A Hybrid Markov/Semi-Markov Conditional Random Field for Sequence Segmentation

Galen Andrew
Microsoft Research
One Microsoft Way
Redmond, WA 98052
galena@microsoft.com

Abstract

Markov order-1 conditional random fields (CRFs) and semi-Markov CRFs are two popular models for sequence segmentation and labeling. Both models have advantages in terms of the type of features they most naturally represent. We propose a hybrid model that is capable of representing both types of features, and describe efficient algorithms for its training and inference. We demonstrate that our hybrid model achieves error reductions of 18% and 25% over a standard order-1 CRF and a semi-Markov CRF (resp.) on the task of Chinese word segmentation. We also propose the use of a powerful feature for the semi-Markov CRF: the log conditional odds that a given token sequence constitutes a chunk according to a generative model, which reduces error by an additional 13%. Our best system achieves 96.8% F-measure, the highest reported score on this test set.

1 Introduction

The problem of segmenting sequence data into chunks arises in many natural language applications, such as named-entity recognition, shallow parsing, and word segmentation in East Asian languages. Two popular discriminative models that have been proposed for these tasks are the conditional random field (CRFs) (Lafferty et al., 2001) and the semi-Markov conditional random field (semi-CRF) (Sarawagi and Cohen, 2004).

A CRF in its basic form is a model for labeling tokens in a sequence; however it can easily be adapted to perform segmentation via labeling each token as BEGIN or CONTINUATION, or according to some similar scheme. CRFs using this technique have been shown to be very successful at the task of Chinese word segmentation (CWS), starting with the model of Peng et al. (2004). In the Second International Chinese Word Segmentation Bakeoff (Emerson, 2005), two of the highest scoring systems in the closed track competition were based on a CRF model. (Tseng et al., 2005; Asahara et al., 2005)

While the CRF is quite effective compared with other models designed for CWS, one wonders whether it may be limited by its restrictive independence assumptions on non-adjacent labels: an order-M CRF satisfies the order-M Markov assumption that, globally conditioned on the input sequence, each label is independent of all other labels given the M labels to its left and right. Consequently, the model only "sees" word boundaries within a moving window of $M + 1$ characters, which prohibits it from explicitly modeling the tendency of strings longer than that window to form words, or from modeling the lengths of the words. Although the window can in principle be widened by increasing M, this is not a practical solution as the complexity of training and decoding a linear sequence CRF grows exponentially with the Markov order.

The semi-CRF is a sequence model that is designed to address this difficulty via careful relaxation of the Markov assumption. Rather than recasting the segmentation problem as a labeling problem, the semi-CRF directly models the distribution of chunk boundaries.[1] In terms of inde-

[1] As it was originally described, the semi-CRF also assigns labels to each chunk, effectively performing joint segmentation and labeling, but in a pure segmentation problem such as CWS, the use of labels is unnecessary.

Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006), pages 465–472,
Sydney, July 2006.

pendence, using an order-M semi-CRF entails the assumption that, globally conditioned on the input sequence, the position of each chunk boundary is independent of all other boundaries given the positions of the M boundaries to its left and right *regardless of how far away they are*. Even with an order-1 model, this enables several classes of features that one would expect to be of great utility to the word segmentation task, in particular *word length* and *word identity*.

Despite this, the only work of which we are aware exploring the use of a semi-Markov CRF for Chinese word segmentation did not find significant gains over the standard CRF (Liang, 2005). This is surprising, not only because the additional features a semi-CRF enables are intuitively very useful, but because as we will show, an order-M semi-CRF is strictly more powerful than an order-M CRF, in the sense that any feature that can be used in the latter can also be used in the former, or equivalently, the semi-CRF makes strictly weaker independence assumptions. Given a judicious choice of features (or simply enough training data) the semi-CRF should be superior.

