Extracting Opinion Expressions with semi-Markov Conditional Random Fields

Bishan Yang
Department of Computer Science
Cornell University
bishan@cs.cornell.edu

Claire Cardie
Department of Computer Science
Cornell University
cardie@cs.cornell.edu

Abstract

Extracting opinion expressions from text is usually formulated as a token-level sequence labeling task tackled using Conditional Random Fields (CRFs). CRFs, however, do not readily model potentially useful segment-level information like syntactic constituent structure. Thus, we propose a semi-CRF-based approach to the task that can perform sequence labeling at the segment level. We extend the original semi-CRF model (Sarawagi and Cohen, 2004) to allow the modeling of arbitrarily long expressions while accounting for their likely syntactic structure when modeling segment boundaries. We evaluate performance on two opinion extraction tasks, and, in contrast to previous sequence labeling approaches to the task, explore the usefulness of segment-level syntactic parse features. Experimental results demonstrate that our approach outperforms state-of-the-art methods for both opinion expression tasks.

1 Introduction

Accurate opinion expression identification is crucial for tasks that benefit from fine-grained opinion analysis (Wiebe et al., 2005): e.g., it is a first step in characterizing the sentiment and intensity of the opinion; it provides a textual anchor for identifying the opinion holder and the target or topic of an opinion; and these, in turn, form the basis of opinion-oriented question answering and opinion summarization systems. In this paper, we focus on opinion expressions as defined in Wiebe et al. (2005) — subjective expressions that denote emotions, sentiment, beliefs, opinions, judgments, or other *private states* (Quirk et al., 1985) in text. These include *direct subjective expressions* (DSEs): explicit mentions of private states or speech events expressing private states; and *expressive subjective expressions* (ESEs): expressions that indicate sentiment, emotion, etc. without explicitly conveying them. Following are two example sentences labeled with DSEs and ESEs.

> (1) The International Committee of the Red Cross, [as usual]$_{[\text{ESE}]}$, [has refused to make any statements]$_{[\text{DSE}]}$.
>
> (2) The Chief Minister [said]$_{[\text{DSE}]}$ that [the demon they have reared will eat up their own vitals]$_{[\text{ESE}]}$.

As a type of information extraction task, opinion expression extraction has been successfully tackled in the past via sequence tagging methods: Choi et al. (2006) and Breck et al. (2007), for example, apply conditional random fields (CRFs) (Lafferty et al., 2001) using sophisticated token-level features. In token-level sequence labeling, labels are assigned to single tokens, and the label of each token depends on the current token and the label of the previous token (we consider the usual first-order assumption). Segment-based features — features that describe a set of related contiguous tokens, e.g., a phrase or constituent — might provide critical information for identifying opinion expressions; they cannot, however, be readily and naturally represented in the CRF model.

Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 1335–1345, Jeju Island, Korea, 12–14 July 2012.

Our goal in this work is to extract opinion expressions at the segment level with semi-Markov conditional random fields (semi-CRFs). Semi-CRFs (Sarawagi and Cohen, 2004) are more powerful than CRFs in that they allow one to construct features to capture characteristics of the subsequences of a sentence. They are defined on semi-Markov chains where labels are attached to segments instead of tokens and label dependencies are modeled at the segment-level. Previous work has shown that semi-CRFs outperform CRFs on named entity recognition (NER) tasks (Sarawagi and Cohen, 2004; Okanohara et al., 2006). However, to the best of our knowledge, semi-CRF techniques have not been investigated for opinion expression extraction.

The contribution of this paper is a semi-CRF-based approach for opinion expression extraction that leverages parsing information to provide better modeling of opinion expressions. Specifically, possible segmentations are generated by taking into account likely syntactic structure during learning and inference. As a result, arbitrarily long expressions can be modeled and their boundaries can be influenced by probable syntactic structure. We also explore the impact of syntactic features for extracting opinion expressions.

We evaluate our model on two opinion extraction tasks: identifying direct subjective expressions (DSEs) and expressive subjective expressions (ESEs). Experimental results show that our approach outperforms the state-of-the-art approach for the task by a large margin. We also identify useful syntactic features for the task.

2 Related Work

Previous research to extract direct subjective expressions exists, but is mainly focused on single-word expressions (Wiebe et al., 2005; Wilson et al., 2005; Munson et al., 2005). More recent studies tackle opinion expression extraction at the expression level. Breck et al. (2007) formulate the problem as a token-level sequence labeling problem; their CRF-based approach was shown to significantly outperform two subjectivity-clue-based baselines. Others extend the token-level approach to jointly identify opinion holders (Choi et al., 2006), and to determine the polarity and intensity of the opinion expressions (Choi and Cardie, 2010). Reranking the output of a simple sequence labeler has been shown to further improve the extraction of opinion expressions (Johansson and Moschitti, 2010; Johansson and Moschitti, 2011); importantly, their reranking approach relied on features that encoded syntactic structure. All of the above approaches, however, are based on token-level sequence labeling, which ignores potentially useful phrase-level information.

