SCARF: A Segmental Conditional Random Field Toolkit for Speech Recognition

Geoffrey Zweig and Patrick Nguyen

Microsoft Corporation
One Microsoft Way, Redmond, WA 98052
gzweig,panguyen@microsoft.com

Abstract

This paper describes a new toolkit - SCARF - for doing speech recognition with segmental conditional random fields. It is designed to allow for the integration of numerous, possibly redundant segment level acoustic features, along with a complete language model, in a coherent speech recognition framework. SCARF performs a segmental analysis, where each segment corresponds to a word, thus allowing for the incorporation of acoustic features defined at the phoneme, multi-phone, syllable and word level. SCARF is designed to make it especially convenient to use acoustic detection events as input, such as the detection of energy bursts, phonemes, or other events. Language modeling is done by associating each state in the SCRF with a state in an underlying n-gram language model, and SCARF supports the joint and discriminative training of language model and acoustic model parameters. SCARF is available for download from http://research.microsoft.com/en-us/projects/scarf/

Index Terms: Segmental Conditional Random Field, Speech Recognition Toolkit

1. Introduction

Current HMM based speech recognizers [1, 2, 3] have reached a state of refinement in which good performance on a variety of tasks is possible. Typically, the core HMM approach is enhanced with training and adaptation methods such as VTLN, HLDA, fMLLR, SAT, (f)MMI, (f)MPE, and MLLR. Each method improves some aspect of system performance, and taken together they make a formidable combination. Despite the success of this approach, there are nevertheless some drawbacks which make it interesting to explore alternative paradigms. These drawbacks include the use of the frame level Markov assumption, the necessity to decorrelate features before modeling them with diagonal covariance gaussians, ad-hoc weighting factors associated with mixing discrete and continuous features, an "add-on" approach to discriminative training, and a forced separation between acoustic and language model training.

The SCARF toolkit is designed to support research into an alternative approach to speech recognition, based from the ground up on the combination of multiple, redundant, heterogeneous knowledge sources [4] in a discriminative framework of manageable complexity. In particular, SCARF has been designed to integrate features

- which may be either discrete or continuous
- which may vary in scale, e.g. being defined at either the sub-phonemic, phonemic, syllabic or word levels
- which are derived from the occurrence of acoustic events such as a phoneme detection
- which may carry redundant information, for example features derived from both phoneme and syllable detections
- whose parameters can be learned from training data, and then used in the context of a new vocabulary with previously unseen words

As its mathematical basis, SCARF adopts segmental conditional random fields also known as semi-Markov CRFs [5]. This is a two layer model, in which the "top" state layer corresponds to words, and the "bottom" layer corresponds to observations. In this model, each word is associated with a definite span of audio, and corresponding segment-level features can be extracted. This approach maintains conceptual simplicity by using a two-layer structure, while at the same time allowing for a rich set of features through the use of a segmental analysis. When SCARF is given input consisting of the detection of acoustic events, a wide variety of features can be automatically constructed, where each feature measures some form of consistency between the acoustic events and a word hypothesis. These features vary in the degree to which ordering constraints are imposed, and in their ability to generalize to unseen words.

As we will see in Section 2, computation with SCRFs involves summing and/or maximizing over all possible segmentations of the observations into words. Therefore, it is convenient to divide the computation between a "fast-match" that quickly identifies possible words and their boundaries, and a "detailed-match" that applies the full model. [6].

In SCARF, the fast-match may be done externally with an HMM system, and provided in the form of a lattice. Alternatively, SCARF implements a TF-IDF based fast match that finds potential words based on the TF-IDF similarity between the observations in a detection stream and those expected on the basis of dictionary pronunciations. This is done by applying a segment-level decoding process where the TF-IDF score is used as the acoustic score, and is fully described in a companion paper [7].

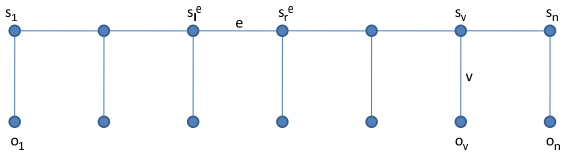

Figure 1: A CRF.

The remainder of this paper is organized as follows. In Section 2, we define the model. Section 3 defines the inputs to SCARF. Section 4 describes the features which SCARF uses. Finally, in Section 5 we describe the built-in fast match function, before concluding in Section 6.