We propose that the reason for this discrepancy may be that despite the greater representational power of the semi-CRF, there are some valuable features that are more naturally expressed in a CRF segmentation model, and so they are not typically included in semi-CRFs (indeed, they have not to date been used in any semi-CRF model for any task, to our knowledge). In this paper, we show that semi-CRFs are strictly more expressive, and also demonstrate how CRF-type features can be used in a semi-CRF model for Chinese word segmentation. Our experiments show that a model incorporating both types of features can outperform models using only one or the other type.

Orthogonally, we explore in this paper the use of a very powerful feature for the semi-CRF derived from a generative model.

It is common in statistical NLP to use as features in a discriminative model the (logarithm of the) estimated probability of some event according to a generative model. For example, Collins (2000) uses a discriminative classifier for choosing among the top N parse trees output by a generative baseline model, and uses the log-probability of a parse according to the baseline model as a feature in the reranker. Similarly, the machine translation system of Och and Ney uses log-probabilities of phrasal translations and other events as features in a log-linear model (Och and Ney, 2002; Och and Ney, 2004). There are many reasons for incorporating these types of features, including the desire to combine the higher accuracy of a discriminative model with the simple parameter estimation and inference of a generative one, and also the fact that generative models are more robust in data sparse scenarios (Ng and Jordan, 2001).

For word segmentation, one might want to use as a local feature the log-probability that a segment is a word, given the character sequence it spans. A curious property of this feature is that it induces a counterintuitive asymmetry between the *is-word* and *is-not-word* cases: the component generative model can effectively dictate that a certain chunk is *not* a word, by assigning it a very low probability (driving the feature value to negative infinity), but it cannot dictate that a chunk *is* a word, because the log-probability is bounded above.[2] If instead the *log conditional odds* $\log \frac{P_i(\mathbf{y}|\mathbf{x})}{P_i(\neg\mathbf{y}|\mathbf{x})}$ is used, the asymmetry disappears. We show that such a log-odds feature provides much greater benefit than the log-probability, and that it is useful to include such a feature even when the model also includes indicator function features for every word in the training corpus.

2 Hybrid Markov/Semi-Markov CRF

The model we describe is formally a type of semi-Markov CRF, distinguished only in that it also involves CRF-style features. So we first describe the semi-Markov model in its general form.

2.1 Semi-Markov CRF

An (unlabeled) semi-Markov conditional random field is a log-linear model defining the conditional probability of a segmentation given an observation sequence. The general form of a log-linear model is as follows: given an input $\mathbf{x} \in X$, an output $\mathbf{y} \in Y$, a feature mapping $\Phi : X \times Y \mapsto \mathbb{R}^n$, and a weight vector $\mathbf{w}$, the conditional probability of $\mathbf{y}$ given $\mathbf{x}$ is estimated as:

$$P(\mathbf{y} \mid \mathbf{x}) = \frac{\exp\left(\mathbf{w} \cdot \Phi(\mathbf{x}, \mathbf{y})\right)}{Z(\mathbf{x})}$$

where $Z : \mathbf{x} \mapsto R$ is a normalizing factor. $\mathbf{w}$ is typically chosen to maximize the conditional likelihood of a labeled training set. In the word

[2] We assume the weight assigned to the log-probability feature is positive.

segmentation task, $\mathbf{x}$ is an ordered sequence of characters $(x_1, x_2, \ldots, x_n)$, and $\mathbf{y}$ is a set of indices corresponding to the start of each word: $\{y_1, y_2, \ldots, y_m\}$ such that $y_1 = 1$, $y_m \leq n$, and for all j, $y_j < y_{j+1}$. A log-linear model in this space is an order-1 semi-CRF if its feature map Φ decomposes according to

$$\Phi(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{m} \phi^S(y_j, y_{j+1}, \mathbf{x}) \qquad (1)$$

where ϕ^S is a local feature map that only considers one chunk at a time (defining $y_{m+1} = n+1$). This decomposition is responsible for the characteristic independence assumptions of the semi-CRF.