Semi-CRFs (Sarawagi and Cohen, 2004) are general CRFs that relax the Markovian assumptions to allow sequence labeling at the segment level. Previous work has shown that semi-CRFs are superior to CRFs for NER and Chinese word segmentation (Sarawagi and Cohen, 2004; Okanohara et al., 2006; Andrew, 2006). The task of opinion expression extraction is known to be harder than traditional NER since subjective expressions exhibit substantial lexical variation and their recognition requires more attention to linguistic structure.

Parsing has been leveraged to improve performance for numerous natural language tasks. In opinion mining, numerous studies have shown that syntactic parsing features are very helpful for opinion analysis. A lot of work uses syntactic features to identify opinion holders and opinion topics (Bethard et al., 2005; Kim and Hovy, 2006; Kobayashi et al., 2007; Joshi and Carolyn, 2009; Wu et al., 2009; Choi et al., 2005). Jakob et al. (2010) recently employed dependency path features for the extraction of opinion targets. Johansson and Moschitti (2010; Johansson and Moschitti (2011) also successfully employed syntactic features that indicate dependency relations between opinion expressions for the task of opinion expression extraction. However, as their approach is based on the output of a sequence labeler, these features cannot be encoded to help the learning of the sequence labeler.

3 Approach

We formulate the extraction of opinion expressions as a sequence labeling problem. Unlike previous sequence-labeling approaches to the task (e.g., Breck et al. (2007)), however, we aim to model segment-level, rather than token-level, information. As a result, we explore the use of semi-CRFs, which

can assign labels to segments instead of tokens; hence, features can be defined at the segment level. For example, features like $[\![X \text{ is a verb phrase}]\!]$ can be easily encoded in the model. In the following subsections, we first introduce standard semi-CRFs and then describe our semi-CRF-based approach for opinion expression extraction.

3.1 Semi-CRFs

In semi-CRFs, each observed sentence x is represented as a sequence of consecutive segments $s = \langle s_1, ..., s_n \rangle$, where s_i is a triple $s_i = (t_i, u_i, y_i)$, t_i denotes the start position of segment s_i, u_i denotes the end position, and y_i denotes the label of the segment. Segments are restricted to have positive length less than or equal to a maximum length of L that has been seen in the corpus ($1 \leq u_i - t_i + 1 \leq L$).

Features in semi-CRFs are defined at the segment level rather than the word level. The feature function $g(i, x, s)$ is a function of x, the current segment s_i, and the label y_{i-1} of the previous segment s_{i-1} (we consider the usual first-order Markovian assumption). It can also be written as $g(x, t_i, u_i, y_i, y_{i-1})$. The conditional probability of a segmentation s given a sequence x is defined as

$$p(s|x) = \frac{1}{Z(x)} \exp\left\{\sum_i \sum_k \lambda_k g_k(i, x, s)\right\} \quad (1)$$

where

$$Z(x) = \sum_{s' \in S} \exp\left\{\sum_i \sum_k \lambda_k g_k(i, x, s')\right\}$$

and the set S contains all possible segmentations obtained from segment candidates with length ranging from 1 to the maximum length L.

The correct segmentation s of a sentence is defined as a sequence of entity segments (i.e., the entities to be extracted) and non-entity segments. For example, the correct segmentation of sentence (2) in Section 1 is ⟨(The,NONE),(Chief,NONE),(Minister,NONE), (said,DSE),(that,NONE),(the demon they have reared will eat up their own vitals,ESE),(.,NONE)⟩. Here, non-entity segments are represented as unit-length segments.

3.2 Semi-CRF-based Approach for Opinion Expression Extraction

In this section, we present an extended version of semi-CRFs in which we can make use of parsing information in learning entity boundaries and labels for opinion expression extraction.

As discussed in Section 3.1, the maximum entity length L is fixed during training to generate segment candidates in the standard semi-CRFs. In opinion expression extraction, L is unbounded since opinion expressions may be clauses or whole sentences, which can be arbitrarily long. Thus, fixing an upper bound on segment length based on the observed entities may lead to an incorrect removal of segments during inference. Also note that possible segment candidates are generated based on the length constraint, which means any span of the text consisting of no more than L words would be considered as a possible segment. This would lead to the consideration of implausible segments, e.g., "The Chief" in sentence (2) is an incorrect segment within the multi-word expression "The Chief Minister".