2. Segmental CRF Model

We begin our discussion with the illustration of a classical CRF model [8] in Figure 1. Associated with each vertical edge v are one or more feature functions $f_k(s_v, o_v)$ relating the state variable to the associated observation. Associated with each horizontal edge e are one or more feature functions $g_d(s_l^e, s_r^e)$ defined on adjacent left and right states. (We use s_l^e and s_r^e to denote the left and right states associated with an edge e.) The set of functions (indexed by k and d) is fixed across segments. A set of trainable parameters λ_k and ρ_d is also present in the model. The conditional probability of a state sequence $\mathbf{s}$ given the observations $\mathbf{o}$ is given by

$$P(\mathbf{s}|\mathbf{o}) = \frac{\exp(\sum_{v,k} \lambda_k f_k(s_v, o_v) + \sum_{d,e} \rho_d g_d(s_l^e, s_r^e))}{\sum_{\mathbf{s}'} \exp(\sum_{v',k} \lambda_k f_k(s'_v, o_v) + \sum_{d,e'} \rho_d g_d(s_l'^e, s_r'^e))}$$

Segmental CRFs (SCRFs) extend CRFs by defining the feature functions at the *segment* rather than frame or single observation level. With segmental CRFs, it is necessary to sum over all possible segmentations consistent with the known word sequence during training, and to maximize over all segmentations consistent with a hypothesized word sequence at decoding time. The notion of different segmentations is illustrated in Figure 2, where two different segmentations of an observation stream are shown. SCRFs are related to the Hidden CRFs of [9] in that there is an unknown segmentation; however [9] applies the frame level Markov assumption to do phoneme recognition, and we are interested in a fully segmental model without the frame level Markov assumption, and one in which the states represent words. The c-Aug model of [10] defines a related segmental model using a fixed set of features.

We now define the SCARF model. Denote by $\mathbf{q}$ a segmentation of the observation sequences, for example that indicated by the boxes at the top of Fig. 2 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 at the top of Fig. 2 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. The segment $o(e)$ will span a block of observations from some start time to some endtime, o_{st}^{et}; at the top of Fig, 2, $o(e)$ is the block o_3^4. With this notation, we represent all functions as $f_k(s_l^e, s_r^e, o(e))$ where

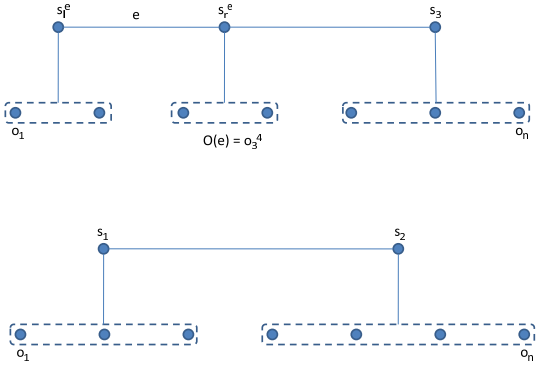

Figure 2: A Segmental CRF and two segmentations.

$o(e)$ are the observations associated with the segment of the right-hand state of the edge. (The first block of observations is treated with an extra notional edge leading into the leftmost state.) The conditional probability of a state sequence $\mathbf{s}$ given an observation sequence $\mathbf{o}$ for a SCRF is given by

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

Training is done by gradient descent using Rprop [11] and SCARF further applies $L1$ and $L2$ norm regularization. Taking the derivative of the log likelihood $\mathcal{L} = \log P(\mathbf{s}|\mathbf{o})$ with respect to λ_k we obtain the necessary gradient:

$$\frac{\partial \mathcal{L}}{\partial \lambda_k} = \frac{\sum_{\mathbf{q} \text{ s.t. } |\mathbf{q}|=|\mathbf{s}|} T_k(\mathbf{q}) \exp(\sum_{e\in\mathbf{q},k} \lambda_k f_k(s_l^e, s_r^e, o(e)))}{\sum_{\mathbf{q} \text{ s.t. } |\mathbf{q}|=|\mathbf{s}|} \exp(\sum_{e\in\mathbf{q},k} \lambda_k f_k(s_l^e, s_r^e, o(e)))} - \frac{\sum_{\mathbf{s}'} \sum_{\mathbf{q} \text{ s.t. } |\mathbf{q}|=|\mathbf{s}'|} T'_k(\mathbf{q}) \exp(\sum_{e\in\mathbf{q},k} \lambda_k f_k(s_l'^e, s_r'^e, o(e)))}{\sum_{\mathbf{s}'} \sum_{\mathbf{q} \text{ s.t. } |\mathbf{q}|=|\mathbf{s}'|} \exp(\sum_{e\in\mathbf{q},k} \lambda_k f_k(s_l'^e, s_r'^e, o(e)))},$$

with

$$T_k(\mathbf{q}) = \sum_{e\in\mathbf{q}} f_k(s_l^e, s_r^e, o(e))$$

$$T'_k(\mathbf{q}) = \sum_{e\in\mathbf{q}} f_k(s_l'^e, s_r'^e, o(e)).$$

This derivative can be computed efficiently with dynamic programming, using the recursions described in [12].

