SPEECH RECOGNITION WITH SEGMENTAL CONDITIONAL RANDOM FIELDS: A SUMMARY OF THE JHU CLSP 2010 SUMMER WORKSHOP

G. Zweig[1], P. Nguyen[1], D. Van Compernolle[2], K. Demuynck[2], L. Atlas[3], P. Clark[3], G. Sell[4], M. Wang[5], F. Sha[5], H. Hermansky[6], D. Karakos[6], A. Jansen[7], S. Thomas[6] S. G.S.V.S.[6], S. Bowman[8], J. Kao[4]

[1]Microsoft Research [2]Katholieke Universiteit Leuven [3]Universtity of Washington [4]Stanford University
[5]University of Southern California [6]Johns Hopkins University [7]JHU HLT COE [8]University of Chicago

ABSTRACT

This paper summarizes the 2010 CLSP Summer Workshop on speech recognition at Johns Hopkins University. The key theme of the workshop was to improve on state-of-the-art speech recognition systems by using Segmental Conditional Random Fields (SCRFs) to integrate multiple types of information. This approach uses a state-of-the-art baseline as a springboard from which to add a suite of novel features including ones derived from acoustic templates, deep neural net phoneme detections, duration models, modulation features, and whole word point-process models. The SCRF framework is able to appropriately weight these different information sources to produce significant gains on both the Broadcast News and Wall Street Journal tasks.

***Index Terms*—** Segmental Conditional Random Field, CRF, Speech Recognition

1. INTRODUCTION

Novel techniques in speech recognition are often hampered by the long road that must be followed to turn them into fully functional systems capable of competing with the state-of-the-art. In this work, we explore the use of Segmental Conditional Random Fields as an integrating technology which can augment the best conventional systems with information from novel scientific approaches.

The Segmental CRF approach [1] is a modeling technique in which the probability of a word sequence $\mathbf{w}$ is estimated from observations $\mathbf{o}$ as $P(\mathbf{w}|\mathbf{o})$ using a log-linear model. Described in Sec. 2, the model determines the probability of a word sequence by weighting features which each measure some form of consistency between a hypothesis and the underlying audio. These features are at the word-segment level, for example a feature might be the similarity between observed and expected formant tracks.

The key characteristic of the SCRF approach is that it provides a principled yet flexible way to integrate multiple information sources: all feature weights are learned jointly, using the conditional maximum likelihood (CML) objective function. In particular, SCRFs can combine information

- of different types, for example both real valued and binary features;
- at different granularities, for example at the frame, phoneme or word level
- of varying quality, for example from a state-of-the-art baseline and from less accurate phoneme or word detectors
- of varying degrees of completeness, for example a feature that detects just one word
- that may be redundant, for example from phoneme and syllable detectors

This flexibility is hard to achieve in standard systems, and opens new possibilities for the integration of novel information sources. The recently released SCARF toolkit [2] is designed to support research in this area, and was used at the workshop.

Over the course of the workshop we exploited several information sources to improve performance on Broadcast News and Wall Street Journal tasks, including:

- Template matching [3]
- Neural-net phoneme detectors, both MLP based [4, 5] and Deep Neural Nets [6]
- Word detectors based on Point Process Models [7]
- Modulation feature [8, 9] based multiphone detectors
- Duration models

In the remainder of the paper, we first summarize the SCRF model, then describe these information sources and their results in isolation, and finally present experimental results combining multiple information sources.

2. SEGMENTAL CRFS

A segmental CRF model extends the original CRF formulation [10] by applying the markov assumption at the segment rather than frame level, thus enabling the use of long-span features. Mathematically, it differs from [11] in that the segmentation is unknown during training, resulting in a non-convex objective function, and differs from variants of [12] in that the set of features is not pre-defined. The model is illustrated in Fig. 1. It is a two layer model, with states that represent words in the top layer, and observations in the bottom layer. All possible segmentations of the observations into words are considered in training and decoding; this figure shows one particular way of segmenting seven observations into three words. For a given segmentation, feature functions are defined which measure the consistency between the word hypothesis and the observations. SCARF is designed to work with word level features, as well as with observations that consist of the detection of acoustic units or events, most commonly phoneme or syllable detections. This is the meaning of the observations in Fig. 1, and critical to the automatically defined features in Sec. 3.

