Weak Semi-Markov CRFs for NP Chunking in Informal Text

Aldrian Obaja Muis and **Wei Lu**
Singapore University of Technology and Design
{aldrian_muis,luwei}@sutd.edu.sg

Abstract

This paper introduces a new annotated corpus based on an existing informal text corpus: the NUS SMS Corpus (Chen and Kan, 2013). The new corpus includes 76,490 noun phrases from 26,500 SMS messages, annotated by university students. We then explored several graphical models, including a novel variant of the semi-Markov conditional random fields (semi-CRF) for the task of noun phrase chunking. We demonstrated through empirical evaluations on the new dataset that the new variant yielded similar accuracy but ran in significantly lower running time compared to the conventional semi-CRF.

1 Introduction

Processing user-generated text data is getting more popular recently as a way to gather information, such as collecting facts about certain events (Ritter et al., 2015), gathering and identifying user profiles (Layton et al., 2010; Li et al., 2014; Spitters et al., 2015), or extracting information in open domain (Ritter et al., 2012; Mitchell et al., 2015).

Most recent work focus on the texts generated through Twitter, which, due to the design of Twitter, contain a lot of announcement-like messages mostly intended for general public. In contrast, SMS was designed as a way to communicate short personal messages to a known person, and hence SMS messages tend to be more conversational and more informal compared to tweets.

As conversational texts, SMS data often contains references to named entities such as people and locations relevant to certain events. Recognizing those references will be useful for further NLP tasks. One way to recognize those named entities is to first create a list of candidates, which can be further filtered to get the desired named entities. Nadeau (Nadeau and Sekine, 2007) lists several methods that work upon candidates for NER. As all named entities are nouns, recognizing noun phrases (NP) is therefore a task that can be potentially useful for further steps in the NLP pipeline to build upon. Figure 1 shows an example SMS message within which noun phrases are highlighted. As can be seen from this example, recognizing the NP information on such a dataset presents some additional challenges over conventional NP recognition tasks. Specifically, the texts are highly informal and noisy, with misspelling errors and without grammatical structures. The correct casing and punctuation information is often missing. The lack of spaces between adjacent words makes the detection of NP boundaries more challenging.

> *Hmm Dr teh says the research presentation should still prepare, butshe's not to sure whether they'd time to present*

Figure 1: Sample SMS, with NPs underlined

Furthermore, the lack of available annotated data for such informal datasets prevents researchers from understanding what effective models can be used to resolve the above issues. In this work, we focus on tackling these issues while making the following two main contributions:

- We build a new corpus of SMS data that is fully annotated with noun phrase information.
- We propose and build a new variant of semi-Markov CRF (Sarawagi and Cohen, 2004) for

the task of NP chunking on our corpus, which is faster and yields a performance similar to the conventional semi-Markov CRF models.

2 NP-annotated SMS Corpus

Our text corpus comes from the NUS SMS Corpus (Chen and Kan, 2013), containing 55,835 SMS messages from university students, mostly in English. We used the 2011 version of the corpus, containing 45,718 messages, as it is more relevant to modern phone models using full keyboard layout.

We note that there are a small portion of the messages written in non-English language, such as Tamil and Chinese. As we are focusing on English, we excluded messages written by non-native English speakers based on the metadata (21.3% of all messages). We also excluded messages which contain only one word (6.1%) and we remove duplicate messages (8.1%). [1]

We assigned the remaining 27,700 messages to 64 university students who conduct annotations, each annotating 500 with 100 messages co-annotated by two other annotators. After manual verification we excluded annotations with low quality from 3 students. We used the resulting 26,500 messages as our dataset. The students were asked to annotate the top-level noun phrases found in each message using the BRAT rapid annotation tool[2], where they were instructed to highlight character spans to be marked as noun phrases. The number of noun phrases per message can be found in Table 1.

