Drop-out Conditional Random Fields for Twitter with Huge Mined Gazetteer

Eunsuk Yang[‡§] **Young-Bum Kim**[†§] **Ruhi Sarikaya**[†] **Yu-Seop Kim**[‡]

[†]Microsoft Corporation, Redmond, WA
[‡]Hallym University, South Korea
esyang219@gmail.com
{ybkim, ruhi.sarikaya}@microsoft.com
yskim01@hallym.ac.kr

Abstract

In named entity recognition task especially for massive data like Twitter, having a large amount of high quality gazetteers can alleviate the problem of training data scarcity. One could collect large gazetteers from knowledge graph and phrase embeddings to obtain high coverage of gazetteers. However, large gazetteers cause a side-effect called "feature under-training", where the gazetteer features overwhelm the context features. To resolve this problem, we propose the dropout conditional random fields, which decrease the influence of gazetteer features with a high weight. Our experiments on named entity recognition with Twitter data lead to higher F1 score of 69.38%, about 4% better than the strong baseline presented in Smith and Osborne (2006).

1 Introduction

Nowadays, people are generating tremendous amount of information on social websites. For example, more than 200 million tweets are generated everyday on Twitter (Ritter et al., 2011). Twitter has become a key news source, in addition to standard news channels. As such, social scientists are starting to pay attention to it in recent years (Bollen et al., 2011; Chung and Mustafaraj, 2011; Xu et al., 2014; Calvin et al., 2015; Baldwin et al., 2015; Bellmore et al., 2015). The traditional machine learned modeling approaches trained with small and clean general text, such as news articles, perform poorly when applied to tweets, because tweets are structurally very different from general text. Thus, it is necessary to build new models for Twitter. One could label a reasonable size of tweets to train a model for a natural language processing (NLP) application. The problem is that it is very expensive to refresh the annotated data to keep the model up-to-date, because users generate tweets in a unprecedented rate (Hachman, 2011).

An obvious solution to the problem is to develop methods of utilizing a large amount of unlabeled data. One way is to induce word embeddings in a real-valued vector space from a large number of tweets (Kim et al., 2015a; Mikolov et al., 2013; Pennington et al., 2014). It is shown that the task-specific embeddings induced on tweets provide more powerful than those created from out-of-domain texts (Owoputi et al., 2012; Anastasakos et al., 2014).

Another method is to build the task-specific gazetteers. Task-specific gazetteers make the models more general and increase their coverage for unseen events. They have been proven to be useful on a number of tasks (Smith and Osborne, 2006; Li et al., 2009; Liu and Sarikaya, 2014; Kim et al., 2015b; Kim et al., 2015c). Since gazetteers can improve modeling performance, here we more focus on how to use gazetteer more effectively. To build gazetteers with sufficient coverage for our task, we first expand gazetteers from knowledge graph and phrase embeddings.

However, since the expanded gazetteers cover significant proportions of the entities in the training data, the weight of gazetteers features are easily inflated and thus the model tends to rely too much on lexical features extracted from the gazetteers fea-

[§] Both authors contributed equally.

tures to assign a tag rather than the contextual features such as n-gram, a phenomenon called "feature under-training". As a result, we often observe noticeable performance degradation at test time when the entity value does not exist in the training set or the entity dictionary.

To solve this problem, we introduce a model called dropout CRFs [1] and compare to the combination model proposed by Smith and Osborne (2006). In our experiments, we show that the proposed method significantly improves the F1 score from 65.54% to 69.38%, compared to the baseline.

2 Model

For the named entity recognition (NER) task, the input is a sentence consisting of a sequence of words, $x = (x_1 \dots x_n)$ and the output is a sequence of corresponding named entity tags $y = (y_1 \dots y_n)$. We model the conditional probability $p(y|x;\theta)$ using linear-chain CRFs (Lafferty et al., 2001):

$$p(y|x;\theta) = \frac{\exp(\theta \cdot \Phi(x,y))}{\sum_{y' \in \mathcal{Y}(x)} \exp(\theta \cdot \Phi(x,y'))}$$

where θ is a set of model parameters. $\mathcal{Y}$ contains all possible label sequences of x, and Φ maps (x,y) into a feature vector that is a linear combination of local feature vectors: $\Phi(x,y) = \sum_{j=1}^{n} \phi(x_j, y_{j-1}, y_j)$. Given fully observed training data, $\{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$, the objective of the training is to find θ that maximizes the log likelihood of the training data under the model with l_2-regularization:

$$\theta^* = \operatorname*{argmax}_{\theta} \sum_{i=1}^{N} \log p(y^{(i)}|x^{(i)};\theta) - \frac{\lambda}{2} ||\theta||^2 . \qquad (1)$$