2.1. Model Definition

Denote by $\mathbf{q}$ a segmentation of the observation sequences, for example in Fig. 1 where $|\mathbf{q}| = 3$. The segmentation induces a set of (horizontal) edges between the states, referred to below as $e \in \mathbf{q}$. One such edge is labeled e in Fig. 1 and connects the state to its left, s_l^e, to the state on its right, s_r^e. Further, for any given edge e, let $o(e)$ be the segment associated with the right-hand state s_r^e, as illustrated in Fig. 1. With this notation, we represent all features as $f_k(s_l^e, s_r^e, o(e))$. The conditional probability of a state sequence $\mathbf{s}$ given an observation sequence $\mathbf{o}$ for a SCRF is then given by

$$P(\mathbf{s}|\mathbf{o}) = \frac{\sum_{\mathbf{q}\ s.t.\ |\mathbf{q}|=|\mathbf{s}|} \exp(\sum_{e\in\mathbf{q},k} \lambda_k f_k(s_l^e, s_r^e, o(e)))}{\sum_{\mathbf{s}'} \sum_{\mathbf{q}\ s.t.\ |\mathbf{q}|=|\mathbf{s}'|} \exp(\sum_{e\in\mathbf{q},k} \lambda_k f_k(s_l'^e, s_r'^e, o(e)))}.$$

Workshop supported by NSF grant IIS-0833652, with supplemental funding from Google Research, Microsoft, and the JHU HLT Center of Excellence. F.S. and M.W. supported by NSF and DARPA under grant and contract numbers NSF 0957742 and DARPA N10AP20019. D.V.C and K.D. supported by FWO travel grant K.2.105.10N, FWO research grant G.0260.07, and the EU MC-RT Network "Sound-to-Sense." L.A. thanks AFOSR grant FA9550-09-1-0060.

978-1-4577-0539-7/11/$26.00 ©2011 IEEE

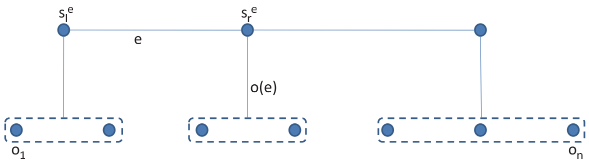

Fig. 1. A Segmental CRF.

Training is done by gradient descent with a conditional maximum likelihood objective function, and a description of the update equations may be found in [1].

The model above is quite general; to adapt SCRFs to large vocabulary continuous speech recognition, its states are made to refer to the states in an underlying finite state language model, for example a trigram LM. Then, transitions from one state to another correspond to changing states in this language model. This has two important consequences: first, we are not constrained to bigram language models, and second, the LM score can easily be used as a feature and its weight learned jointly with the others. As a second adaptation to LVCSR, we use lattices generated by a conventional system to constrain the set of segmentations that are considered.

3. FEATURES

SCARF automatically creates a wide range of features that relate the detections present in an observation stream to the words being hypothesized. These are briefly summarized below. In addition, the lattices can be annotated with user-defined word-level features.

3.1. Expectation Features

Expectation features are defined with reference to a dictionary that specifies the spelling of each word in terms of the units. The expectation features are:

- correct-accept of unit u: u is expected on the basis of the dictionary, and it exists in the span
- false-reject of u: u is expected but not observed
- false-accept of u: u is not expected and it is observed

Expectation features may further be defined on n-grams of units.

3.2. Levenshtein Features

Levenshtein features are computed by aligning the observed unit sequence in a hypothesized span with that expected based on the dictionary entry for the word. Based on this alignment, the following features are extracted:

- the number of times unit u is correctly matched
- the number of times u in the pronunciation is substituted
- the number of times u is deleted from the pronunciation
- the number of times u is inserted

3.3. Existence Features

Whereas Expectation and Levenshtein features are defined with reference to a dictionary, Existence features indicate the simple association between a unit in a detection stream, and a hypothesized word. An existence feature is present for each unit/word combination seen in the training data, and indicates whether the unit is seen within the hypothesized word's span. N-grams of units may be used as well.

3.4. Baseline Feature

To ensure that baseline performance can be achieved, SCARF implements a simple feature which requires only the one-best sequence of a baseline system. The baseline feature for a segment is always either $+1$ or -1. It is $+1$ when a hypothesized segment spans exactly one baseline word, and the label of the segment matches the baseline word. Otherwise it is -1. The contribution of the baseline feature to a hypothesis score will be maximized when the hypothesis exactly matches the baseline. Thus, by assigning a high enough weight to the baseline feature, baseline performance is guaranteed. In practice, the baseline weighting is learned and its value will depend on the relative power of the additional features.

4. DATASETS

In subsequent sections, we will describe the information sources we used. In order to present the relevant experimental results as we go, we now to describe our datasets.

