UW-CSE at SemEval-2016 Task 10: Detecting Multiword Expressions and Supersenses using Double-Chained Conditional Random Fields

Mohammad Javad Hosseini Noah A. Smith Su-In Lee
Computer Science and Engineering
University of Washington
Seattle, WA 98195, USA
{hosseini,nasmith,suinlee}@cs.washington.edu

Abstract

We describe our entry to SemEval 2016 Task 10: Detecting Minimal Semantic Units and their Meanings. Our approach uses a discriminative first-order sequence model similar to Schneider and Smith (2015). The chief novelty in our approach is a factorization of the labels into multiword expression and supersense labels, and restricting first-order dependencies within these two parts. Our submitted models achieved first place in the closed competition (CRF) and second place in the open competition (2-CRF).

1 Introduction

Schneider and Smith (2015) argued that the problems of segmenting a piece of text into minimal semantic units, and of labeling those units with semantic classes (e.g., supersenses), are intimately connected.

We propose to use a double-chained conditional random field (which we refer to as "2-CRF," an example of a factorial CRF; §3.4) for joint multiword expression identification and supersense tagging. Like other CRFs, 2-CRF is a feature-rich probabilistic model that can represent probabilistic dependencies between features and labels and between the labels of the consecutive words. The 2-CRF models local dependencies between MWE and supersense sequences with two parallel chains of labels, restricting direct interaction between the two to local, single-word positions. Label constraints on tag bigrams ensure a globally consistent tagging.

Our experiments show that 2-CRF outperforms a zero-order baseline, the structured perceptron used by Schneider and Smith (2015), and a conventional CRF (§4). For SemEval 2016 Task 10: Detecting Minimal Semantic Units and their Meanings, we submitted a CRF for the closed condition and a 2-CRF (incompletely trained) for the open condition, achieving first and second place, respectively.

2 Task Description

For completeness, we briefly review the shared task. The shared task training dataset, called "Detecting Minimal Semantic Units and their Meanings" (DiMSUM) (Schneider et al., 2016),[1] consists of sentences with multiword expression (MWE) and supersense annotations. The data combine and harmonize the STREUSLE 2.1 corpus of web reviews (Schneider and Smith, 2015)[2] and Ritter and Lowlands Twitter datasets (Johannsen et al., 2014).[3]

Similar to prior work (Schneider and Smith, 2015), the annotation for MWEs extends the conventional BIO scheme (Ramshaw and Marcus, 1995) to include gappy MWEs with one level of nesting.[4] Segmentations are represented using six tags; the lower-case variants indicate that an expression is within another MWE's gap.

- *O* and *o*: single word expression
- *B* and *b*: the first word of a MWE

[1] https://github.com/dimsum16/dimsum-data/blob/1.5/README.md
[2] http://www.cs.cmu.edu/~ark/LexSem
[3] https://github.com/coastalcph/supersense-data-twitter
[4] Unlike in Schneider and Smith (2015), there is no notion of weak and strong MWEs.

Proceedings of SemEval-2016, pages 856–861,
San Diego, California, June 16-17, 2016.

- *I* and *i*: a word continuing a MWE

We call a tag sequence *valid* if it matches the regular expression $(O|B(o|bi^+|I)^*I^+)^+$. Validity can be ensured using label constraints on tag bigrams (Schneider, 2014).

Each noun or verb expression is also annotated with a supersense; there are 26 supersenses for nouns and 15 for verbs. Only the first word of a MWE receives a supersense tag.

One approach to encoding the MWE and supersense tags is to define an extended label set containing both tags (Schneider and Smith, 2015). This will result in 170 potential labels: *I*, *i*, and each of *B*, *b*, *O* and *o* paired with one of the 41 supersenses and no supersense ($2 + 4 \times 42 = 170$). Only 110 of these are attested in the training data, and these are the combinations our approach considers.

There are 4,799 sentences in the training data. For each token, the dataset provides its offset in the sentence, lemma, POS tag, MWE tag, offset of parent, and supersense label (if applicable).

The blind test set consists of 1,000 sentences from three sources: online reviews from the TrustPilot corpus (Hovy et al., 2015), tweets from the Tweebank corpus (Kong et al., 2014) and TED talk transcripts from the IWSLT MT evaluation campaigns, obtained from the WIT[3] archive (Cettolo et al., 2012).