To address these problems, we propose techniques to incorporate parsing information into the modeling of segments in semi-CRFs. More specifically, we construct segment units from the parse tree of each sentence[1], and then build up possible segment candidates based on those units. In the parse tree, each leaf phrase or leaf word is considered to be a segment unit. Each segment unit performs as the smallest unit in the model (words within a segment unit will be automatically assigned the same label). The segment units are highlighted in rectangles in the parse tree example in Figure 1. As the segment units are not separable, we avoid implausible segments, which truncate multi-word expressions. For example, "both ridiculous and", would not be considered a possible segment in our model.

To generate segment candidates for the model, we consider meaningful combinations of consecutive segment units. Intuitively, a sentence is made up of several parts, and each has its own grammatical role or meaning. We define the boundary of these parts based on the parse tree structure. Specifically,

[1]We use the Stanford Parser `http://nlp.stanford.edu/software/lex-parser.shtml` to generate the parse trees.

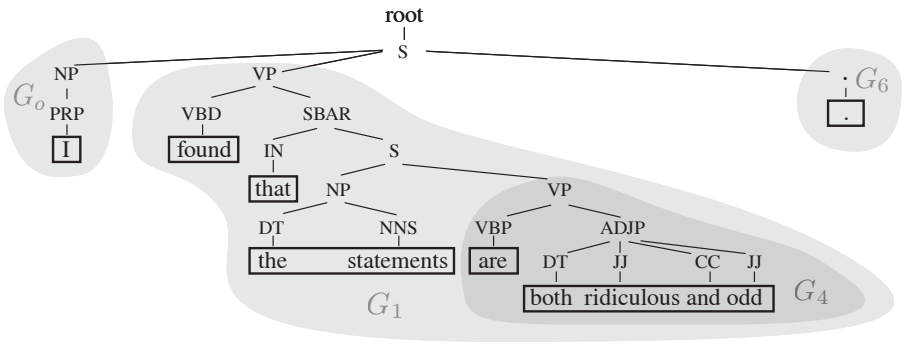

Figure 1: A parse tree example. There are seven segment units in the sentence. The shaded regions correspond to segment groups, where G_i represents the segment group starting from segment unit U_i.

we consider each segment unit to belong to a meaningful group defined by the span of its parent node. Two consecutive segment units are considered to belong to the same group if the subtrees rooted in their parent nodes have the same rightmost child. For example, in Figure 1, segment units "are" and "both ridiculous and odd" belong to the same group, while "I" and "found" belong to different groups.

Algorithm 1 Construction of segment candidates

Input: A training sentence x
Output: A set of segment candidates S

1: Obtain the segment units $U = (U_1, ..., U_m)$ by preorder traversal of the parse tree T, each U_i corresponds to a node in T
2: **for** $i = 1$ to m **do**
3: $j \leftarrow i - 1$
4: **while** $j < m - 1$ and $\text{commonGroup}(U_i, ..., U_{j+1})$ **do**
5: $j \leftarrow j + 1$
6: **for** $k = i$ to j **do**
7: **for** $t = 0$ to $j - k$ **do**
8: $s \leftarrow \text{segment}(U_k, ..., U_{k+t})$
9: $S \leftarrow S \cup s$
10: Return S

Following this idea, we generate possible segment candidates by Algorithm 1. Starting from each segment unit U_i, we first find the rightmost segment unit U_j that belongs to the same group as U_i. Function $\text{commonGroup}(U_i, ..., U_j)$ returns True if $U_i, ..., U_j$ are within the same group (the parent nodes of U_i,...,U_j have the same rightmost child in their subtrees), otherwise it returns False. Then we enumerate all possible combinations of segment units $U_i, ..., U_k$ where $i \leq k \leq j$. $\text{segment}(U_i, ..., U_j)$ denotes the segment obtained by concatenating words in the consecutive segment units U_i,...,U_j. This way, segment candidates are generated without constraints on length and are meaningful for learning entity boundaries.

Based on the generated segment candidates, the correct segmentation for each training sentence can be obtained as follows. For opinion expressions that do not match any segment candidate, we break them down into smaller segments using a greedy matching process. Starting from the start position of the expression, we search for the longest candidate that is contained in the expression, add it to the correct segmentation for the sentence, set the start position to be the next position, and repeat the process. Using this process, the correct segmentation of sentence (2) would be $s = \langle$(The Chief Minister,NONE),(said,DSE),(that,NONE),(the demon they have reared,ESE), (will eat up their own vitals,ESE),(.)$\rangle$. Note that here non-entities correspond to segment units instead of single-word segments in the original semi-CRF model.[2]

After obtaining the set of possible segment candidates and the correct segmentation s for each training sentence, the semi-CRF model can be trained. The goal of learning is to find the optimal parameter λ by maximizing log-likelihood. We use the limited-

[2]There are cases where words within a segment unit have different labels. This may be due to errors by the human annotators or the errors in the parser. In such cases, we consider each word within the segment unit as a segment.

memory BFGS algorithm (Liu and Nocedal, 1989) for optimization in our implementation, where the gradient of the log-likelihood L (corresponding to one instance x) is computed:

$$\frac{\partial L}{\partial \lambda_k} = \sum_i g_k(x, t_i, u_i, y_i, y_{i-1}) - \sum_{s' \in S} \sum_{y,y'} \sum_j g_k(x, t'_j, u'_j, y, y') p(y, y'|x) \quad (2)$$

where S is all possible segmentations consisting of the generated segment candidates, $p(y, y'|x)$ is the probability of having label y for the current segment s'_j (with boundary (t'_j, u'_j)) and label y' for the previous segment s'_{j-1}.