CRFs have benefited from having a rich set of gazetteers as features in the model (Smith and Osborne, 2006; Liu and Sarikaya, 2014; Hillard et al., 2011; Kim et al., 2014; Kim et al., 2015c; Kim et al., 2015b; Kim et al., 2015d). Smith and Osborne (2006) point out that common gazetteer features fire often enough to overwhelm other features during inference. They address this problem by building a combination of two models: one without gazetteers and another with gazetteers. Instead of combining two models, we propose a simple model by having a new penalty term to the equation (1):

$$\theta^* = \operatorname*{argmax}_{\theta} \sum_{i=1}^{N} \log p(y^{(i)}|x^{(i)};\theta) - \frac{\lambda_1}{2} ||\theta||^2 - \lambda_2 \sum_{g \in G} \theta_g \,\mathrm{freq}(g), \qquad (2)$$

where G is a set of gazetteers and freq(g) counts how many times words appear in gazetteer g from training data. In our experiments, we tuned both penalty weights for local features and for gazetteer features based on a small held-out validation set. The θ_g is a member of model parameter θ and each gazetteer has its own parameter θ_g. The introduced penalty decreases common gazetteers' influence on model's decisions. By this term, we call our model *dropout CRFs*. The original dropout technique removes features randomly - for each training instance, only a random subset of the features will be activated (Hinton et al., 2012; Xu and Sarikaya, 2014). While it can be perceived as a general treatment to the under-training problem, it is not specifically directed at the problem we are facing in named entity recognition (NER) task. In NER, the under-training problem is more specific - the contextual features may not get large enough weights due to the strong influence of the gazetteer features. The negative impact of such under-training is also more measurable - if a named entity is unseen, the chance of a detection error becomes much higher. Therefore, we focus on decreasing influence of specific features. For specific features, we reduce the coverage of dropout from all features to gazetteer feature through *feature dependent regularization*. Also, the objective function of dropout CRFs, given in equation (2), is still convex because the equation (1) is convex and the new penalty term is linear with respect to θ. Therefore, a standard optimization algorithm finds optimal θ without sacrificing any abilities, which original CRFs have.

[1] The original dropout technique is to inactivate features randomly. Here, we consider to decrease the weight of a specific feature.

3 Features

In this section, we detail the feature templates used for our experiments. Besides basic features, we also employ part-of-speech (POS) tags, chunks, word representations and gazetteers. We run task-specific POS-tagger and chunker, which are trained on tweets annotated with Twitter-specific tags (Ritter et al., 2011) as well as standard Penn Treebank tags, of Owoputi et al. (2012) to produce POS tags and chunks. We explain the word representations and gazetteer features in the following subsections.

3.1 Basic Features

The model of Ritter et al. (2011) employs the features described in this subsection. They are composed of the following features: (1) n-grams: unigrams and bigrams, (2) capitalization, (3) three character suffix and prefix presence, (4) binary features that indicate presence of hyphen, punctuation mark, single-digit and double-digit, (5) gazetteers (6) topics inferred by LabeledLDA (Ramage et al., 2009), and (7) brown cluster (Brown et al., 1992) produced by Ritter et al. (2011).

To alleviate the problem of word sparsity, we also use task-specific latent continuous word representations, induced on 65 million unlabeled tweets with 1.3 billion tokens. We create three sets of word representations: CCA (Dhillon et al., 2012; Kim et al., 2015a) based on matrix factorization, word2vec (Mikolov et al., 2013) and glove (Pennington et al., 2014), which are gradient based. All word representation algorithms produce 50-dimensional word vectors for all words occurring at least 40 times in the data. We use left and right word of the target word as context for learning the word representations.

We also use compounding embeddings as an additional feature. Combining multiple sets of features has been proven to be effective (Koo et al., 2008; Kim and Snyder, 2013; Yu et al., 2013). We explore four different ways of combining the word representations: element-wise averaging, element-wise multiplication, concatenation and hierarchical clustering. We empirically determined that the element-wise averaging achieves better performance than single embeddings and other combination methods. We do not describe the results for embedding combinations in detail here.