The shared task has three data conditions: supervised closed, semi-supervised closed, and open. In the supervised closed condition, only the labeled data, the English WordNet lexicon, a provided Brown clustering (Brown et al., 1992) on the 21-million-word Yelp Academic Dataset[5] (Schneider et al., 2014), and any of the ARK Tweet NLP clusters[6] are allowed. The semi-supervised closed condition adds the Yelp Academic Dataset to the resources. The open condition allows the use of any available resources. We have participated in the supervised closed and open conditions. The evaluation is based on F_1 score for MWE identification, supersense labeling, and their combination.[7]

[5] https://www.yelp.com/academic_dataset

[6] http://www.cs.cmu.edu/~ark/TweetNLP/#resources

[7] For details, see http://dimsum16.github.io

3 Models

3.1 Input Features

For the open condition, we use all features introduced in Schneider and Smith (2015): a.) Basic MWE features used by Schneider et al. (2014), including lemma, POS tags, word shapes and features indicating whether the token matches entries in any of several multiword lexicons (WordNet, SemCor, SAID, WikiMwe, English Wiktionary and Multiword Entries on the Phrases.net website), b.) the provided Brown clusters and c.) capitalization features, an auxiliary verb vs. main verb feature and unlexicalized WordNet supersense features. Based on the model, these features are conjoined with the MWE, supersense, or extended label set to form zero-order features. For the closed condition, we exclude the features based on multiword lexicons.

3.2 Baseline: Multinomial Logistic Regression

As a baseline, we predict the label of each word based on the features of the word within a sentence. The multinomial logistic regression models the conditional probability of the label of the ith word, denoted by Y_i, in a sentence $\boldsymbol{x}$ as:

$$p(Y_i = y \mid \boldsymbol{x}, i; \boldsymbol{\lambda}) = \frac{\exp \boldsymbol{\lambda}^\top \mathbf{h}(\boldsymbol{x}, y, i)}{\sum_{y'} \exp \boldsymbol{\lambda}^\top \mathbf{h}(\boldsymbol{x}, y', i)}, \quad (1)$$

where $\mathbf{h}$ denotes a feature vector that contains features that describe the token i, and its relationships with some of its adjacent words in $\boldsymbol{x}$ conjoined with the label y. $\boldsymbol{\lambda}$ denotes a vector of feature weights and is learned from data.

Constraints on labels are not taken into account during training. We incorporated these constraints during testing in a greedy manner: For the ith word, we considered only the labels that make it valid with respect to the bigram label constraints based on the predicted label for the $(i-1)$th word.

3.3 Conditional Random Field

In the linear chain CRF (Lafferty et al., 2001), the conditional probability of a valid label sequence $\boldsymbol{y}$ of words in a sentence $\boldsymbol{x}$ is modeled as:

$$p(\boldsymbol{y} \mid \boldsymbol{x}; \boldsymbol{\lambda}) = \frac{\exp \boldsymbol{\lambda}^\top \sum_{i=1}^{|\boldsymbol{x}|} \mathbf{h}(\boldsymbol{x}, y_i, y_{i-1}, i)}{\sum_{\boldsymbol{y}'} \exp \boldsymbol{\lambda}^\top \sum_{i=1}^{|\boldsymbol{x}|} \mathbf{h}(\boldsymbol{x}, y'_i, y'_{i-1}, i)}, \tag{2}$$

where $\boldsymbol{\lambda}$ is a vector of feature weights shared across all positions (i.e., words) and sentences. The feature vector $\mathbf{h}$ contains the zero-order features described above and the first-order features. The first-order features model the dependencies between the label of ith word and that of $(i+1)$th word. We assume a dummy label y_0 for notational convenience.

In both training and testing, we ensure that the constraints on the consecutive labels are satisfied. The label of the ith word only depends on the token sequence, its offset in the sentence and the labels of $(i-1)$th and $(i+1)$th words. Dynamic programming is used for exact inference; runtime is quadratic in the size of the label set and linear in the sequence length. We maximize ℓ_2-regularized log-likelihood using L-BFGS to learn the feature weights $\boldsymbol{\lambda}$:

$$\sum_{(\boldsymbol{x}, \boldsymbol{y} \in \mathcal{D})} \log p(\boldsymbol{y} \mid \boldsymbol{x}; \boldsymbol{\lambda}) - \alpha_1 \|\boldsymbol{\lambda}_1\|_2^2 - \alpha_2 \|\boldsymbol{\lambda}_2\|_2^2, \tag{3}$$

where $\mathcal{D}$ contains all training instances, $\boldsymbol{\lambda}_1$ ($\boldsymbol{\lambda}_2$) corresponds to the parameters for zero-order (first-order) features, and α_1 (α_2) is the regularization strength for zero-order (first-order) feature weights. In our preliminary experiments, we found that using different regularization strengths for zero-order and first-order features can benefit accuracy.