We use a forward-backward algorithm to compute the marginal distribution $p(y, y'|x)$ and the normalization factor $Z(x)$ efficiently. For inference we seek the best segmentation $s^* = \arg\max_s p(s|x)$, where $p(s|x)$ is defined by Equation 1. We implement efficient inference using an extension of Viterbi algorithm to segments. In particular, define $V(j, y)$ as the largest unnormalized probability of $p(s_{1:j}|x)$ with label y at the ending position j. Then we have

$$V(j, y) = \max_{(i,j) \in s_{:,j}} \max_{y'} \phi(x, i, j, y, y') V(i-1, y')$$

where

$$\phi(x, i, j, y, y') = \exp\left\{\sum_k \lambda_k g_k(x, i, j, y, y')\right\}$$

and $s_{:,j}$ denotes the set of the generated segment candidates ending at position j. The best segmentation can be obtained from tracing the path of $\max_y V(n, y)$.

3.3 Features

Here we described the features used in our model. Very generally, we include CRF-style features that are segment-level extensions of the token-level features. We also include new segment-level features that can be naturally represented in semi-CRFs but not CRFs.

For CRF-style features, we consider the string representation of the current word, its part-of-speech, and a dictionary-derived feature, which is based on a subjectivity lexicon provided by Wilson et al. (2005). The lexicon consists of a set of words that can act as strong or weak cues to subjectivity. If the current word appears as an entry in the lexicon, then a feature *strong* or *weak* will be fired if the entry is of that strength. These features have been successfully employed in previous work (Breck et al., 2007). To employ them in our model, we simply extend the feature definition to the segment level. For example, a token-level feature $[\![x$ is *great* $]\!]$ will be extended to a segment-level feature $[\![s$ contains *great* $]\!]$.

Previous work on semi-CRFs has explored features such as the length of the segment, the position of the segment in the current segmentation (at the beginning or at the end), indicators for the start word and end word within the segment, and indicators for words before and after the segment. These features have been shown useful for the task of NE recognition (Sarawagi and Cohen, 2004; Okanohara et al., 2006). However, we only found the position of the segment to be helpful for the extraction of opinion expressions, probably due to the lack of patterns in the length distribution and word choices of opinion expressions.

Besides the above features, we design new segment-level syntactic features to capture the syntactic patterns of opinion expressions. Syntactic patterns are often used to identify useful information in information extraction tasks. In our task, we found that the majority of opinion expressions involve verb phrases.[3] For example, "was encouraged", "expressed goodwill", "cannot accept" are all within a VP constituent. To capture such structural preferences, we define several syntax-based parse features for VP-related constituents.[4]

Let VPROOT denote a VP constituent whose parent node is not VP, and let VPLEAF denote a VP constituent whose children nodes are non-VP. Denote the head of VPLEAF as the predicate, and its next segment unit as the argument. If a segment consists of words in the VP nodes visited by the preorder

[3]The percentages of opinion expressions involving VP/NP/PP are 64.13%/18.43%/5.92% for DSEs and 43.22%/24.99%/11.77% for ESEs in the data set we used.

[4]We also conducted experiments with NP and PP-related features, and could not find any performance improvement for the tasks.

traversal from a VPROOT to a VPLEAF, then we refer to it as a verb-cluster segment. If a segment consists of a verb cluster and the argument in VPLEAF, we consider it as a VP segment. The following features are defined for verb-cluster segments and VP segments.

VPcluster: Indicates whether or not the segment matches the verb-cluster structure.

VPpred: A feature of the syntactic category and the word of the head of VPLEAF. The head of VPLEAF is the predicate of the verb phrase, which may encode some intention of opinions in the verb phrase. For example, if "warned" is the head of VPLEAF rather than "informed", the chance of the segment being an opinion expression increases.

VParg: A feature of the syntactic category and the head word of the argument in VPLEAF. For example, the noun phrase "a negative stand" is the argument of the predicate "take" in the verb phrase "take a negative stand". The argument in the verb phrase (could be a noun phrase, adjectival phrase or prepositional phrase) may convey some relevant information for identifying opinion expressions.