4 Gazetteers

NER models degrades when they encounter unseen words during training. To make the problem worse, tweets contain many rare words and it is prohibitively expensive to create a training set with sufficient lexical coverage. To alleviate the problem, we extend the original gazetteers with two methods: gathering data from knowledge graph and constructing task-specific gazetteer with phrase embeddings.

4.1 Expansion from Knowledge Graph

To expand gazetteers from knowledge graph, we apply the following processing steps. We first extract the seed words from training data. With seed words, we then collect the relevant lexicons from knowledge graph such as Freebase, Wikipedia and Yelp. For example, “Dior” is related to *company* and *product* from knowledge graph. We collect all lexicons associated with seed words. In addition, we post-process gazetteers for variance: i) organization: it is composed with full name with abbreviation, such as “Indigenous Land Corporation (ILC)”. We also generate variants of full names (“Indigenous Land Corporation”) and abbreviation (“ILC”), respectively, ii) facility: because the term *elementary* indicates a school, we add a lexicon removing the word *school* of “tedder elementary school”. At the end of the processing, we end up with 2.7 millions lexicon items.

4.2 Constructing Gazetteers with Phrase Embeddings

We now describe how to construct task-specific gazetteer with phrase embeddings. We use canonical correlation analysis (CCA) (Hotelling, 1936) to induce vector representations for phrase embeddings. To extract candidate phrases from unlabeled Twitter data, we first count the frequency of the context words set for each token. The size of context words set ranges from 1 to 3. The context words set occurring more than 100 are used as a rule to extract candidate phrases.

Let n be the number of candidate phrases extracted by rules. Let $x_1 \ldots x_n$ be the original representations of the candidate phrases itself and $y_1 \ldots y_n$ be the original representations of two words to the left and right of the candidate phrases.

We use the following definition for the original representations. Let d be the number of distinct candidate phrases and d' be the number of distinct context words set.

- $x_l \in \mathbb{R}^d$ is a zero vector, in which the entry corresponding to the candidate phrases of the l-th instance is set to 1.
- $y_l \in \mathbb{R}^{d'}$ is a zero vector, in which the entries corresponding to context words set surrounding candidate phrases are set to 1.

Using CCA, we obtain phrase embeddings U with k-dimensional space. To train a classifier, we manually construct a training data with 5 positive and 5 negative samples, for each gazetteer. With this data, we learn a binary classifier with the phrase embeddings as a feature. Using this classifier, we predict whether the phrases fit to the gazetteers; we refer the readers to Neelakantan and Collins (2014) for details.

5 Experiments

To demonstrate the effectiveness of the dropout CRFs, we run experiments on named entity recognition task on the Twitter dataset of Baldwin et al. (2015). We refer the readers to Baldwin et al. (2015) for the details of the dataset. We split the data into 70% for training, 10% for tuning, and 20% for testing. For all the experiments presented in this section, both CRFs and dropout CRFs are trained using the L-BFGS (Liu and Nocedal, 1989).

5.1 Effectiveness of the Gazetteers

One of our contributions is to augment the size of gazetteers with knowledge graph and phrase embeddings. Table 1 represents the performance of a model with original gazetteers, which are collected by Ritter et al. (2011) from freebase (Base Gazet) and with gazetteers we extended (Our Gazet). The size of *Base Gazet* is 2.9 million and the size of *Our Gazet* is 6.6 million, which has an additional 3.7 million entries compared to the *Base Gazet*. The model trained *Our Gazet* improves the F1 score from 62.76% to 64.67%, compared to the baseline. As shown in Table 1, we believe that larger gazetteers can mitigate the “unseen words” problem by increasing the coverage of the gazetteers.

	F1
Base Gazet	62.76
Our Gazet	64.67

Table 1: Comparison of models with or without new gazetteers. **Base Gazet** is a model with gazetteers collected by Ritter et al. (2011) and **Our gazet** is a model with gazetteers we constructed by augmenting the *Base Gazet* with additional items, using knowledge graph and phrase embeddings.

5.2 Effectiveness of the Dropout CRFs

We conducted additional experiments with the CRF model that uses *Our Gazet*. Table 2 shows the overall results for models with and without dropout. We compare three models: the vanilla CRFs (CRFs$_{vanilla}$), the combination model as described in Smith and Osborne (2006) (CRFs$_{LOP}$) and our dropout model (CRFs$_{dropout}$). To avoid model parameters for gazetteer features getting over-regularized, Smith and Osborne (2006) propose to train separate models with and without gazetteers. They combine predictions from the two models by using logarithmic opinion pool (LOP). We refer the reader to Smith et al. (2005) for further details.