Hand-in-hand with the feature decomposition and independence assumptions comes the capacity for exact decoding using the Viterbi algorithm, and exact computation of the objective gradient using the forward-backward algorithm, both in time quadratic in the lengths of the sentences. Furthermore, if the model is constrained to propose only chunkings with maximum word length k, then the time for inference and training becomes linear in the sentence length (and in k). For Chinese word segmentation, choosing a moderate value of k does not pose any significant risk, since the vast majority of Chinese words are only a few characters long: in our training set, 91% of word tokens were one or two characters, and 99% were five characters or less.

Using a semi-CRF as opposed to a traditional Markov CRF allows us to model some aspects of word segmentation that one would expect to be very informative. In particular, it makes possible the use of local indicator function features of the type "the chunk consists of character sequence $\chi_1, \ldots, \chi_\ell$," or "the chunk is of length ℓ." It also enables "pseudo-bigram language model" features, firing when a given word occurs in the context of a given character unigram or bigram.[3] And crucially, although it is slightly less natural to do so, any feature used in an order-1 Markov CRF can also be represented in a semi-CRF. As Markov CRFs are used in the most competitive Chinese word segmentation models to date, one might expect that incorporating both types of features could yield a superior model.

[3]We did not experiment with this type of feature.

2.2 CRF vs. Semi-CRF

In order to compare the two types of linear CRFs, it is convenient to define a representation of the segmentation problem in terms of character labels as opposed to sets of whole words. Denote by $L(\mathbf{y}) \in \{B, C\}^n$ (for BEGIN vs. CONTINUATION) the sequence $\{L_1, L_2, \ldots L_n\}$ of labels such that $L_i = B$ if and only if $y_i \in \mathbf{y}$. It is clear that if we constrain $L_1 = B$, the two representations $\mathbf{y}$ and $L(\mathbf{y})$ are equivalent. An order-1 Markov CRF is a log-linear model in which the global feature vector Φ decomposes into a sum over local feature vectors that consider bigrams of the label sequence:

$$\Phi(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \phi^M(L_i, L_{i+1}, i, \mathbf{x}) \qquad (2)$$

(where L_{n+1} is defined as B). The local features that are most naturally expressed in this context are indicators of some joint event of the label bigram (L_i, L_{i+1}) and nearby characters in $\mathbf{x}$. For example, one might use the feature "the current character x_i is χ and $L_i = C$", or "the current and next characters are identical and $L_i = L_{i+1} = B$."

Although we have heretofore disparaged the CRF as being incapable of representing such powerful features as word identity, the type of features that it most naturally represents should be helpful in CWS for generalizing to unseen words. For example, the first feature mentioned above could be valuable to rule out certain word boundaries if χ were a character that typically occurs only as a suffix but that combines freely with a variety of root forms to create new words. This type of feature (specifically, a feature indicating the *absence* as opposed to the *presence* of a chunk boundary) is a bit less natural in a semi-CRF, since in that case local features $\phi^S(y_j, y_{j+1}, \mathbf{x})$ are defined on pairs of adjacent boundaries. Information about which tokens are *not* on boundaries is only implicit, making it a bit more difficult to incorporate that information into the features. Indeed, neither Liang (2005) nor Sarawagi and Cohen (2004) nor any other system using a semi-Markov CRF on any task has included this type of feature to our knowledge. We hypothesize (and our experiments confirm) that the lack of this feature explains the failure of the semi-CRF to outperform the CRF for word segmentation in the past.