VPsubj: Whether the verb clusters or the argument in the segment contains an entry from the subjectivity lexicon. For example, the word "negative" is in the lexicon, so the segment "take a negative stand" has a feature ISVPSUBJ.

4 Experiments

For evaluation, we use the MPQA 1.2 corpus (Wiebe et al., 2005)[5], a widely used data set for fine-grained opinion analysis. It contains 535 news articles, a total of 11,114 sentences with subjectivity-related annotations at the phrase level. We focus on the task of extracting two types of opinion expressions: direct subjective expressions (DSEs) and expressive subjective expressions (ESEs). Table 1 shows some statistics of the corpus. As in prior research that uses the corpus, we set aside the standard 135 documents as a development set and use 400 documents as the evaluation set. All experiments employ 10-fold cross validation on the evaluation set, and the average over all runs is reported.

[5] Available at http://www.cs.pitt.edu/mpqa/.

	DSEs	ESEs
Sentences with opinions(%)	55.89	57.93
TotalNum	9746	11730
MaxLength	15	40
Length ≥ 1 (%)	43.38	71.65
Length ≥ 4 (%)	9.44	35.01

Table 1: Statistics of opinion expressions in the MPQA Corpus.

4.1 Evaluation Metrics

We use precision, recall, and F-measure to evaluate the quality of the model. Precision is defined as $\frac{|C \cap P|}{|P|}$ and recall, as $\frac{|C \cap P|}{|C|}$, where C and P are the sets of correct and predicted expression spans, respectively. F-measure is computed as $\frac{2PR}{P+R}$. Because the boundaries of opinion expressions are hard to define even for human annotators (Wiebe et al., 2005), previous research mainly focused on soft precision and recall measures for performance evaluation. Breck et al. (2007) introduced an overlap measure, which considers a predicted expression to be correct if it overlaps with a correct expression. We refer to this metric as *Binary Overlap*. Johansson and Moschitti (2010) provides a stricter measure that computes the proportion of overlapping spans: if a correct expression s overlaps with a predicted expression s', the overlap contributes value $\frac{|s \cap s'|}{|s'|}$ to $|C \cap P|$ instead of value 1. We refer to this metric as *Proportional Overlap*. To compare with previous work, we present our results according to both metrics.

4.2 Baseline Methods

As a baseline, we use the token-level CRF-based approach of Breck et al. (2007) applied to the MPQA dataset. We employ a very similar, but not identical set of features: indicators for specific words at the current location and neighboring words in a $[-4, +4]$ window, part-of-speech features, and opinion lexicon features for tokens that are contained in the subjectivity lexicon (see Section 3.3). We do not include WordNet, Levin's verb categorization, and FrameNet features.

We also include two variants of standard CRFs as baselines: segment-CRF and syntactic-CRF. They incorporate segmentation information into standard CRFs without modifying the Markovian assump-

	DSE Extraction			ESE Extraction		
Method	**Precision**	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
CRF	82.83	49.38	61.87	78.56	43.57	56.05
segment-CRF	82.52	51.48	63.41	78.90	44.46	56.88
syntactic-CRF	82.48	49.09	61.55	78.41	43.39	55.95
semi-CRF	66.67	74.13	70.20	71.21	57.41	63.57
new-semi-CRF	67.72^{**}	74.33	70.87^{*}	73.57^{***}	57.63	64.74^{**}
semi-CRF(w/ syn)	64.86	74.10	69.17	70.68	56.61	62.87
new-semi-CRF(w/ syn)	70.12^{***}	74.74^{*}	$\mathbf{72.36}^{***}$	73.61^{***}	59.27^{***}	$\mathbf{65.67}^{***}$

Table 2: Results for extracting opinion expressions with Binary-Overlap metric. (w/ syn) indicates the inclusion of syntactic parse features VPpre, VParg and VPsubj. Results of new-semi-CRF that are statistically significantly greater than semi-CRF according to a two-tailed t-test are indicated with $^{*}(p < 0.1)$, $^{**}(p < 0.05)$, $^{***}(p < 0.005)$. T-test results are also shown for new-semi-CRF(w/ syn) versus semi-CRF(w/ syn).