The CRFs$_{vanila}$ yields 64.03% F1 score and the CRFs$_{LOP}$ improves the performance to 65.54%. The CRFs$_{dropout}$, which reduces the influence of gazetteer features, improves the F1 score to 69.38%, which corresponds to a 13% decrease in error relative to vanilla CRFs.

	F1
CRFs$_{vanila}$	64.67
CRFs$_{LOP}$	65.54
CRFs$_{dropout}$	69.38

Table 2: Comparison of models with or without dropout. CRFs$_{vanilla}$ is the vanilla CRFs with all features. CRFs$_{LOP}$ is a combination of CRFs with all features except for gazetteers and CRFs with gazetteers only, using logarithmic opinion pool (LOP). CRFs$_{dropout}$ is the dropout CRFs with all features.

5.3 Analysis

While previous NER tasks mostly focus on reporting numbers on the original data set (Baldwin et al., 2015; Yang and Kim, 2015; Kim et al., 2015c), we further investigate how the tagging performance may change, if entities are unseen at test time. To enable such analysis, we create additional test set based on the original test set by replacing each word in `person` and `company` entities with a special token, *XXXXX*, indicating unseen words. This new test set represents an extreme case, where none of the words contained in the gazetteers are observed in the training data.

Table 3 represents the comparison of vanilla CRF model and dropout model for unseen test. Gazetteer is helpful to resolve "unseen words" problem. Unfortunately, frequent appearance of gazetteer makes a model learn weak context feature and strong gazetteer feature. By forcing a weight of gazetteer feature low, the dropout model allows the weak context features to become strong and the large weight of gazetteer feature to become smaller. Consequently, CRF$_{dropout}$ shows the significant improvement compared to CRF$_{vanilla}$.

Tags	**CRF**$_{dropout}$	**CRF**$_{vanilla}$
person	74.43	65.81
company	65.74	57.19

Table 3: Comparison of vanilla CRF model and dropout model for unseen test

To see a change of feature weight when we apply dropout technique, we show the feature weights for the word "cahill" of vanilla CRFs and dropout CRFs in Table 4. In vanilla CRFs, gazetteers have a strong weight compared to the context features. However, our dropout CRFs decrease the weight of gazetteer features, while making the context features larger, to steer the models' decision in the right direction.

6 Conclusion

In this paper, we showed how to improve the CRF based NER model for Twitter by exploiting a large number of gazetteers. Using gazetteers in modeling helps the coverage and generalization but simply incorporating gazetteers of all of large sizes into the model may lead to "under-training" of parameters corresponding to the context features. We addressed this problem by adding the dropout penalty term in the CRF training, which improves better parameter. The proposed technique results in significant improvements over the baseline.

cahill (answer: geo-loc prediction: person)

CRFs$_{vanila}$
people.person → I-person : 7.46
lastname.5000 → I-person: 9.63
lastname.5000 → I-geo-loc: 4.01
people.person.lastnames → I-person : 6.6
w[-1]|w[0]='s|Cahill → I-person : -1.24
w[-1]|w[0]='s|Cahill → I-geo-loc : 0.28
w[-1]='s → I-person : 0.97
w[-1]='s → I-geo-loc : -0.13
CRFs$_{dropout}$
people.person → I-person : 5.2
lastname.5000 → I-person: 4.67
lastname.5000 → I-geo-loc: 4.19
people.person.lastnames → I-person : 4.36
w[-1]|w[0]='s|Cahill → I-person : 1.98
w[-1]|w[0]='s|Cahill → I-geo-loc : 3.02
w[-1]='s → I-person : 1.41
w[-1]='s → I-geo-loc : 1.82

Table 4: Snapshot of feature weights for the word "cahill", given sentence *tonight 's cahill event*. The vanilla CRFs predict it to `person` and dropout CRFs predict it to `geo-loc` correctly.

One of the future directions of research is to extend the same idea to various sequence learning problems: part-of-speech tagging and slot tagging.

Acknowledgments

We thank Do Kook Choe, Puyang Xu, Alan Ritter and Karl Startos for helpful discussion and feedback.

References

Tasos Anastasakos, Young-Bum Kim, and Anoop Deoras. 2014. Task specific continuous word representations for mono and multi-lingual spoken language understanding. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 3246–3250. IEEE.