Before showing how CRF-type features can be used in a semi-CRF, we first demonstrate that the semi-CRF is indeed strictly more expressive than

the CRF, meaning that any global feature map Φ that decomposes according to (2) also decomposes according to (1). It is sufficient to show that for any feature map Φ^M of a Markov CRF, there exists a semi-Markov-type feature map Φ^S such that for any $\mathbf{x}, \mathbf{y}$,

$$\Phi^M(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \phi^M(L_i, L_{i+1}, i, \mathbf{x}) \quad (3)$$
$$= \sum_{j=1}^{m} \phi^S(y_j, y_{j+1}, \mathbf{x}) = \Phi^S(\mathbf{x}, \mathbf{y})$$

To this end, note that there are only four possible label bigrams: BB, BC, CB, and CC. As a direct result of the definition of $L(\mathbf{y})$, we have that $(L_i, L_{i+1}) = (B, B)$ if and only if some word of length one begins at i, or equivalently, there exists a word j such that $y_j = i$ and $y_{j+1} - y_j = 1$. Similarly, $(L_i, L_{i+1}) = (B, C)$ if and only if some word of length > 1 begins at i, etc. Using these conditions, we can define ϕ^S to satisfy equation 3 as follows:

$$\phi^S(y_j, y_{j+1}, \mathbf{x}) = \phi^M(B, B, y_j, \mathbf{x})$$

if $y_{j+1} - y_j = 1$, and

$$\phi^S(y_j, y_{j+1}, \mathbf{x}) = \phi^M(B, C, y_j, \mathbf{x}) + \sum_{k=y_j+1}^{y_{j+1}-2} \phi^M(C, C, k, \mathbf{x}) + \phi^M(C, B, y_{j+1} - 1, \mathbf{x}) \quad (4)$$

otherwise. Defined thus, $\sum_{j=1}^{m} \phi^S$ will contain exactly n ϕ^M terms, corresponding to the n label bigrams.[4]

2.3 Order-1 Markov Features in a Semi-CRF

While it is fairly intuitive that any feature used in a 1-CRF can also be used in a semi-CRF, the above argument reveals an algorithmic difficulty that is likely another reason that such features are not typically used. The problem is essentially an effect of the sum for CC label bigrams in (4): quadratic time training and decoding assumes that the features of each chunk $\phi^S(y_j, y_{j+1}, \mathbf{x})$ can be multiplied with the weight vector $\mathbf{w}$ in a number of operations that is roughly constant over all chunks,

procedure $ComputeScores(\mathbf{x}, \mathbf{w})$
for $i = 2 \ldots (n-1)$ **do**
  $\sigma_i^{CC} \leftarrow \phi^M(C, C, i, \mathbf{x}) \cdot \mathbf{w}$
end for
for $a = 1 \ldots n$ **do**
  $CCsum \leftarrow 0$
  for $b = (a+1) \ldots (n+1)$ **do**
    if $b - a = 1$ **then**
      $\sigma_{ab} \leftarrow \phi^M(B, B, a, \mathbf{x}) \cdot \mathbf{w}$
    else
      $\sigma_{ab} \leftarrow \phi^M(B, C, a, \mathbf{x}) \cdot \mathbf{w} + CCsum + \phi^M(C, B, b-1, \mathbf{x}) \cdot \mathbf{w}$
      $CCsum \leftarrow CCsum + \sigma_{b-1}^{CC}$
    end if
  end for
end for

Figure 1: Dynamic program for computing chunk scores σ_{ab} with 1-CRF-type features.

but if one naïvely distributes the product over the sum, longer chunks will take proportionally longer to score, resulting in cubic time algorithms.[5]

In fact, it is possible to use these features without any asymptotic decrease in efficiency by means of a dynamic program. Both Viterbi and forward-backward involve the scores $\sigma_{ab} = \mathbf{w} \cdot \phi^S(a, b, \mathbf{x})$. Suppose that before starting those algorithms, we compute and cache the score σ_{ab} of each chunk, so that remainder the algorithm runs in quadratic time, as usual. This pre-computation can be done quickly if we first compute the values $\sigma_i^{CC} = \mathbf{w} \cdot \phi^M(C, C, i, \mathbf{x})$, and use them to fill in the values of σ_{ab} as shown in Figure 1.