3.4 Double-Chained CRF

We propose a double-chained CRF (2-CRF) that factors the labels into separate MWE and supersense annotations. Such a model has been used for joint POS tagging and noun-phrase chunking by Sutton et al. (2007). The model is illustrated in Fig. 1; the heart of the difference lies in restricting first-order dependencies within MWE or supersense labels, not the combination of the two.

Concretely, the 2-CRF separates the zero-order features for MWE and for supersense tags. Second, while the traditional chain-structured CRF has a feature for each pair of labels in the extended label set, the 2-CRF introduces first-order features capturing each consecutive pair of MWE labels, and (separately) each consecutive pair of supersense labels. This model removes some repetitive parameters. For example, instead of having parameters to capture the relation between consecutive *B* and *I* tags paired with all supersenses, 2-CRF will have only one parameter. Moreover, if the feature weights $\boldsymbol{\lambda}$ for all the features between m_i and s_i pairs are zero, the 2-CRF model is equivalent to two separate CRFs for the two tasks. Therefore, it has a flexibility to learn the parameters for the two tasks jointly or separately. Due to this kind of flexibility, we expect that the 2-CRF model has a better generalization ability.

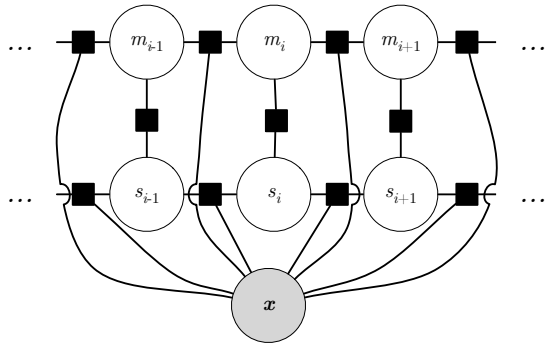

Figure 1: Double-chained CRF expressed as a factor graph: $\boldsymbol{x}$ is the whole sentence. For a token in position i, its MWE label is m_i and its supersense tag is s_i.

For a sentence $\boldsymbol{x}$ with a valid label sequence $\boldsymbol{y} = (\boldsymbol{m}, \boldsymbol{s})$, where $\boldsymbol{m}$ denotes the MWE tag sequence and $\boldsymbol{s}$ denotes supersense tag sequence, the conditional probability of $(\boldsymbol{m}, \boldsymbol{s})$ given $\boldsymbol{x}$ is defined as:

$$p(\boldsymbol{m}, \boldsymbol{s} \mid \boldsymbol{x}) = \frac{\exp \boldsymbol{\lambda}^\top \sum_{i=1}^{|\boldsymbol{x}|} \mathbf{h}(\boldsymbol{x}, m_i, s_i, m_{i-1}, s_{i-1}, i)}{\sum_{\boldsymbol{m}', \boldsymbol{s}'} \exp \boldsymbol{\lambda}^\top \sum_{i=1}^{|\boldsymbol{x}|} \mathbf{h}(\boldsymbol{x}, m'_i, s'_i, m'_{i-1}, s'_{i-1}, i)}, \tag{4}$$

where the feature vector function $\mathbf{h}$ can be written as:

$$\begin{aligned} &\mathbf{h}(\boldsymbol{x}, m_i, s_i, m_{i-1}, s_{i-1}, i) = \\ &\big\langle \mathbf{h}^m(\boldsymbol{x}, m_i, i); \mathbf{h}^s(\boldsymbol{x}, s_i, i); \\ &\mathbf{h}^{mm}(m_i, m_{i-1}); \mathbf{h}^{ss}(s_i, s_{i-1}); \mathbf{h}^{ms}(m_i, s_i) \big\rangle. \end{aligned} \tag{5}$$

$\mathbf{h}$ contains the following features: two copies of the zero-order features conjoined with the MWE tag m_i and supersense tag s_i, first-order features between m_i and m_{i-1}, s_i and s_{i-1} and m_i and s_i.