Due to the noisy nature of SMS messages, there may not be proper capitalization or punctuation, and in some cases there might be missing spaces between words. Figure 1 shows a sample SMS message taken from the corpus. We can see that "Dr teh" is not properly capitalized and "she" in "butshe's" is missing spaces around it. NPs which do not have clear boundaries (*improper* NPs) constitutes 4.0% of all NPs.

We then use this dataset to evaluate some models on base NP chunking task, where, given a text, the system should return a list of character spans denoting the noun phrases found in the text.

[1]We also manually excluded some messages (ID 1017-4016) which are mostly not written in English (4.0% of all messages).

[2]`http://brat.nlplab.org`

	#SMS	#NPs	#*improper*	#tokens
total	26,500	76,490	3,066 (4.0)	359,009
train	21,200	61,212	2,406 (3.9)	287,590
dev	2,650	7,617	338 (4.4)	35,470
test	2,650	7,661	322 (4.2)	35,949

Table 1: Number of messages, NPs, number of *improper* NPs (as percentage in brackets), which are NPs that do not have clear boundaries, and number of tokens.

3 Models

In this paper, we will build our models based on a class of discriminative graphical models, namely conditional random fields (CRFs) (Lafferty et al., 2001), for extracting NPs. The edges in the graph represents the dependencies between states and the features are defined over each edge in the graph. Though CRFs are undirected graphical models, we can use directed acyclic graphs with a root, a leaf, and some inner nodes to represent label sequences[3]. A path in the graph from the root to the leaf represents one possible label assignment to the input. In the labeled instance, there will be only one single path from the root to the leaf, while for the unlabeled instance, the graph will compactly encode all possible label assignments. The learning procedure is essentially the process that tries to tune the feature weights such that the true structures get assigned higher weights as compared to all other alternative structures in the graph.

In general, a CRF tries to maximize the following objective function:

$$\mathcal{L}(\mathcal{T}) = \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{T}} \left[\sum_{e\in\mathcal{E}(\mathbf{x},\mathbf{y})} \mathbf{w}^T \mathbf{f}(e) - \log Z_{\mathbf{w}}(\mathbf{x}) \right] - \lambda ||\mathbf{w}||^2 \quad (1)$$

where $\mathcal{T}$ is the training set, $(\mathbf{x}, \mathbf{y})$ is a training instance consisting of the sentence $\mathbf{x}$ and the label sequence $\mathbf{y} \in \mathcal{Y}^n$ for a label set $\mathcal{Y}$, $\mathbf{w}$ is the feature weight vector, $\mathcal{E}(\mathbf{x}, \mathbf{y})$ is the set of edges which defines the input $\mathbf{x}$ labeled with the label sequence $\mathbf{y}$, $\mathbf{f}(e)$ is the feature vector of the edge e, $Z_{\mathbf{w}}(\mathbf{x})$ is the normalization term which sums over all possible paths from the root to the leaf node, and λ is the regularization parameter.

[3]Extension to directed hypergraphs is possible. See (Lu and Roth, 2015).

The set of edges and features defined in each model affects the feature expectation and the normalization term. Computation of the normalization term, being the highest in time complexity, will determine the overall complexity of training the model. The set of edges and the normalization term in each model will be described in the following sections.

3.1 Linear CRF

A linear-chain CRF, or linear CRF is a standard version of CRF which was introduced in Lafferty et al. (2001), where each word in the sentence is given a set of nodes representing the possible labels, and edges are present between any two nodes from adjacent words, forming a trellis graph. Here we consider only the first-order linear CRF.

The normalization term $Z_{\mathbf{w}}(\mathbf{x})$ is calculated as:

$$Z_{\mathbf{w}}(\mathbf{x}) = \sum_{\mathbf{y}} \exp \sum_{(y',y,i)\in\mathcal{E}(\mathbf{x},\mathbf{y})} \mathbf{w}^T \mathbf{f}_{\mathbf{x}}(y', y, i) \qquad (2)$$

where $\mathbf{f}_{\mathbf{x}}(y', y, i)$ represents the feature vector on the edge connecting state y' at position $i-1$ to state y at position i. The time complexity of the inference procedure for this model is $O(n\,|\mathcal{Y}|^2)$.