	DSE Extraction			ESE Extraction		
Method	**Precision**	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
CRF	77.91	46.45	58.20	67.72	37.55	48.31
segment-CRF	77.86	48.58	59.83	68.03	38.34	49.04
syntactic-CRF	77.73	46.27	58.01	67.80	37.60	48.37
semi-CRF	60.38	68.34	64.11	57.30	46.20	51.16
new-semi-CRF	62.50^{**}	68.59^{*}	65.41^{*}	61.69^{***}	47.44^{**}	53.63^{***}
semi-CRF(w/ syn)	58.69	67.80	62.92	57.09	45.63	50.72
new-semi-CRF(w/ syn)	65.52^{***}	68.91^{***}	$\mathbf{67.17}^{***}$	61.66^{***}	48.77^{***}	$\mathbf{54.47}^{***}$

Table 3: Results for extracting opinion expressions with Proportional-Overlap metric. Notation is the same as above.

tion. Segment-CRF treats segment units obtained from the parser as word tokens. For example, in Figure 1, the segment units *the statement* and *both ridiculous and odd* will be treated as word tokens. Syntactic-CRF encodes segment-level syntactic information in a standard token-level CRF as input features. We consider the VP-related segment features introduced in Section 3.3. VPPRE and VPARG are added to the head word of the corresponding verb phrase, and VPSUBJ and VPCLUSTER are added to each token within the corresponding segment.

Another baseline method is the original semi-CRF model (Sarawagi and Cohen, 2004). To the best of our knowledge, our work is the first to explore the use of semi-CRFs on the extraction of opinion expressions. They are considered to be more powerful than CRFs since they allow information to be represented at the expression level. The model requires an input of the maximum entity length. We set it to 15 for DSE and 40 for ESE. For segment features, we used the same features as in our approach (see Section 3.3).

4.3 Results

Table 2 and Table 3 show the results of DSE and ESE extraction using two different metrics. The standard token-based CRF baseline of Breck et al. (2007) is labeled **CRF**; the original semi-CRF baseline is labeled **semi-CRF**; and our extended semi-CRF approach is labeled **new-semi-CRF**. For semi-CRF and new-semi-CRF, the results were obtained using two different settings of features: the basic feature set includes features described in Section 3.3 excluding the segment-level syntactic features. In the second feature setting (labeled as **w/ syn** in the tables), we further augment the basic features with the syntactic parse features.

Using the basic features, we observe that semi-CRF-based approaches significantly outperform CRF and its two variants segment-CRF and syntactic-CRF in F-Measure on both DSE and ESE extraction, and new-semi-CRF achieves the best results. By simply incorporating the segmentation prior into the standard CRF, segment-CRF achieves a slight improvement over standard CRF, but the results are still worse than those of semi-CRF and new-semi-CRF. However, adding segment-level

	DSE Extraction			ESE Extraction		
Feature set	**Precision**	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
Basic	67.72	74.33	70.87	73.57	57.63	64.74
Basic+VPpre	70.88	71.44	71.16	73.20	58.20	64.85
Basic+VParg	70.12	74.03	72.02	73.05	58.20	64.79
Basic+VPcluster	70.08	72.94	71.48	73.06	58.45	64.94
Basic+VPsubj	70.04	72.34	71.17	73.31	58.53	65.09
Basic+VPpre+VPsubj	70.91	72.54	71.72	73.61	58.29	65.07
Basic+VParg+VPsubj	70.45	73.53	71.96	74.45	57.80	65.07
Basic+VPpre+VParg+VPsubj	70.12	74.74	**72.36**	73.61	59.27	**65.67**
Basic+VPcluster+VPpre+VParg+VPsubj	70.91	72.54	71.72	72.84	58.45	64.86

Table 4: Effect of syntactic features on extracting opinion expressions with Binary-Overlap metric

syntactic features into standard CRF yields slightly reduced performance. This is not surprising as encoding segment-level information into the token-level CRF is not natural. These experiments indicate that simply encoding segmentation information into standard CRF cannot result in large performance gains. The promising F-measure results obtained by semi-CRF and new-semi-CRF confirm that relaxing the Markovian assumption on segments leads to better modeling of opinion expressions. We can also see that new-semi-CRF consistently outperforms the original semi-CRF model. This further confirms the benefit of taking into account syntactic parsing information in modeling segments. In Table 3, we observe the same general results trend as in Table 2. The scores are generally lower since the metric *Proportional Overlap* is stricter than *Binary Overlap*.

We also study the impact of syntactic parse features on the semi-Markov CRF models. Here we consider the combination of VPPRE, VPARG and VPSUBJ since they turned out to be the most helpful features for our tasks. Interestingly, we found that after incorporating the syntactic parse features, performance decreases on semi-CRF. This indicates that syntactic information does not help if learning and inference take place on segment candidates generated without accounting for parse information. In contrast, our approach incorporates syntactic parsing information in modeling segments and meaningful segmentations. We can see in Tables 2 and 3 that adding syntactic features successfully boosts the performance of our approach.

To further explore the effect of the syntactic features, we include the results of our model with different configurations of syntactic features in Table 4 (here we focus on the *Binary Overlap* metric as the results with *Proportional Overlap* demonstrate a similar conclusion). We can see that using the basic features and the combination of VPPRE, VPARG and VPSUBJ yields the best results for both DSE and ESE extraction. For DSE extraction, combining these three features improves the precision noticeably from 67.72% to 70.12% while the recall slightly improves. This indicates that VP-related structural information is very helpful for modeling segments as DSEs. However, this trend is not so clear for ESE extraction. This may be due to the fact that DSEs often involve verb phrases while ESEs are represented via a variety of syntactic structures.