Similar to the CRF, we enforce label constraints on the MWE label sequence both at training and prediction time.

Inference can be carried out exactly using similar dynamic programming algorithms to those used for the CRF. Training is carried out as for the CRF (i.e., ℓ_2-regularized log-likelihood; see Eq. 3).[8]

4 Experiments

4.1 Experimental Setup

We compare the performance of the following four models that use exactly the same input features §3.1:

- Multinomial logistic regression (MLR) as described in §3.2 (a zero-order model)
- Structured perceptron as used by Schneider and Smith (2015) with the same set of features (first-order, similar to our CRF)
- CRF as described in §3.3
- Double-chained CRF as described in §3.4

We used the AMALGrAM[9] code base for feature extraction (Schneider and Smith, 2015). For hyperparameter tuning, we hold out 30% randomly selected training samples of the DiMSUM dataset as validation data. Using preliminary experiments on validation data, we set the number of L-BFGS iterations for multinomial logistic regression, CRF, and 2-CRF as 100, 120, and 120, respectively. We set the number of iterations of averaged perceptron algorithm for structured percetpron as 10. We also impose a percept cutoff of 3 on the minimum number of occurrences for a zero-order percept to be considered in the models. We use validation data to tune α_1 and α_2 (where applicable) hyperparameters. After tuning the parameters, we use the whole DiMSUM training dataset to train the models.

4.2 Results and Discussion

Tables 1 and 2 show the results for the closed and open conditions. The selected hyperparameters α_1 and α_2 are shown for each model. In each table, we show the results on our held-out validation data and on the DiMSUM test datasets.

The official submitted systems are marked with * in the tables.[10] For the closed condition, the 2-CRF had not completed training, so our entry was the CRF; it achieved first place.

For the open condition, training of 2-CRF had only completed 80 iterations at the submission deadline, so that is what was entered (it achieved second place). We report those scores, as well as the slightly improved scores obtained after 120 iterations.

Across the board, there is roughly a 9% decrease in the F_1 score when we move from validation to DiMSUM test datasets. This is not surprising because the validation and DiMSUM datasets represent different text genres and styles.

We measured the statistical significance of the difference between the structured perceptron (SP) and the other methods. We used a randomization test (Yeh, 2000) at the sentence level to estimate the confidence level of the difference (p-value < 0.05). We indicate in Tables 1 and 2 in italics the cases where the improvement over the structured perceptron is significant.

In the closed condition, the 2-CRF model leads to the highest F_1 scores for all evaluation metrics. Interestingly, the structured perceptron improves on MWE but suffers on supersenses, relative to the zero-order MLR model.[11] CRF and 2-CRF show improvements against MLR on both tasks, with the latter winning overall on validation and (slightly) on test data.

In the open condition, we see similar patterns except a few cases: structured perceptron has the highest F_1 score in MWE identification on validation data and MLR slightly outperforms 2-CRF in supersense tagging on test data. However, the differences are not statistically significant over 2-CRF, and it has the highest combined score.

Finally, we observe that adding the features based

[8] An open source efficient cython implementation of our method will be made publicly available at: `https://github.com/mjhosseini/2-CRF-MWE`.

[9] `http://www.cs.cmu.edu/~ark/LexSem`

[10] The official results of the shared task are based on the macroaverages of the per-domain scores, while we have done the detailed analysis based on microaverage scores of the whole dataset. Our system got combined macroaverage F_1 score of 57.10% for the closed condition and 57.71% for the open condition.

[11] The MLR model could potentially be improved with dynamic programming instead of greedy prediction.

				Validation Data			DiMSUM Data		
	# iter.	α_1	α_2	MWE	SST	Combined	MWE	SST	Combined
MLR	100	1.6	-	58.68	*66.24*	64.96	49.84	*57.14*	*56.04*
SP	10	-	-	63.39	65.15	64.83	52.37	55.85	55.29
CRF*	120	1.6	0.32	60.76	*66.66*	*65.57*	53.93	*57.47*	*56.88*
2-CRF	120	1.6	0.12	**64.13**	***67.02***	***66.46***	**54.02**	***57.89***	***57.23***

Table 1: Closed condition: Results on validation (left) and DiMSUM data (right) for multinomial logistic regression (MLR), structured perceptron (SP), CRF, and 2-CRF models. The hyperparameters and F_1 scores for identifying MWEs, supersenses, and their combination are reported. The best result in each column is bolded. The results that are significant over SP (p-value < 0.05) are italicized. The system denoted by * is our official submission for the supervised closed condition.