3.2 Semi-CRF

In semi-CRF (Sarawagi and Cohen, 2004), in addition to the edges defined in linear CRF, there are additional edges from a node to all nodes up to L next words away, representing a segment within which the words will be labeled with a single label.

The normalization term $Z_{\mathbf{w}}(\mathbf{x})$ is calculated as:

$$Z_{\mathbf{w}}(\mathbf{x}) = \sum_{\mathbf{y}\in\mathcal{Y}^n} \exp \sum_{(y',y,i-k,i)\in\mathcal{E}(\mathbf{x},\mathbf{y})} \mathbf{w}^T \mathbf{g}_{\mathbf{x}}(y', y, i-k, i) \qquad (3)$$

where $\mathbf{g}_{\mathbf{x}}(y', y, i-k, i)$ represents the feature vector on the edge connecting state y' at position $i-k$ to state y at position i. The time complexity for this model is $O(nL\,|\mathcal{Y}|^2)$.

3.3 Weak Semi-CRF

Note that in semi-CRF, each node is connected to $L \times |\mathcal{Y}|$ next nodes. Intuitively, the model tries to decide the next segment length and type at the same time. We now propose a weaker variant that makes the two decisions separately by restricting each node to connect to either only the nodes of the same label up to L next words away, or to all the nodes only in the next word. We call this variant *Weak Semi-CRF*.

To implement this, we need to split the original nodes into Begin and End nodes, representing the start and end of a segment. The End nodes connect only to the very next Begin nodes of any label, while the Begin nodes connect only to the End nodes of same label up to next L words. We denote the set of the earlier edges as $\mathcal{E}_A(\mathbf{x}, \mathbf{y})$ and the latter edges as $\mathcal{E}_J(\mathbf{x}, \mathbf{y})$. The normalization term $Z_{\mathbf{w}}(\mathbf{x})$ is then:

$$Z_{\mathbf{w}}(\mathbf{x}) = \sum_{\mathbf{y}\in\mathcal{Y}^n} \exp \Bigg(\sum_{(y',y,i)\in\mathcal{E}_A(\mathbf{x},\mathbf{y})} \mathbf{w}^T \mathbf{f}_{\mathbf{x}}(y', y, i) + \sum_{(y,i-k,i)\in\mathcal{E}_J(\mathbf{x},\mathbf{y})} \mathbf{w}^T \mathbf{g}_{\mathbf{x}}(y, i-k, i) \Bigg) \qquad (4)$$

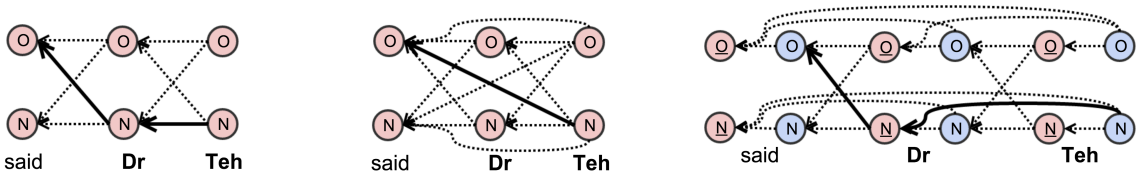
where $\mathbf{g}_{\mathbf{x}}(y, i-k, i)$ represents the feature vector on the edge connecting the Begin node with state y at position $i-k$ to the End node with the same state y at position i. Note that, different from the $\mathbf{g}_{\mathbf{x}}$ function defined in Equation (3), this new $\mathbf{g}_{\mathbf{x}}$ function is defined over a single (current) y label only, making the time complexity $O(n\,|\mathcal{Y}|^2 + nL\,|\mathcal{Y}|)$. Theoretically this model is slightly more efficient than the conventional semi-CRF model.