4.1. Wall Street Journal

The Wall Street Journal database was used for training and testing both template and phone detector features. Results are presented on the nov92 20k open vocabulary test set using the default trigram LM. Training is done on the SI-284 data from WSJ0+1 comprising 81 hours from 284 speakers. The dictionary used was CMUdict 0.6d. SPRAAK [13] was used to create a conventional HMM system using Mel Spectra, postprocessed by mutual information discriminant analysis, VTLN and CMS as features. The acoustic models use a shared pool of 32k Gaussians and 5875 cross-word context-dependent tied triphone states. This results in a baseline 7.3% WER. One phone detector was derived from the baseline system by decoding with a phoneme LM; three others from variations on the baseline setup with a different preprocessing or size of Gaussian pool.

4.2. Broadcast News

Broadcast News data was used to test all features except those based on templates (due to computational requirements). The acoustic model is based on that of [14] and trained on about 430 hours of HUB4 and TDT4 data. A 4-gram language model was trained with about 400M words from HUB4 and Newswire data. The development data consisted of the NIST dev04f set (22k words), and the test set was the NIST RT04f data (50k words).

The acoustic modeling included LDA+MLLT, VTLN, fMLLR based SAT training, fMMI and mMMI discriminative training, and MLLR. After training, decoding with the IBM Attila decoder [14] produced lattices and the baseline feature of Sec. 3.4. A separate system was trained at Microsoft Research using just the HUB4 transcribed data, and had a WER about 4% absolute higher. Decoding with this system produced a word detector stream, and the lattices were annotated with feature values derived in the same way as the baseline features. In subsequent tables, SCARF1 refers to using SCARF with just two features: the baseline feature and the language model score.

5. TEMPLATE FEATURES

The work on template features started from a system operating according to the principles described in [15], i.e. the baseline HMM system generates word graphs enriched with phone segmentations after which each word arc score is replaced with the sum of the corresponding context-dependent phone template scores.

A first set of improvements done at the workshop was related to the DTW implementation and included: (i) score adjustment in function of duration, (ii) using the K=5-best average score instead of single best, (iii) assigning a local sensitivity matrix (diagonal covariance) to each test frame instead of to each reference frame as was done in [3], (iv) using soft template boundaries, and (v) adding context-dependent word templates.

Setup	WER
initial template system	9.6%
improved template system	8.2
SCARF+meta information	7.6
+HMM baseline system	6.9
+phone detectors	**6.6**

Table 1. Results using template features in WSJ.

All of the above fit within the framework of a single best Viterbi decoding strategy. However, the template based matching gives us a wealth of meta-information about the top-N templates for each segment which can not trivially be incorporated in single best decoding, but which can be harvested by the SCARF framework. The most important features measure (i) if the phone templates used originate from the hypothesized word, (ii) if word initial and final phone templates are used for hypothesized word initial and final phones, (iii) the speaker entropy and (iv) the average warping factors.

Two companion papers describe the above listed additions in detail. All parameter settings were optimized by means of the SCARF toolkit on the dev92 development data. Table 1 summarizes the results of the intermediate template based systems and the final results after combining the template system with the baseline (7.3% WER) HMM system and four phone detector streams. We see a 19% relative improvement over the initial template system, and 9.6% relative improvement over our best previous HMM system.

6. NEURAL NET FEATURES

6.1. Multi-layer Perceptrons

Phoneme posterior probabilities estimated using Multi-Layer Perceptrons (MLPs) are extensively used both as features and scores. In the SCARF framework, we explore a new application of these posteriors as phonetic event detectors for speech recognition. In this approach, we use two MLPs in a hierarchical fashion to estimate phoneme posterior probabilities. The first MLP transforms acoustic features with a context of 9 frames to regular posterior probabilities. The second MLP is trained in turn on posterior outputs from the first MLP. By using a context of 11 frames, we allow the second MLP to learn temporal patterns in the posterior features. These patterns include phonetic confusions at the output of the first MLP as well as the phonotactics of the language. The enhanced posteriors at the output of the second MLP are finally used as emission probabilities of a hybrid HMM-ANN phoneme decoder to produce a phoneme sequence for use as as a detector stream in SCARF.

For our experiments we trained the MLP networks using a 2-fold cross validation on 400 hours of broadcast news. The input features were short-term spectral envelope (FDLP-S) and modulation frequency features (FDLP-M) derived from sub-band temporal envelopes of speech [4] along with conventional PLP features. We have also used sparse PLP features, derived from a sparse auto-associative neural network trained on 35 hours of speech [5]. While the first MLP in the hierarchy is trained using 8000 hidden nodes, the second uses a much simpler network with 800 hidden nodes. Both the networks use an output phoneset of 42 phones. Table 2 shows the effectiveness of these detectors, along with the deep net detectors of the next section, when used as the sole source of acoustic information. In these experiments we removed the baseline feature and used Levenshtein features along with order 2 Expectation and Existence features. In the context of SCARF1 + MSR word detectors, we observe an overall improvement from 15.3% to 15.1%.