2.1. Continuous Speech Recognition

In order to model continuous speech, the values of the state variable in the SCRF model are made to correspond to states in a finite state representation of an n-gram language model. This is illustrated in Figure 3. In this figure, a fragment of a finite state language model is shown on the left. The states are numbered, and the words next to the states specify the linguistic state. At the right of this figure is a fragment of a CRF illustrating the word sequence "the dog nipped." The states are labeled with the index of the underlying language model state. This enables the transition features to fully encode an n-gram language model. Features derived from the language model are described in Section 4.4.

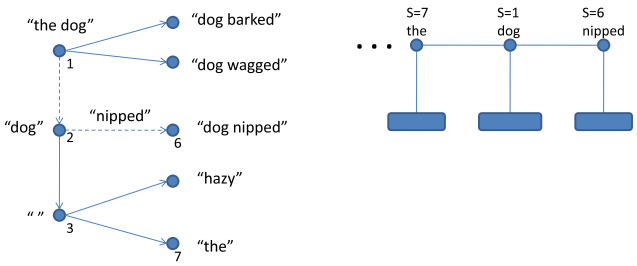

Figure 3: Correspondence between language model state and SCRF state. The dotted lines indicate the path taken in hypothesizing "nipped" after "the dog." A line from state 7 to state 1 has been omitted for clarity.

3. Inputs

The inputs to SCARF consist of the following elements:

- Acoustic detector streams, which indicate acoustic events that have been detected.
- Dictionaries that provide a "pronunciation" of each word in terms of the units in a detector stream.
- Transcriptions, used in training to define word/unit co-occurrence features.
- Numerator and denominator constraint files or lattices. These represent the segmentations into word sequences that have appreciable probability. The numerator lattices are restricted to paths consistent with the transcriptions.
- An ARPA-format language model.
- User defined features, expressed as lattice annotations.
- A "baseline" detector stream, consisting of the 1-best word output of any baseline system. This allows SCARF to build on existing "best-systems."

The formats of SCARF detector and lattice files are illustrated in Figure 4. A detection file simply shows which units are detected, and associates a single time with each unit, e.g. its midpoint. Multiple detector streams, in separate files, may be specified. In a lattice file, the first two columns are the start and end times of a word; the third column is the word value; the fourth and fifth columns are optional and are the indices of the start and end states (not times) of the word in a lattice graph. In the absence of these constraints, any word ending at time t can be followed by any other beginning at $t + 1$. The last column, also optional, provides the values of user-defined features. An arbitrary number of user features can be added using comma separation. A full specification of all file formats can be found in the *SCARF Manual* provided with the distribution.

4. Features

From each detection stream, several features may be extracted. Each is defined with respect to a temporal *span* of detection events and a specific word hypothesis for that

```
fileid
# phoneme stream
b 790
er 880
g 1045
ax 1125
r 1210
z 1265
.
```

```
fileid
1 140 <s>  0 1 feat1=0.12,feat2=0.23
1 160 <s>  0 2 feat1=0.0,feat2=0.1
141 160 uh 1 2  feat1=0.9,feat2=-1.0
161 220 hello 2 3 feat1=-0.1,feat2=-0.3
221 260 </s>  3 4 feat4=0.6,feat2=0.6
.
```

Figure 4: A SCARF detector stream (left) and lattice file (right). They are shown together to save space.

span. The Expectation and Levenshtein features below are notable in that their weights can be trained with one set of data, and then used in cases where other, previously unseen, words may be present.

4.1. 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

4.2. 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: All edits are considered to operate on the dictionary pronunciation.

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

Compared to Expectation features, Levenshtein features are more sensitive to the exact ordering of the units within a segment.

4.3. Existence Features

Whereas Expectation and Levenshtein features require 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.

4.4. Language Model Features

SCARF uses the language model in two ways. First, conventional smoothed n-gram probabilities can be returned as transition features. A single λ is trained to weight these

features, resulting in a single discriminatively trained language model weight. Secondly, 0/1 indicator features can be introduced, one for each arc in the finite state language model, which indicate when an arc is traversed in the transition from one state to another. For example, in Figure 3, the arcs $(1, 2)$ and $(2, 6)$ are traversed in moving from state 1 to state 6. Learning the weights on the indicator features results in a discriminatively trained language model, trained jointly with the acoustic model.