Comparison with previous work. In Table 5, we compare our results to the previous work on opinion expression extraction (here we also focus on the *Binary Overlap* metric due to the similar trend demonstrated by the *Proportional Overlap* metric). Breck et al. (2007) presents the state-of-the-art sequence labeling approach on the tasks of DSE and ESE extraction. Their best results are shown as **Breck et al. Baseline** in the table. Johansson and Moschitti (2010) use a reranking technique on the best k outputs of a sequence labeler to further improve their sequence labeling results on the task of extracting DSEs, ESEs and OSEs (Objective Speech Events) (we don't consider OSEs here). Results using our re-implementation of their approach using SVM^{struct} (Tsochantaridis et al., 2004) on the output of CRF are labeled **CRF+Reranking Baseline** in the table. We use the same features and

parameter settings as in their approach. **Our approach+Reranking** are results obtained by applying the reranking step on the output of our new-semi-CRF approach.

We can see that our approach outperforms the Breck et al. Baseline on both DSE extraction and ESE extraction in spite of the fact that we do not use their WordNet, Levin's verb categorization, and FrameNet features. The CRF+Reranking Baseline does provide a performance increase over the the baseline CRF results, but overall it cannot beat the other methods since the CRF baseline is very low. As one might expect, reranking also succeeds in boosting the performance of new-semi-CRF, achieving the best performance on F-measure for both DSE and ESE extraction. Note that the interannotator agreement results for these two tasks are 75% for DSE and 72% for ESE using a similar metric to *Binary Overlap*. Our results are much closer to these interannotator scores than previous systems especially for DSEs.

Task	Method	F-measure
DSE Extraction	Breck et al. Baseline	70.65
	CRF+Reranking Baseline	63.87
	Our approach	72.36
	Our approach+Reranking	**73.12**
ESE Extraction	Breck et al. Baseline	63.43
	CRF+Reranking Baseline	58.21
	Our approach	65.67
	Our approach+Reranking	**67.01**

Table 5: Comparison of our work with previous work on opinion expression extraction using the Binary-Overlap metric

4.4 Discussion

We note that our new-semi-CRF approach outperforms the original semi-CRF w.r.t. both precision and recall, but compared to CRF, our approach yields a clear improvement on recall but not on precision. An error analysis helps explain why. We found that our semi-CRF approach predicted almost the same number of DSEs as the gold standard labels while CRF only predicted half of them (for ESE extraction, the trend is similar). With more predicted entities, the precision is sacrificed but recall is boosted substantially, and overall we see an increase in F-measure.

Looking further into the errors, we found several mistakes that could potentially be fixed to yield better a precision score. Some errors were due to the false prediction of speech events like "said" or "told" as DSEs in cases where they actually just introduced statements of fact without expressing any private state. Adding features to distinguish such cases should help improve performance. Other errors were due to inadequate modeling of the context surrounding the expressions. For example, "enjoy a relative advantage" was falsely predicted as an ESE. If incorporating information about the subject of this verb phrase which is "products", this mistake could be avoided since "products" cannot hold or express private state. We also noticed some errors caused by inaccurate parsing and hope to study ways to account for these in our approach as future work.

By comparing the extraction results across different methods, we see that full parsing provides many benefits for modeling segment boundaries and improving the prediction precision for opinion expression extraction. For example, given the sentence, "... who are living [a lot better]$_{[\text{ESE}]}$ ...", both CRF and the original semi-CRF extract "lot better" as an ESE, while our approach correctly extracts "a lot better" as an ESE. And we also found many cases where the original semi-CRF cannot extract the opinion expressions while our approach can. Another benefit of utilizing parsing is to speed up learning and inference. Although in theory, the computational cost of parsing is $O(g \times n^3)$ where g is the grammar size and n is the sentence length while the cost of semi-CRFs is $O(K^2 \times L \times n)$ where K is the number of labels and L is the maximum entity length, feature extraction overhead and the potentially large number of learning iterations in parameter optimization may lead to a long training time for semi-CRFs. In our experiments on the MPQA data set, our machine with Intel Core 2 Duo CPU and 4GB RAM took 2 hours to fully parse 11,114 sentences using the Stanford Parser, and also 2 hours to train the standard semi-CRF. With the parsing information, our semi-CRF-based approach is able to finish training in 15 minutes. As full parsing would be expensive when the average sentence length is very large, it would be interesting to study how to utilize parsing with less cost in our task.