Unlike conventional (first-order) semi-Markov CRF, this new model does not allow us to capture the dependencies between one segment and its adjacent segment's label information. We argue that, however, such dependencies can be less crucial for our task. We will empirically assess this aspect through experiments. Figure 2 illustrates the differences among the three models.

4 Features

In linear CRF, the baseline feature set considers the previous word, current word, and the tag transition.

In semi-CRF, following Sarawagi and Cohen (2004) we put each word which is not part of a noun phrase in its own segment, and put each noun phrase in one segment, possibly spanning over multiple words. Here we set $L = 6$ and ignored NPs with more than six words during training, which is less than 0.5% of all NPs. For each segment, we defined the following features as the baseline: (1)



Linear CRF Semi CRF Weak Semi-CRF

Figure 2: Graphical illustrations of the differences between three models. The bold arrows represent the path in each model to label "Dr Teh" as a noun phrase. For Linear CRF, this is a simplified diagram; in the implementation we used the "BIO" approach to represent text chunks. The underlined nodes in Weak Semi-CRF are the Begin nodes.

indexed words inside current segment, running from the start and from the end of the segment, (2) the word before and after current segment, and (3) the labels of previous segment and current segment.

In weak semi-CRF we use the same feature set as semi-CRF, adjusting the features accordingly where segment-specific features (1) are defined only in the Begin-End edges, and transition features (3) are defined only in the End-Begin edges.

For each model we then add the character prefixes and suffixes up to length 3 for each word (`+a`), Brown cluster (Brown et al., 1992) for current word (`+b`), and word shapes (`+s`). For Brown cluster features we used 100 clusters trained on the whole NUS SMS Corpus. The cluster information is then used directly as a feature.

Word shapes can be considered a generic representation of words that retains only the "shape" information, such as whether it starts with capital letter or whether it contains digits. The Brown clusters and word shapes features are applied to each of the word features described in each model.

5 Experiments

All models were built by us using Java, and were optimized with L-BFGS. Models are all tuned in the development set for optimal λ. The optimal λ values are noted in Table 2.

Since the models that we consider are all word-based [4], we tokenize the corpus using a regex-based tokenizer similar to the `wordpunct_tokenize` function in Python NLTK package. We also included some rules to consider special anonymization tokens in the SMS dataset (Chen and Kan, 2013).

[4]We experimented with character-based models, but they do not perform well. We leave them for future investigations.

	Linear CRF	Semi-CRF	Weak Semi-CRF
`base`	0.125	2.0	2.0
`+s`	0.25	1.0	2.0
`+b`	0.5	1.0	2.0
`+b+s`	0.5	2.0	2.0
`+a`	1.0	2.0	2.0
`+a +s`	2.0	1.0	2.0
`+a+b`	1.0	2.0	2.0
`+a+b+s`	2.0	2.0	2.0

Table 2: Tuned regularization parameter λ from the set $\{0.125, 0.25, 0.5, 1.0, 2.0\}$ for various feature sets. `+a`, `+b`, and `+s` refer to the affix, Brown cluster, and word shape features respectively.

The gold character spans are converted into word labels in BIO format, reducing or extending the character spans as necessary to the closest word boundaries. The converted annotations are regarded as gold word spans. Note that this conversion is lossy due to the presence of *improper* NPs, which makes it impossible for the converted format to represent the original gold standard.

We evaluated the models in the original character-level spans and also in the converted word-level spans, to see the impact of the lossy conversion on the scores. In character-level evaluation, the system output is converted back into character boundaries and compared with the original gold standard, while in the word-level evaluation, the system output is compared directly with the gold word spans. For this reason, we anticipate that the scores in word-level evaluation will be higher than in the character-level evaluation. The results are shown in Table 3. The scores for "Gold" in the character-level evaluation mark the upperbound of word-based models due to the presence of *improper* NPs.

The average time per training iteration on the base models is 1.311s, 2.072s, and 1.811s respectively for Linear CRF, Semi-CRF, and Weak Semi-CRF.