4.5. The Baseline Feature

The baseline feature is designed to be used in association with an existing HMM system, to provide a baseline level of performance on which to build. It requires only the baseline one-best sequence, which is treated as a detector sequence. The baseline feature for a segment is always either $+1$ or -1. It is $+1$ when the 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 has the same number of words as the baseline decoding, and the identities of the words match. Thus, by assigning a high enough weight to the baseline feature, the best scoring hypothesis can be guaranteed to be the baseline and thus match its performance. In practice, the baseline weighting is learned and its value will depend on the relative power of the additional features.

4.6. Summary of Features

Table 1 summarizes the automatically defined features, and whether, after training, the associated weights still can be used with new vocabularies and language models. U is the number of distinct units in a detector stream. Though not previously mentioned, Existence and Expectation features can be automatically created for n-grams of atomic units, and this is the meaning of n in those columns. V is the size of the word vocabulary, and A the number of arcs in the finite state language model representation. Note that multiple detector streams may be used, in which case the number of features is linearly increased.

5. TF-IDF Fast Match

The use of lattices to constrain the set of possible segmentations is important to keep the SCARF runtime reasonable (typically below real time). However, it creates an external dependency on a separate system to provide the lattices. As another option, SCARF provides a built in "fast-match" which operates directly on a detector stream to produce lattices. Algorithmically, the process is identical to the SCARF search process itself, except that no a-priori constraints on the search space are required, and there is a single acoustic feature - the TF-IDF similarity between the detected units in a segment, and the expected pronunciation of the hypothesized word. We refer to the reader to a companion paper [7] for a full description.

Name	**Number**	**Vocab & LM Independent**
Expectation	$3U^n$	yes
Levenshtein	$4U$	yes
Existence	VU^n	no
Single LM weight	1	yes
LM arc features	A	no
Baseline	1	yes

Table 1: Summary of automatically constructed features. The weights of vocabulary and LM independent features can be learned with once and then applied in the context of a new vocabulary or LM.

6. Conclusion

This paper has described the SCARF toolkit for speech recognition with segmental conditional random fields. SCARF is designed to further research in speech recognition with multiple partially redundant detection events, and is available at http://research.microsoft.com/en-us/projects/scarf/.

7. References

[1] S.F. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau, and G. Zweig, "Advances in Speech Transcription at IBM Under the DARPA EARS Program," *IEEE Transactions on Audio, Speech amd Language Processing*, vol. 14, no. 5, 2006.

[2] S. Matsoukas, J.L. Gauvain, G. Adda, T. Colhurst, C.L. Kao, O. Kimball, L. Lamel, F. Lefevre, J.Z. Ma, J. Makhoul, L. Nguyen, R. Prasad, R. Schwatrz, H. Schwenk, and B. Xiang, "Advances in Transcription of Broadcast News and Conversational Telephone Speech Within the Combined EARS BBN/LIMSI System," *IEEE Transactions on Audio, Speech amd Language Processing*, vol. 14, no. 5, 2006.

[3] Gales M.J.F., D.Y. Kim, Woodland P.C., Chan H.Y., R. Mrva D. amd Sinha, and S.E. Tranter, "Progress in the CU-HTK Broadcast News Transcription System," *IEEE Transactions on Audio, Speech amd Language Processing*, vol. 14, no. 5, 2006.

[4] C-H. Lee, "From Knowledge-Ignorant to Knowledge-Rich Modeling: A New Speech Research Paradigm for Next Generation Automatic Speech Recognition," in *ICSLP*, 2004.

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

[6] L.R. Bahl, P.V. de Souza, P.S. Gopalakrishnan, D. Nahamoo, and M. Picheny, "A Fast Match for Continuous Speech Recognition using Allophonic Models," in *ICASSP*, 1992.

[7] G. Zweig, P. Nguyen, J. Droppo, and A. Acero, "Continuous Speech Recognition with a TF-IDF Acoustic Model," in *Proc. Interspeech*, Submitted.

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

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

[10] M. I. Layton and M. J. F. Gales, "Augmented Statistical Models for Speech Recognition," in *in Proc. ICASSP*, 2006.

[11] M. Reidmiller, "Rprop - Description and Implementation Details," Tech. Rep., University of Karlsruhe, January 1994.

[12] G. Zweig and P. Nguyen, "A Segmental CRF Approach to Large Vocabulary Continuous Speech Recognition," in *Proc. ASRU*, 2009.