Towards Robust Chemical Recognition with TaggerOne at the BioCreative V.5 CEMP Task

Robert Leaman, Zhiyong Lu*

National Center for Biotechnology Information, Bethesda, Maryland, USA

robert.leaman@nih.gov; *zhiyong.lu@nih.gov

Abstract. We describe our submissions to the BioCreative V.5 CEMP task for chemical named entity recognition in patents. We experimented with improving the robustness of the predictions made by TaggerOne – a biomedical named entity recognition system intended to be generic to any entity type – through three methods. First, we improve the feature representation for out-of-vocabulary words with Brown clusters. Second, we improved the generalization of the model under cross-domain shifts with adversarial training. Third, we apply an ensemble approach. We find all three approaches to improve performance. Our highest performance was 0.8847 F-score. TaggerOne is publicly available at https://www.ncbi.nlm.nih.gov/bionlp/tools/taggerone/

Keywords. Chemical named entity recognition; adversarial training; ensemble methods

1 Introduction

Chemical patents are an attractive target for text mining due to their importance as a primary source for medicinal chemistry. However patents are less formal documents than published articles, and therefore more likely to contain noise – mistakes or even intentional obfuscations – in addition to jargon specific to biomedical chemistry. The recent series of shared tasks in chemical text mining at the BioCreative workshops have focused on chemical named entity recognition (NER) in both PubMed abstracts [1, 2] and chemical patents [3].

NCBI developed a pair of machine learning based systems for the CHEMDNER chemical named entity recognition task in PubMed abstracts. These systems, tmChem model 1 and model 2 [4], are both based on conditional random fields [5] and use a rich feature approach [6, 7]. NCBI created an ensemble system for the subsequent CEMP task for

chemical NER recognition in patents, taking advantage of the numerous open source chemical NER systems created for the previous CHEMDNER task [8]. The resulting ensemble had very high performance but was of limited practical use due to the significant computational overhead of obtaining predictions from multiple models and the difficulty of simultaneously deploying the various systems.

Previous work by Sutton, Sindelar and McCallum [9] shows that the performance improvements achieved when combining classifiers are due, at least in part, to a reduction in weight undertraining. When training a single model, the presence of one or more strong features during training can "drown out" the contribution of weaker features, causing their weights to be too low when the strong feature is not present at test time. Ensemble methods address this by emphasizing different subsets of the feature space, thus reducing the availability of the highly predictive features and making the average of the model predictions more generalizable. Neural networks address this issue with dropout: a percentage of inputs to each layer are randomly dropped during training [10]. In structured machine learning methods, however, Søgaard [11] suggests training with an antagonistic adversary: rather than removing features at random, remove a randomly selected subset of those that are highly predictive. Søgaard shows that training with an antagonistic adversary is particularly effective for cross-domain shifts, where the distribution of the test data does not match that of the training distribution.

In NER a primary source of error is vocabulary that was not observed during training. Our experiments therefore attempted to address this source of error in two primary ways. First, we improve the feature representation for out-of-vocabulary words by learning word representations from a large amount of unlabeled data. Second, since out-of-vocabulary effects are a form of cross-domain shift, we experiment with training using an antagonistic adversary. We also create an ensemble system as a benchmark for the upper limit of the performance that can be expected. We perform our experiments using TaggerOne, a recently released system for joint named entity recognition and normalization for various biomedical entities [12]. The highly flexible online training algorithm used by TaggerOne makes it ideal for experimentation.

2 Methods

TaggerOne is a machine learning based system for joint named entity

recognition and normalization [12]. Joint training and inference allows the model to use the normalization information to inform the NER component, resulting in increased performance for both subtasks. The model consists of a semi-Markov [13] structured linear classifier [14] using a rich feature approach for NER [6, 7], a supervised semantic indexing approach for normalization [15, 16]. The model is trained with the online training algorithm MIRA (the margin-infused relaxed algorithm) [17], and requires the specification of two hyperparamters: the regularization, which controls the size of the updates, and the maximum step size, which sets an upper bound on the update size. As a semi-Markov model, it performs segmentation and classification simultaneously, allowing one state per entity type instead of two states (as in the BIO scheme) or four states (as in the BIOEW scheme). Our adaptation of TaggerOne in this manuscript does not make use of its normalization capability.

The original TaggerOne feature set includes a wide variety of features. At the token level, these features include the token text, stem, part of speech, character n-grams, and patterns. Features at the segment level include surrounding characters, tokens and whether the segment contains unbalanced parenthesis. Some models in this work include a dictionary feature containing the chemicals lexicon provided by the Comparative Toxicogenomics Database (CTD, http://ctdbase.org/), which is derived from the chemical branch of MeSH (https://www.nlm.nih.gov/mesh). We slightly augmented this list to include the names of all of the chemical elements.

We improved the feature representation for out of vocabulary words by leveraging the availability of a large unlabeled text dataset from a similar domain, namely, PubMed. Previous work has shown Brown clusters [18] to be useful in NER [19]. In addition, more recent work has improved performance by learning a word representation from a large amount of unlabeled data [20]. Our experiments employed the Brown clusters and the clustered word representation vectors distributed by the banner-chemdner tool [21]. Our preliminary experiments showed an improvement with Brown clusters for lengths 4, 6, 10 and 20 (data not shown), which we adopted for the final experiment. Our preliminary ex-

periments did not show an improvement for word vector clusters, however (data not shown), and word vector clusters were therefore not considered further.