				Validation Data			DiMSUM Data		
	# iter	α_1	α_2	MWE	SST	Combined	MWE	SST	Combined
MLR	100	2.4	-	62.75	*66.40*	65.76	51.71	***58.04***	*57.08*
SP	10	-	-	**69.21**	65.72	66.37	56.79	55.93	56.08
CRF	120	1.2	0.12	66.78	*66.74*	66.75	56.61	*57.62*	*57.44*
2-CRF	120	1.6	0.2	67.30	***67.29***	***67.29***	56.42	*57.87*	***57.61***
2-CRF*	80	1.6	0.2	67.37	*66.78*	66.89	**57.24**	*57.64*	*57.57*

Table 2: Open condition: Results on validation (left) and DiMSUM data (right) for multinomial logistic regression (MLR), structured perceptron, CRF, and 2-CRF models. The hyperparameters and F_1 scores for identifying MWEs, supersenses, and their combination are reported. In the open condition, we have added features using the following lists of English MWEs based on: WordNet, SemCor, SAID, WikiMwe, English Wiktionary, Multiword Entries on the Phrases.net website (Schneider et al., 2014). The best result in each column is bolded. The results that are significant over SP are italicized. The system denoted by * is our official submission for the open condition.

on multiword lexicons (moving from supervised closed to open condition) improves MWE identification, without harming supersense tagging performance. The increase in the performance of MWE identification is statistically significant across all methods and test datasets.

5 Conclusions

We presented the results of four models for the joint prediction of MWE annotations and supersense annotations: multinomial logistic regression, structured perceptron, CRF and double-chained CRF. We found that double-chained CRF performs well on both tasks. We showed that, consistent with past work, adding features based on multiword lexicons improves the performance of all models.

Acknowledgments

We are grateful to Nathan Schneider for helping us use his feature extraction codebase. We also thank Lingpeng Kong and the reviewers for helpful feedback. This research was supported in part by DARPA DEFT (FA8750-12-2-0342), National Science Foundation (DBI-1355899), the American Cancer Society (127332-RSG-15-097-01-TBG), and Solid Tumor Translational Research.

References

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.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. Wit3: Web inventory of transcribed and translated talks. In *Proceedings of EAMT*.

Dirk Hovy, Anders Johannsen, and Anders Søgaard. 2015. User review sites as a resource for large-scale sociolinguistic studies. In *Proceedings of WWW*.

Anders Johannsen, Dirk Hovy, Héctor Martinez, Barbara Plank, and Anders Sgaard. 2014. More or less supervised super-sense tagging of Twitter. In *Proceedings of *SEM*.

Lingpeng Kong, Nathan Schneider, Swabha

Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A. Smith. 2014. A dependency parser for tweets. In *Proc. of EMNLP*.

John Lafferty, Andrew McCallum, and Fernando C.N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of ICML*.

Lance A. Ramshaw and Mitchell P. Marcus. 1995. Text chunking using transformation-based learning. arXiv preprint cmp-lg/9505040.

Nathan Schneider and Noah A. Smith. 2015. A corpus and model integrating multiword expressions and supersenses. In *Proc. of NAACL*.

Nathan Schneider, Emily Danchik, Chris Dyer, and Noah A. Smith. 2014. Discriminative lexical semantic segmentation with gaps: running the mwe gamut. *Transactions of the Association for Computational Linguistics*, 2:193–206.

Nathan Schneider, Dirk Hovy, Anders Johannsen, and Marine Carpuat. 2016. SemEval 2016 Task 10: Detecting Minimal Semantic Units and their Meanings (DiMSUM). In *Proc. of SemEval*, San Diego, California, USA, June.

Nathan Schneider. 2014. *Lexical Semantic Analysis in Natural Language Text*. Ph.D. thesis, Carnegie Mellon University.

Charles Sutton, Andrew McCallum, and Khashayar Rohanimanesh. 2007. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. *The Journal of Machine Learning Research*, 8:693–723.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of COLING*.