6.2. Deep Neural Nets

Deep neural nets (DNNs) are similar to MLPs in architecture. However, the parameters of DNN are trained very differently. In particular, there is an initial phase of unsupervised learning in which DNNs are built *layer by layer* with weights trained to maximally improve the likelihood of the unlabeled training data. This is achieved in our case by stacking several one-hidden-layer restricted Boltzman machines (RBMs)[6]. The primary goal of this phase is to learn good initial network parameters from data. Once all layers are learnt, the second phase of supervised learning is invoked. At this phase, the labels of training data are used to fine-tune the previously learned weights with error back-propagation. To explore the utility of DNNs for large-vocabulary speech recognition, we used them to create phoneme detectors for SCARF, integrating their posteriors in the same way as the outputs of MLPs. Our DNNs have 3 hidden layers, with 2048 hidden units each and one 132-unit softmax output layer for multiway classification at the phoneme sub-state level. In the unsupervised learning phase, we used contrastive divergence, a RBM-style parameter learning technique. In the supervised learning phase, we use stochastic gradient descent.

Acoustic Input	PER	WER
None	-	17.9
PLP	32.5%	17.2
PLP-Sparse	31.0	17.3
FDLP-S	31.1	17.0
FDLP-M	28.9	16.9
DNN-20hrs.	28.8	17.1
DNN-40hrs.	28.2	17.0

Table 2. Phoneme detectors as the acoustic model for dev04f.

Table 2 shows the effectiveness of the DNN phoneme detectors when used as the sole source of acoustic information. These results are comparable to the MLP results, though we note that the DNN detectors had the benefit of using fMMI features as input. In combination with SCARF1 and MSR word detectors, the both MLP and DNN detections produce 0.1 to 0.2% improvement on dev04f.

7. POINT PROCESS MODELS

A discriminatively trained variant of the point process model (PPM) described in [7] was used to construct whole word classifiers for 72 common error producing words. PPMs provide a means to explicitly model the temporal patterns of acoustic or phonetic events present when a word or syllable is produced. The MLP-based phonetic posteriorgrams described in Sec. 6.1 were used to define phonetic events (local maxima exceeding posterior probability of 0.5) that provided our input representation. Critical to our success, positive and negative training examples of each word were extracted directly from Attila BN lattice competitors, allowing the PPMs to focus on fine grain distinctions between the correct and incorrect baseline hypotheses.

Using the PPM classifier scores, we defined a SCARF lattice annotation feature stream. Table 3 shows the dev04f word error rates for the SCARF1 baseline with and without our PPM features using both a unigram and trigram language model. Interestingly, we see that the PPM annotation features achieve 0.7% of the 0.9% gain observed when moving from a unigram to a trigram language model, while using only within-arc acoustics and unigram statistics. Combined with the trigram LM, the PPM annotations provided 0.2% absolute improvement over SCARF1; including the MSR word detector stream, we observed an additive 0.3% improvement (Table 5).

8. MODULATION BASED LATTICE ANNOTATION

Demodulation extracts slowly-varying envelope waveforms of speech subband signals to obtain a multidimensional vector time-series expansion. We used two new methods of demodulation, coherent [8] and convex [9], as alternatives to the conventional

Setup	Unigram LM	Trigram LM
SCARF1	16.9%	16.0
+PPM Features	16.2	15.8

Table 3. Effect of PPM features as a function of LM for dev04f.

Setup	WER
SCARF1+MSR	15.3%
+Word-duration features	15.2
+Pre/post pausal features	15.1
SCARF1+MSR+Word-confusion features	15.0

Table 4. Effect of duration models on dev04f.

Hilbert envelope that underlies the mel-frequency cepstral coefficient (MFCC) representation. Coherent and convex demodulation each enforce bandwidth constraints across speech frames on an approximately syllabic time scale, and offer a new source of acoustic information that is potentially complementary to detectors derived from MFCCs. To take advantage of the long-term coherent and convex demodulation features, we trained maximum entropy 1 vs. all word classifiers using demodulation features as input, and annotated the lattices with word probability scores. Using a unigram language model and no baseline feature, we observe about 0.4% absolute improvement from both sources of modulation information.