5 Conclusion

In this paper we propose a semi-CRF-based approach for extracting opinion expressions that takes into account during learning and inference the structural information available from syntactic parsing. Our approach allows opinion expressions to be identified at the segment level and their boundaries to be influenced by their probable syntactic structure. Experimental evaluations show that our model outperforms the best existing approaches on two opinion extraction tasks. In addition, we identify useful syntactic parse features for these tasks that have not been explored in previous work. Our error analysis indicates that adding additional features that account for subjectivity cues in the local context might further improve the performance. In future work, we hope to explore better ways of utilizing parsing information with less cost. Also, we will apply our model to additional opinion analysis tasks such as fine-grained opinion summarization and relation extraction.

6 Acknowledgement

This work was supported in part by National Science Foundation Grants IIS-1111176 and IIS-0968450, and by a gift from Google. We thank Nikos Karampatziakis, Igor Labutov, Veselin Stoyanov, Ainur Yessenalina and Jason Yosinski for their helpful comments.

References

Galen Andrew. 2006. A hybrid Markov/semi-Markov conditional random field for sequence segmentation. In Proceedings of EMNLP '06.

Steven Bethard, Hong Yu, Ashley Thornton, Vasileios Hatzivassiloglou, and Dan Jurafsky. 2005. Extracting opinion propositions and opinion holders using syntactic and lexical cues. In Shanahan, James G., Yan Qu, and Janyce Wiebe, editors, Computing Attitude and Affect in Text: Theory and Applications.

Eric Breck, Yejin Choi, and Claire Cardie. 2007. Identifying expressions of opinion in context. IJCAI'07.

Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005. Identifying sources of opinions with conditional random fields and extraction patterns. In Proceedings of HLT '05.

Yejin Choi, Eric Breck, and Claire Cardie. 2006. Joint extraction of entities and relations for opinion recognition. In Proceedings of EMNLP '06.

Yejin Choi and Claire Cardie. 2010. Hierarchical sequential learning for extracting opinions and their attributes. In Proceedings of ACL 2010, Short Papers.

Richard Johansson and Alessandro Moschitti. 2010. Syntactic and semantic structure for opinion expression detection. In Proceedings of CoNLL '10.

Niklas Jakob and Iryna Gurevych. Extracting opinion targets in a single- and cross-domain setting with conditional random fields. In Proceedings of EMNLP' 10.

Mahesh Joshi and Penstein-Ros'e Carolyn. 2009. Generalizing dependency features for opinion mining. In Proceedings of ACL/IJCNLP 2009, Short Papers Track.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In Proceedings of ACL '03.

Soo-Min Kim and Eduard Hovy. 2006. Extracting opinions, opinion holders, and topics expressed in online news media text. In Proceedings of the ACL Workshop on Sentiment and Subjectivity in Text.

Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto. 2007. Extracting aspect-evaluation and aspect-of relations in opinion mining. In Proceedings of EMNLP-CoNLL-2007.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In Proceedings of ICML '01.

Dong C. Liu and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. Mathematical Programming B 45(3): 503-528.

M Arthur Munson, Claire Cardie, and Rich Caruana. Optimizing to arbitrary NLP metrics using ensemble selection. In HLT-EMNLP05, 2005.

Daisuke Okanohara, Yusuke Miyao, Yoshimasa Tsuruoka, and Jun'ichi Tsujii. Improving the scalability of semi-Markov conditional random fields for named entity recognition. In Proceedings of ACL'06.

Randolph Quirk, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. A comprehensive grammar of the English language. New York: Longman, 1985.

Richard Johansson and Alessandro Moschitti. Extracting Opinion Expressions and Their Polarities - Exploration of Pipelines and Joint Models. In Proceedings of ACL '11, Short Paper.

Ellen Riloff and Janyce M Wiebe. Learning extraction patterns for subjective expressions. In Proceedings of EMNLP 2003.

Sunita Sarawagi and William W. Cohen. 2004. Semi-Markov Conditional Random Fields for Information Extraction. In Proceedings of NIPS 2004.

Charles Sutton and Andrew McCallum. An Introduction to Conditional Random Fields. Foundations and Trends in Machine Learning (FnT ML), 2010.

Janyce Wiebe, Theresa Wilson , and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. Language Resources and Evaluation, volume 39, issue 2-3, pp. 165-210.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In Proceedings of HLT '05.

Theresa Wilson, Paul Hoffmann, Swapna Somasundaran, Jason Kessler, Janyce Wiebe, Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. OpinionFinder: A system for subjectivity analysis. EMNLP 2005. Demo abstract.

Yuanbin Wu, Qi Zhang, Xuanjing Huang, and Lide Wu. Phrase dependency parsing for opinion mining. In Proceedings of EMNLP 2009.

Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemi Altun. Support Vector Learning for Interdependent and Structured Output Spaces. In Proceedings of ICML 2004.