In addition, computing the gradient of the semi-CRF objective requires that we compute the expected value of each feature. For CRF-type features, this is tantamount to being able to compute the probability that each label bigram (L_i, L_{i+1}) takes any value. Assume that we have already run standard forward-backward inference so that we have for any (a, b) the probability that the subsequence $(\mathbf{x}_a, \mathbf{x}_{a+1}, \ldots, \mathbf{x}_{b-1})$ segments as a chunk, $P(chunk(a, b))$. Computing the probability that (L_i, L_{i+1}) takes the values BB, BC or CB is simple to compute:

$$P(L_i, L_{i+1} = BB) = P(chunk(i, i+1))$$

[4] We have discussed the case of Markov order-1, but the argument can be generalized to show that an order-M CRF has an equivalent representation as an order-M semi-CRF, for any M.

[5] Note that the problem would arise even if only zero-order Markov (label unigram) features were used, only in that case the troublesome features would be those that involved the label unigram C.

and, e.g.,

$$P(L_i, L_{i+1} = BC) = \sum_{j>i+1} P(chunk(i,j)),$$

but the same method of summing over chunks cannot be used for the value CC since for each label bigram there are quadratically many chunks corresponding to that value. In this case, the solution is deceptively simple: using the fact that for any given label bigram, the sum of the probabilities of the four labels must be one, we can deduce that

$$P(L_i, L_{i+1} = CC) = 1.0 - P(L_i, L_{i+1} = BB) - P(L_i, L_{i+1} = BC) - P(L_i, L_{i+1} = CB).$$

One might object that features of the C and CC labels (the ones presenting algorithmic difficulty) are unnecessary, since under certain conditions, their removal would not in fact change the expressivity of the model or the distribution that maximizes training likelihood. This will indeed be the case when the following conditions are fulfilled:

1. All label bigram features are of the form

   $$\phi^M(L_i, L_{i+1}, i, \mathbf{x}) = \mathbf{1}\{(L_i, L_{i+1}) = \alpha \;\&\; pred(i, \mathbf{x})\}$$

   for some label bigram α and predicate $pred$, and any such feature with a given predicate has variants for all four label bigrams α.

2. No regularization is used during training.

A proof of this claim would require too much space for this paper, but the key is that, given a model satisfying the above conditions, one can obtain an equivalent model via adding, for each feature type over $pred$, some constant to the four weights corresponding to the four label bigrams, such that the CC bigram features all have weight zero.

In practice, however, one or both of these conditions is always broken. It is common knowledge that regularization of log-linear models with a large number of features is necessary to achieve high performance, and typically in NLP one defines feature templates and chooses only those features that occur in some positive example in the training set. In fact, if both of these conditions are fulfilled, it is very likely that the optimal model will have some weights with infinite values. We conclude that it is not a practical alternative to omit the C and CC label features.

2.4 Generative Features in a Discriminative Model

When using the output of a generative model as a feature in a discriminative model, Raina et al. (2004) provide a justification for the use of log conditional odds as opposed to log-probability: they show that using log conditional odds as features in a logistic regression model is equivalent to discriminatively training weights for the features of a Naïve Bayes classifier to maximize conditional likelihood.[6] They demonstrate that the resulting classifier, termed a "hybrid generative/discriminative classifier", achieves lower test error than either pure Naïve Bayes or pure logistic regression on a text classification task, regardless of training set size.

The hybrid generative/discriminative classifier also uses a unique method for using the same data used to estimate the parameters of the component generative models for training the discriminative model parameters $\mathbf{w}$ without introducing bias. A "leave-one-out" strategy is used to choose $\mathbf{w}$, whereby the feature values of the i-th training example are computed using probabilities estimated with the i-th example held out. The beauty of this approach is that since the probabilities are estimated according to (smoothed) relative frequency, it is only necessary during feature computation to maintain sufficient statistics and adjust them as necessary for each example.