9. DURATION MODELS

A segmental model is well suited to the application of duration models, and a companion paper describes our duration modeling approaches in detail. Briefly, three types of features were used:

1. Word duration features. For a given word hypothesis w with a hypothesized length l, we add features $P_c(l|w)$ and $P_i(l|w)$ where P_c and P_i represent probability of the observed length given that the word is correct/incorrect.
2. Pre and Post-Pausal durations. Word duration distributions are affected by the presence of a pause at the beginning or ending of a word. To model this, we created separate duration features for use in this condition.
3. Word Span Confusions. Sometimes long words in the lattice such as *Attendees* are mistaken for multiple short words, e.g. *A ten D*. When a word spans or is spanned by another, separate word duration features are used.

Duration scores were computed for the 100 most frequent words, which account for over 48% of all errors. Table 4 summarizes the results of these duration models. The results are in the context of the MSR word detectors, resulting in a relatively good initial error rate, and we are able to reduce the error rate by 0.3% absolute.

10. COMBINED BROADCAST NEWS RESULTS

The results are summarized for both development and test sets in Table 5. We see a consistent gain from both implementing SCARF training on top of the baseline, and adding the MSR word detectors. We hypothesize that the 0.3% improvement observed from retraining with the baseline feature is because a) the dynamic range of the baseline score is more limited than the original acoustic score and b) the LM weight is discriminatively tuned. Note, however, that this tuning is on the training data - not dev or test data.

Adding the MSR word detectors provides another large improvement. Adding the various information sources either individually or together produces about 0.3% improvement. Altogether we observe a relative reduction in WER of 8% for dev04f and over 9% for RT04.

Setup	dev04f	RT04
Baseline (Attila)	16.3%	15.7
SCARF1	16.0	15.4
+MSR Word detectors	15.3	14.5
+Duration,PPM,Phoneme Detectors	**15.0**	**14.2**
Lattice Oracle (lower bound)	**11.8**	**10.2**

Table 5. Combined Broadcast News Results.

11. CONCLUSION

This work has demonstrated the ability of Segmental Conditional Random Fields to integrate new sources of information on top of state-of-the-art baselines in two different ASR tasks. By using features based on template matching, duration models, phoneme detections, and Poisson process models, we observed 8-9% relative improvement in Broadcast News and Wall Street Journal recognition. The flexibility of the approach opens the possibility of building new systems based on information integration across numerous sources.

Acknowledgements

We thank Brian Kingsbury for his invaluable help with Attila.

12. REFERENCES

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

[2] G. Zweig and Nguyen, "SCARF: A segmental conditional random field toolkit for speech recognition," in *Proc. Interspeech*, 2010.

[3] M. De Wachter, M. Matton, K. Demuynck, P. Wambacq, R. Cools, and D. Van Compernolle, "Template based continuous speech recognition," *IEEE Trans. on Audio Speech and Language Processing*, vol. 15, pp. 1377–1390, May 2007.

[4] S. Thomas, S. Ganapathy, and H. Hermansky, "Phoneme recognition using spectral envelope and modulation frequency features," in *ICASSP*, 2009.

[5] G.S.V.S. Sivaram, S. Ganapathy, and H. Hermansky, "Sparse auto-associative neural networks: Theory and application to speech recognition," in *Proc. Interspeech*, 2010.

[6] A. Mohamed, G. Dahl, and G.E. Hinton, "Deep belief networks for phone recognition," in *NIPS Workshop on Deep Learning for Speech Recognition and Related Applications*, 2009.

[7] A. Jansen and P. Niyogi, "Point process models for spotting keywords in continuous speech," *IEEE Transactions on Audio Speech and Language Processing*, 2009.

[8] P. Clark and L. Atlas, "Time-frequency coherent modulation filtering of non-stationary signals," *IEEE Transactions on Signal Processing*, vol. 57, no. 11, 2009.

[9] G. Sell and M. Slaney, "Solving demodulation as an optimization problem," *IEEE Transactions on Signal Processing*, 2010.

[10] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *Proc. ICML*, 2001.

[11] S. Sarawagi and W. Cohen, "Semi-Markov Conditional Random Fields for Information Extraction," in *Proc. NIPS*, 2005.

[12] M. I. Layton and M. J. F. Gales, "Augmented statistical models for speech recognition," in *Proc. ICASSP*, 2006.

[13] K. Demuynck, J. Roelens, D. Van Compernolle, and P. Wambacq, "SPRAAK: an open source SPeech Recognition and Automatic Annotation Kit," in *Interspeech*, 2008.

[14] S.F. Chen, B. Kingsbury, L. Mangu, D. Povey, H. Soltau, and G. Zweig, "Advances in speech transcription at IBM under the DARPA EARS program," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, no. 5, 2006.

[15] S. Demange and D. Van Compernolle, "HEAR: an hybrid episodic-abstract speech recognizer," in *Proc. Interspeech*, 2009.