Timothy Baldwin, Young-Bum Kim, Marie Catherine de Marneffe, Alan Ritter, Bo Han, and Wei Xu. 2015. Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. *ACL-IJCNLP 2015*, page 126.

Amy Bellmore, Angela J Calvin, Jun-Ming Xu, and Xiaojin Zhu. 2015. The five ws of bullying on twitter: Who, what, why, where, and when. *Computers in Human Behavior*, 44:305–314.

Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8.

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.

Angela J Calvin, Amy Bellmore, Jun-Ming Xu, and Xiaojin Zhu. 2015. # bully: Uses of hashtags in posts about bullying on twitter. *Journal of School Violence*, 14(1):133–153.

Jessica Elan Chung and Eni Mustafaraj. 2011. Can collective sentiment expressed on twitter predict political elections? In *AAAI*.

Paramveer Dhillon, Jordan Rodu, Dean Foster, and Lyle Ungar. 2012. Two step cca: A new spectral method for estimating vector models of words. *arXiv preprint arXiv:1206.6403*.

Mark Hachman. 2011. Humanitys tweets: Just 20 terabytes. *PCMAG. COM*.

Dustin Hillard, Asli Celikyilmaz, Dilek Z Hakkani-Tür, and Gökhan Tür. 2011. Learning weighted entity lists from web click logs for spoken language understanding. In *INTERSPEECH*, pages 705–708.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Harold Hotelling. 1936. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377.

Young-Bum Kim and Benjamin Snyder. 2013. Unsupervised consonant-vowel prediction over hundreds of languages. In *ACL (1)*, pages 1527–1536.

Young-Bum Kim, Heemoon Chae, Benjamin Snyder, and Yu-Seop Kim. 2014. Training a korean srl system with rich morphological features. In *ACL (2)*, pages 637–642.

Young-Bum Kim, Benjamin Snyder, and Ruhi Sarikaya. 2015a. Part-of-speech taggers for low-resource languages using cca features. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1292–1302.

Young-Bum Kim, Karl Stratos, Xiaohu Liu, and Ruhi Sarikaya. 2015b. Compact lexicon selection with spectral methods. In *Proceedings of Association for Computational Linguistics (ACL)*, pages 806–811.

Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. 2015c. Pre-training of hidden-unit crfs. *ACL. Association for Computational Linguistics*.

Young-Bum Kim, Karl Stratos, Ruhi Sarikaya, and Minwoo Jeong. 2015d. New transfer learning techniques for disparate label sets. *ACL. Association for Computational Linguistics*.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Xiao Li, Ye-Yi Wang, and Alex Acero. 2009. Extracting structured information from user queries with semi-supervised conditional random fields. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*.

D.C. Liu and J. Nocedal. 1989. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528.

Xiaohu Liu and Ruhi Sarikaya. 2014. A discriminative model based entity dictionary weighting approach for spoken language understanding.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Arvind Neelakantan and Michael Collins. 2014. Learning dictionaries for named entity recognition using minimal supervision. *EACL 2014*, page 452.

Olutobi Owoputi, Brendan OConnor, Chris Dyer, Kevin Gimpel, and Nathan Schneider. 2012. Part-of-speech tagging for twitter: Word clusters and other advances. *School of Computer Science, Carnegie Mellon University, Tech. Rep*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*, 12.

Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D Manning. 2009. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 248–256. Association for Computational Linguistics.

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

Andrew Smith and Miles Osborne. 2006. Using gazetteers in discriminative information extraction. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, pages 133–140. Association for Computational Linguistics.

Andrew Smith, Trevor Cohn, and Miles Osborne. 2005. Logarithmic opinion pools for conditional random fields. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 18–25. Association for Computational Linguistics.

Puyang Xu and Ruhi Sarikaya. 2014. Targeted feature dropout for robust slot filling in natural language understanding. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Jun-Ming Xu, Hsun-Chih Huang, Amy Bellmore, and Xiaojin Zhu. 2014. School bullying in twitter and weibo: a comparative study. *Reporter*, 7(16):10–14.

Eun-Suk Yang and Yu-Seop Kim. 2015. Hallym: Named entity recognition on twitter with induced word representation. *ACL-IJCNLP 2015*, page 72.

Mo Yu, Tiejun Zhao, Daxiang Dong, Hao Tian, and Dianhai Yu. 2013. Compound embedding features for semi-supervised learning. In *HLT-NAACL*, pages 563–568.