In this paper, we experiment with the use of a single "hybrid" local semi-CRF feature, the smoothed log conditional odds that a given subsequence $\mathbf{x}_{ab} = (\mathbf{x}_a, \ldots, \mathbf{x}_{b-1})$ forms a word:

$$\log \frac{wordcount(\mathbf{x}_{ab}) + 1}{nonwordcount(\mathbf{x}_{ab}) + 1},$$

where $wordcount(\mathbf{x}_{ab})$ is the number of times $\mathbf{x}_{ab}$ forms a word in the training set, and $nonwordcount(\mathbf{x}_{ab})$ is the number of times $\mathbf{x}_{ab}$ occurs, not segmented into a single word. The models we test are not strictly speaking hybrid generative/discriminative models, since we also use indicator features not derived from a generative model. We did however use the leave-one-out approach for computing the log conditional odds feature during training.

[6] In fact, one more step beyond what is shown in that paper is required to reach the stated conclusion, since their features are not actually log conditional odds, but $\log \frac{P(\mathbf{x}|\mathbf{y})}{P(\mathbf{x}|\neg\mathbf{y})}$. It is simple to show that in the given context this feature is equivalent to log conditional odds.

3 Experiments

To test the ideas discussed in this paper, we compared the performance of semi-CRFs using various feature sets on a Chinese word segmentation task. The data used was the Microsoft Research Beijing corpus from the Second International Chinese Word Segmentation Bakeoff (Emerson, 2005), and we used the same train/test split used in the competition. The training set consists of 87K sentences of Beijing dialect Chinese, hand segmented into 2.37M words. The test set contains 107K words comprising roughly 4K sentences. We used a maximum word length k of 15 in our experiments, which accounted for 99.99% of the word tokens in our training set. The 249 training sentences that contained words longer than 15 characters were discarded. We did not discard any test sentences.

In order to be directly comparable to the Bakeoff results, we also worked under the very strict "closed test" conditions of the Bakeoff, which require that no information or data outside of the training set be used, not even prior knowledge of which characters represent Arabic numerals, Latin characters or punctuation marks.

3.1 Features Used

We divide our main features into two types according to whether they are most naturally used in a CRF or a semi-CRF.

The CRF-type features are indicator functions that fire when the character label (or label bigram) takes some value and some predicate of the input at a certain position relative to the label is satisfied. For each character label unigram L at position i, we use the same set of predicate templates checking:

- The identity of $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$
- The identity of the character bigram starting at positions $i-2$, $i-1$ and i
- Whether $\mathbf{x}_j$ and $\mathbf{x}_{j+1}$ are identical, for $j = (i-2)\ldots i$
- Whether $\mathbf{x}_j$ and $\mathbf{x}_{j+2}$ are identical, for $j = (i-3)\ldots i$
- Whether the sequence $\mathbf{x}_j \ldots \mathbf{x}_{j+3}$ forms an $AABB$ sequence for $j = (i-4)\ldots i$
- Whether the sequence $\mathbf{x}_j \ldots \mathbf{x}_{j+3}$ forms an $ABAB$ sequence for $j = (i-4)\ldots i$

The latter four feature templates are designed to detect character or word reduplication, a morphological phenomenon that can influence word segmentation in Chinese. The first two of these were also used by Tseng et al. (2005).

For label bigrams (L_i, L_{i+1}), we use the same templates, but extending the range of positions by one to the right.[7] Each label uni- or bigram also has a "prior" feature that always fires for that label configuration. All configurations contain the above features for the label unigram B, since these are easily used in either a CRF or semi-CRF model. To determine the influence of CRF-type features on performance, we also test configurations in which both B and C label features are used, and configurations using all label uni- and bigrams.

In the semi-Markov conditions, we also use as feature templates indicators of the length of a word ℓ, for $\ell = 1\ldots k$, and indicators of the identity of the corresponding character sequence.

All feature templates were instantiated with values that occur in positive training examples. We found that excluding CRF-type features that occur only once in the training set consistently improved performance on the development set, so we use a count threshold of two for the experiments. We do not do any thresholding of the semi-CRF features, however.