	Character-level Eval.			Word-level Eval.		
	Prec	*Rec*	*F*	*Prec*	*Rec*	*F*
Linear CRF						
base	72.29	70.13	71.19	74.04	71.93	72.97
+s	72.56	70.50	71.52	74.38	72.38	73.36
+b	72.48	71.82	72.15	74.32	73.77	74.04
+b+s	72.90	72.10	72.50	74.70	73.99	74.34
+a	72.56	72.41	72.49	74.66	74.62	74.64
+a +s	72.65	71.93	72.29	74.69	74.07	74.38
+a+b	72.63	72.80	72.71	74.70	75.00	74.85
+a+b+s	72.63	72.74	72.68	74.77	74.99	74.88
Semi-CRF						
base	74.94	73.80	**74.37**	76.50	75.45	75.97
+s	75.14	73.48	74.30	76.81	75.23	76.01
+b	73.95	74.50	74.22	75.82	76.50	76.15
+b+s	73.79	74.08	73.93	75.67	76.09	75.88
+a	74.31	75.08	**74.69**	76.20	77.11	**76.65**
+a +s	74.36	74.49	**74.42**	76.32	76.57	**76.44**
+a+b	74.30	74.88	**74.58**	76.20	76.92	**76.55**
+a+b+s	74.24	74.93	**74.58**	76.23	77.06	**76.64**
Weak Semi-CRF						
base	74.84	73.94	**74.39**	76.47	75.67	76.07
+s	74.84	72.67	73.74	76.50	74.40	75.43
+b	74.13	74.12	74.12	75.97	76.08	76.02
+b+s	74.19	74.21	74.20	76.06	76.19	76.13
+a	74.07	75.13	**74.60**	76.02	77.23	**76.62**
+a +s	74.47	74.49	**74.48**	76.44	76.58	**76.51**
+a+b	74.08	74.57	**74.32**	76.01	76.64	**76.32**
+a+b+s	74.19	74.43	**74.31**	76.15	76.52	**76.33**
Gold	95.96	95.81	95.88	100.0	100.0	100.0

Table 3: Scores on test set (both character-level and word-level evaluation) using optimal λ. `+a`, `+b`, and `+s` refer to the affix, Brown cluster, and word shape features respectively. Best F1 scores are underlined, and values which are not significantly different in 95% confidence interval are in bold

5.1 Discussion

First, we see that the two variants of semi-CRF models perform better compared to the baseline linear CRF model, showing the benefit of using segment features over only single word features.

It is also interesting that, while being a weaker version of the semi-CRF, the weak semi-CRF can actually perform in the same level within 95% confidence interval as the conventional semi-CRF. This shows that some of the dependencies in the conventional semi-CRF do not really contribute to the strength of semi-CRF over standard linear CRF. As noted in Section 3.3, weak semi-CRF makes the decision on the segment type and length separately. This means there is enough information in the local features to decide the segment type and length separately, and so we can remove some combined features while retaining the same performance.

This result, coupled with the fact that the weak semi-CRF requires 12.5% less time than the conventional semi-CRF (1.811s vs 2.072s), shows the potentials of using this weak semi-CRF as an alternative of the conventional semi-CRF. With more label types (here only two), the difference will be larger, since the weak semi-CRF is linear in number of label types, while conventional semi-CRF is quadratic.

6 Related Work

Ritter et al. (2011) previously showed that off-the-shelf NP-chunker performs worse on informal text. Then they trained a linear-CRF model on additional in-domain data, reducing the error up to 22%. However no results on semi-CRF was given.

Semi-CRF has proven effective in chunking tasks. Other variants of semi-CRF models also exist. Nguyen et al. (2014) explored the use of higher-order dependencies to improve the performance of semi-CRF models on synthetic data and on handwriting recognition. They exploited the sparsity of label sequence in order to make the training efficient.

It is also known that feature selection is an important aspect when trying to use semi-CRF models to improve on the linear CRF. Andrew (2006) reported an error reduction of up to 25% when using features that are best exploited by semi-CRF.

