INCREASED THE MOST AFTER DISCRIMINATIVE LANGUAGE MODEL TRAINING.

Phone transitions	Log prob difference	Phone transitions	Log prob difference
r uw	0.722	eh **r**	0.660
er **ix**	0.657	**ix** l	0.628
vcl t	0.610	r **ax**	0.591
y **ix**	0.589	**er ax**	0.577
ax n	0.571	**n** d	0.568
aa **r**	0.568	**ax r**	0.546
m cl	0.542	**cl** b	0.536
n cl	0.531	l **ih**	0.531
m vcl	0.527	v **vcl**	0.502
vcl k	0.500	f **cl**	0.497

within each phone) should give an improvement over these conventional CRFs.

D. Time complexity

Since HCRF systems use the same sufficient statistics of data as HMM systems do, all four systems have similar decoding time. In training, however, the three discriminative models use either a lattice or N-best list and so require significantly more computational time than the generative model. Furthermore, two reasons make HCRF training more time consuming than MMI and MPE HMM training.

First, HCRFs simultaneously train acoustic and language models. In phone recognition, since the number of parameters in language models is smaller than that in acoustic models, the difference between HCRFs and MMI or MPE HMMs is small. But if HCRFs are extended to word recognition, the number of language model parameters would increase significantly, and training time for HCRFs would become an issue.

Second, we decode new N-best lists at each pass of HCRF training. By contrast, for MMI and MPE training, lattices are generated once with initial models and used for all iterations. It is possible that implementing lattices in HCRF models would add enough hypotheses to avoid regenerating at each pass.

VII. CONCLUSION

In this paper, we applied Hidden Conditional Random Fields to monophone recognition in the TIMIT corpus. Extending previous work on HCRFs for phone classification with known boundaries, we show that HCRFs work significantly better than both generatively and discriminatively trained HMMs, resulting in a 28.3% phone error rate with the same feature sets. We show how to use N-best inference in both learning and decoding, and give some analytic results on the differences between HMM and HCRF models. We also show that the ability of HCRFs to simultaneously optimize the acoustic model and language model parameters allows the training of a discriminative language model, which results in a 1% absolute gain in phone error rate.

In current work we are investigating richer features that can take advantage of the ability of HCRFs to incorporate new global features not possible in HMMs. Some obvious next directions are to extend monophone HCRFs to triphone HCRFs in order to compare to state of the art phone recognizers and to apply our training method as an alternative to Extended Baum Welch for training MMI HMMs, based on the equivalence of HCRFs and HMMs [3].

ACKNOWLEDGMENT

Thanks to Asela Gunawardana at Microsoft Research and Valentin Spitkovsky at Stanford for helpful comments and the anonymous reviewers for their suggestions. This work was supported by the ONR (MURI award N000140510388).

REFERENCES

[1] Y. L. Chow and R. Schwartz, "The N-Best Algorithm: An Efficient Procedure for Finding Top N Sentence Hypotheses", in the DARPA Speech and Natural Language Workshop, 81–84, 1990.

[2] A. Gunawardana and M. Mahajan and A. Acero and J. C. Platt., "Hidden Conditional Random Fields for Phone Classification", in Interspeech, 1117–1120, 2005.

[3] G. Heigold, P. Lehnen, R. Schluter, H. Ney, "On the Equivalence of Guassian and Log-Linear HMMs", in Interspeech, 273–276, 2008.

[4] S. Kapadia, V. Valtchev, and S. J. Young, "MMI training for continuous phoneme recognition on the TIMIT database", in ICASSP, 491–494, 1993.

[5] J. Lafferty, A. McCallum, and F. Pereia, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data", in ICML, 282–289, 2001.

[6] L. Lamel, R. Kassel, and S. Seneff, "Speech database development: Design an analysis of the acoustic-phonetic corpus", in the DARPA Speech Recognition Workshop, 1986.

[7] K. F. Lee and H. W. Hon, "Speaker independent Phone Recognition Using Hidden Markov Models", in ICASSP, 1641–1648, 1989.

[8] J. Morris and E. Fosler-Lussier, "Conditional Random Fields for Integrating Local Discriminative Classifiers", in IEEE Transactions on Audio, Speech, and Language Processing 617–628, 2008.

[9] A. Y. Ng and M. Jordan, "On Discriminative vs. Generative Classifiers: A comparison of logistic regression and Naive Bayes" in NIPS 14, 2002.

[10] J. Park and H. Ko "Real-Time Continuous Phoneme Recognition System Using Class-Dependent Tied-Mixture HMM With HBT Structure for Speech-Driven Lip-Sync" in IEEE Transactions on Multimedia 10, 7 1299–1306, 2008.

[11] D. Povey and P.C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training" in ICASSP, 105–108, 2002

[12] S. Sarawagi and W. W. Cohen, "Semi-Markov conditional random fields for information extraction", in NIPS 17, 1185–1192, 2004.

[13] R. Schwartz and S. Austin, "A comparison of several approximate algorithms for finding multiple (N-Best) sentence hypothesis", in ICASSP, 1993

[14] F. Sha and L. K. Saul, "Comparison of Large Margin Training to Other Discriminative Methods for Phonetic Recognition by Hidden Markov Models", in ICASSP, 2007.

[15] Y. H. Sung, C. Boulis, C. Manning, and D. Jurafsky, "Regularization, adaptation, and non-independent features improve hidden conditional random fields for phone classification", in IEEE ASRU, 347–352, 2007.

[16] Y. H. Sung, C. Boulis, and D. Jurafsky, "Maximum Conditional Likelihood Linear Regression and Maximum a Posteriori for Hidden Conditional Random Fields Speaker Adaptation", in ICASSP, 2008.

[17] C. Sutton and A. McCallum, "An Introduction to Conditional Random Fields for Relational Learning", in Introduction to Statistical Relational Learning, 2006.