Finally, we use the single generative feature, log conditional odds that the given string forms a word. We also present results using the more typical log conditional probability instead of the odds, for comparison. In fact, these are both semi-Markov-type features, but we single them out to determine what they contribute over and above the other semi-Markov features.

3.2 Results

The results of test set runs are summarized in table 3.2. The columns indicate which CRF-type features were used: features of only the label B, features of label unigrams B and C, or features of all label unigrams and bigrams. The rows indicate which semi-Markov-type features were used:

[7] For both label unigram and label bigram features, the indices are chosen so that the feature set exhibits no asymmetry with respect to direction: for each feature considering some boundary and some property of the character(s) at a given offset to the left, there is a corresponding feature considering that boundary and the same property of the character(s) at the same offset to the right, and vice-versa.

Features	B only	uni	uni+bi
none	92.33	94.71	95.69
semi	95.28	96.05	96.46
prob	93.86	95.40	96.04
semi+prob	95.51	96.24	96.55
odds	95.10	96.06	96.40
semi+odds	96.27	96.77	96.84

Table 1: Test F-measure for different model configurations.

"semi" means length and word identity features were used, "prob" means the log-probability feature was used, and "odds" means the log-odds feature was used.

To establish the impact of each type of feature (C label unigrams, label bigrams, semi-CRF-type features, and the log-odds feature), we look at the reduction in error brought about by adding each type of feature. First consider the effect of the CRF-type features. Adding the C label features reduces error by 31% if no semi-CRF features are used, by 16% when semi-CRF indicator features are turned on, and by 13% when all semi-CRF features (including log-odds) are used. Using all label bigrams reduces error by 44%, 25%, and 15% in these three conditions, respectively.

Contrary to previous conclusions, our results show a significant impact due to the use of semi-CRF-type features, when CRF-type features are held constant. Adding semi-CRF indicator features results in a 38% error reduction without CRF-type features, and 18% with them. Adding semi-CRF indicator features plus the log-odds feature gives 52% and 27% in these two conditions, respectively.

Finally, across configurations, the log conditional odds does much better than log conditional probability. When the log-odds feature is added to the complete CRF model (uni+bi) as the only semi-CRF-type feature, errors are reduced by 24%, compared to only 7.6% for the log-probability. Even when the other semi-CRF-type features are present as well, log-odds reduces error by 13% compared to 2.5% for log-probability.

Our best model, combining all features, resulted in an error reduction of 12% over the highest score on this dataset from the 2005 Sighan closed test competition (96.4%), achieved by the pure CRF system of Tseng et al. (2005).

3.3 Discussion

Our results indicate that both Markov-type and semi-Markov-type features are useful for generalization to unseen data. This may be because the two types of features are in a sense complementary: semi-Markov-type features such as word-identity are valuable for modeling the tendency of known strings to segment as words, while label based features are valuable for modeling properties of sub-lexical components such as affixes, helping to generalize to words that have not previously been encountered. We did not explicitly test the utility of CRF-type features for improving recall on out-of-vocabulary items, but we note that in the Bakeoff, the model of Tseng et al. (2005), which was very similar to our CRF-only system (only containing a few more feature templates), was consistently among the best performing systems in terms of test OOV recall (Emerson, 2005).

We also found that for this sequence segmentation task, the use of log conditional odds as a feature results in much better performance than the use of the more typical log conditional probability. It would be interesting to see the log-odds applied in more contexts where log-probabilities are typically used as features. We have presented the intuitive argument that the log-odds may be advantageous because it does not exhibit the 0-1 asymmetry of the log-probability, but it would be satisfying to justify the choice on more theoretical grounds.

4 Relation to Previous Work

There is a significant volume of work exploring the use of CRFs for a variety of chunking tasks, including named-entity recognition, gene prediction, shallow parsing and others (Finkel et al., 2005; Culotta et al., 2005; Sha and Pereira, 2003). The current work indicates that these systems might be improved by moving to a semi-CRF model.