7 Conclusion and Future Work

In this paper we present a new NP-annotated SMS corpus, together with a novel variant of the semi-CRF model, which runs in significantly lower time while maintaining similar accuracy on the NP chunking task on the new dataset. Future work includes the application of the weak semi-CRF model to other structured prediction problems, as well as performing investigations on handling other types of informal or noisy texts such as speech transcripts. We make the code and data available for download at `http://statnlp.org/research/ie/`.

Acknowledgements

We would like to thank Alexander Binder, Jie Yang, Dinh Quang Thinh as well as the 64 undergraduate students who helped us with annotations. We would also like to thank the three anonymous reviewers for their helpful comments. This work is supported by SUTD grant SRG ISTD 2013 064 and MOE Tier 1 grant SUTDT12015008. We thank Razvan Bunescu for pointing out an error in Equations 2, 3, and 4 in an earlier version of this paper.

References

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

Peter F. Brown, Peter V. DeSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-Based n-gram Models of Natural Language. *Computational Linguistics*, 18(4):467–479.

Tao Chen and Min-Yen Kan. 2013. Creating a live, public short message service corpus: the NUS SMS corpus. In *Language Resources and Evaluation*, volume 47, pages 299–335. Springer Netherlands.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *International Conference on Machine Learning (ICML)*, pages 282–289.

Robert Layton, Paul Watters, and Richard Dazeley. 2010. Authorship Attribution for Twitter in 140 Characters or Less. In *2010 Second Cybercrime and Trustworthy Computing Workshop*, pages 1–8.

Jiwei Li, Alan Ritter, and Eduard Hovy. 2014. Weakly Supervised User Profile Extraction from Twitter. In *Association for Computational Linguistics*, pages 165–174.

Wei Lu and Dan Roth. 2015. Joint mention extraction and classification with mention hypergraphs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 857–867, Lisbon, Portugal, September. Association for Computational Linguistics.

Tom M. Mitchell, William Cohen, Estevam Hruschka, Partha Talukdar, Justin Betteridge, Andrew Carlson, Bhavana Dalvi Mishra, Matthew Gardner, Bryan Kisiel, Jayant Krishnamurthy, Ni Lao, Kathryn Mazaitis, Thahir Mohamed, Ndapa Nakashole, Emmanouil Antonios Platanios, Alan Ritter, Mehdi Samadi, Burr Settles, Richard Wang, Derry Wijaya, Abhinav Gupta, Xinlei Chen, Abulhair Saparov, Malcolm Greaves, and Joel Welling. 2015. Never-Ending Learning. In *AAAI Conference on Artificial Intelligence*, pages 2302–2310.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Viet Cuong Nguyen, Nan Ye, Wee Sun Lee, and Hai Leong Chieu. 2014. Conditional Random Field with High-order Dependencies for Sequence Labeling and Segmentation. *Journal of Machine Learning Research 2014*, 15:981–1009.

Alan Ritter, Sam Clark, and Oren Etzioni. 2011. Named entity recognition in tweets: an experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Edinburgh.

Alan Ritter, Oren Etzioni, and Sam Clark. 2012. Open Domain Event Extraction from Twitter. In *Proceedings Workshop on Text Mining, ACM International Conference on Knowledge Discovery and Data Mining 2012 (KDD'12)*.

Alan Ritter, Evan Wright, William Casey, and Tom M. Mitchell. 2015. Weakly Supervised Extraction of Computer Security Events from Twitter. In *Proceedings of the 24th International Conference on World Wide Web*, volume i, pages 896–905.

Sunita Sarawagi and William W. Cohen. 2004. Semimarkov conditional random fields for information extraction. In *Advances in Neural Information Processing Systems 17*, pages 1185–1192.

Martijn Spitters, Femke Klaver, Gijs Koot, and Mark van Staalduinen. 2015. Authorship Analysis on Dark Marketplace Forums. In *Proceedings of the IEEE European Intelligence & Security Informatics Conference 2015 (EISIC 2015)*.