There have not been a large number of studies using the semi-CRF, but the few that have been done found only marginal improvements over pure CRF systems (Sarawagi and Cohen, 2004; Liang, 2005; Daumé III and Marcu, 2005). Notably, none of those studies experimented with features of chunk *non*-boundaries, as is achieved by the use of CRF-type features involving the label C, and we take this to be the reason for their not obtaining higher results.

Although it has become fairly common in NLP to use the log conditional probabilities of events as features in a discriminative model, we are not aware of any work using the log conditional odds.

5 Conclusion

We have shown that order-1 semi-Markov conditional random fields are strictly more expressive than order-1 Markov CRFs, and that the added expressivity enables the use of features that lead to improvements on a segmentation task. On the other hand, Markov CRFs can more naturally incorporate certain features that may be useful for modeling sub-chunk phenomena and generalization to unseen chunks. To achieve the best performance for segmentation, we propose that both types of features be used, and we show how this can be done efficiently.

Additionally, we have shown that a log conditional odds feature estimated from a generative model can be superior to the more common log conditional probability.

6 Acknowledgements

Many thanks to Kristina Toutanova for her thoughtful discussion and feedback, and also to the anonymous reviewers for their suggestions.

References

Masayuki Asahara, Kenta Fukuoka, Ai Azuma, Chooi-Ling Goh, Yotaro Watanabe, Yuji Matsumoto, and Takahashi Tsuzuki. 2005. Combination of machine learning methods for optimum chinese word segmentation. In *Proc. Fourth SIGHAN Workshop on Chinese Language Processing*, pages 134–137.

Michael Collins. 2000. Discriminative reranking for natural language parsing. In *Proc. 14th International Conf. on Machine Learning*.

Aron Culotta, David Kulp, and Andrew McCallum. 2005. Gene prediction with conditional random fields. Technical report, University of Massachusetts Dept. of Computer Science, April.

Hal Daumé III and Daniel Marcu. 2005. Learning as search optimization: Approximate large margin methods for structured prediction. In *Proc. 19th International Conf. on Machine Learning*.

Thomas Emerson. 2005. The second international chinese word segmentation bakeoff. In *Proc. Fourth SIGHAN Workshop on Chinese Language Processing*, pages 123–133.

Jenny Finkel, Trond Grenager, and Christopher D. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. *Proc. 41th Annual Meeting of the Assiciation of Computation Linguistics*.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA.

Percy Liang. 2005. Semi-supervised learning for natural language. Master's thesis, Massachusetts Institute of Technology.

Andrew Y. Ng and Michael I. Jordan. 2001. On discriminative vs. generative classifiers: A comparison of logistic regression and Naïve Bayes. In *Proc. Advances in Neural Information Processing 14*.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. *Proc. 38th Annual Meeting of the Assiciation of Computation Linguistics*.

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449, December.

Fuchun Peng, Fangfang Feng, and Andrew McCallum. 2004. Chinese segmentation and new word detection using conditional random fields. In *Proc. 20th International Conf. on Computational Linguistics*.

Rajat Raina, Yirong Shen, Andrew Y. Ng, and Andrew McCallum. 2004. Classification with hybrid generative/discriminative models. In *Proc. Advances in Neural Information Processing 17*.

Brian Roark and Seeger Fisher. 2005. OGI/OHSU baseline multilingual multi-document sumarization system. In *Proc. Multilingual Summarization Evaluation in ACL Workshop: Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*.

Sunita Sarawagi and William Cohen. 2004. Semimarkov conditional random fields for information extraction. In *Proc. 18th International Conf. on Machine Learning*.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. *Proc. HLT-NAACL*.

Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher Manning. 2005. A conditional random field word segmenter for sighan bakeoff 2005. In *Proc. Fourth SIGHAN Workshop on Chinese Language Processing*, pages 168–171.