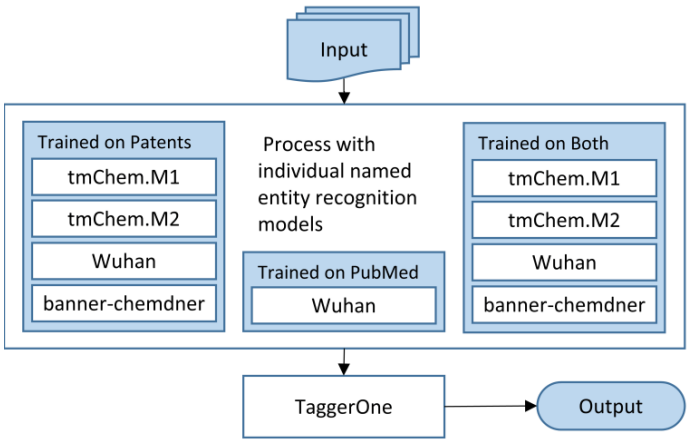

Figure 1. Description of the individual models used to train the ensemble and the flow of data through the system. Diagram adapted from [8].

Our implementation of antagonistic adversaries [11] selects a subset of features at random for each training instance – the percentage of features to select being an additional hyperparameter – then drops those features whose weight is greater than one standard deviation of the mean weight for all features. We consider the weight of the feature to be the Euclidean length of its weights across all states. Features that are dropped have their values set to zero for all feature vectors in an instance and across all states. For efficiency, features are selected by first sampling from a binomial distribution to determine the number of features that should be dropped, then the features themselves are selected randomly.

We included one ensemble run, using a configuration similar to our ensemble for the previous CEMP task but using TaggerOne to combine the individual predictions. The ensemble used four systems: tmChem model 1 and tmChem model 2 [4], the Wuhan University CHEMDNER tagger [22] and banner-chemdner [21]. All systems are open source, based on conditional random fields [5] and a rich feature approach. The systems in the ensemble were trained using combinations of patent and PubMed training data, as described in Figure 1. The output of each individual system was then input into TaggerOne as a binary feature, which

were the only features used, and TaggerOne was trained using the remaining training data. We also included the output of tmChem model 1 as one submission.

The initial implementation of TaggerOne instantiated features for all instances prior to training, making the memory requirement scale roughly linearly with the number of instances. This becomes unacceptable for very large datasets, and is unnecessary since TaggerOne uses online training. Merely instantiating features prior to training, however, would cause an unacceptable increase in training time. Instead we used separate concurrent processes to perform feature extraction and training. This allows TaggerOne to scale to arbitrarily large datasets without increasing the training time or the memory requirement.

3 Results

Our five submitted runs consisted of three with TaggerOne alone, one with TaggerOne as an ensemble, and one with tmChem model 1. We separated two thousand patents from the initial training set as a holdout set, and designated the remaining documents as the training set. The three runs with TaggerOne alone also included the PubMed abstracts from the CHEMDNER task as training data [2]. The TaggerOne ensemble was trained as described in Figure 1. tmChem was trained by combining all available patent data with the PubMed abstracts from the CHEMDNER task, as for the runs with TaggerOne alone, but also included the chemical annotations from the BC5CDR corpus [23]. The value of all TaggerOne hyperparameters, when used, was set by cross-validation on the holdout set. The four configurations of TaggerOne are described in Table 1.

Table 1. Configuration of the four variations of TaggerOne submitted. The Enhanced feature set consists of the Initial feature set plus the dictionary feature from the CTD chemical vocabulary and Brown clusters.

Run	Regularization	Maximum step size	Adversary	Feature set
TaggerOne-Raw	n/a	n/a	n/a	Initial
TaggerOne-Brown	10.0	0.001	0.00	Enhanced
TaggerOne-Adversary	10.0	0.001	0.03	Enhanced
TaggerOne-Ensemble	0.1	0.001	0.10	Ensemble

The performance of the five models submitted to the task on our internal holdout set are described in Table 2.

Table 2. Results for the five submitted runs on our internal holdout set. The highest value is shown in bold.

Run	Precision	Recall	F-score
TaggerOne-Raw	0.8405	0.8630	0.8516
TaggerOne-Brown	0.8383	0.8739	0.8558
TaggerOne-Adversary	0.8424	0.8746	0.8582
TaggerOne-Ensemble	0.8532	**0.9150**	**0.8830**
tmChem model 1	**0.8799**	0.8623	0.8710

The official performance of the five models submitted to the task are described in Table 3.

Table 3. Official results for the five submitted runs. The highest value is shown in bold.

Run	Precision	Recall	F-score
TaggerOne-Raw	0.8639	0.8733	0.8686
TaggerOne-Brown	0.8641	0.8807	0.8723
TaggerOne-Adversary	0.8635	0.8795	0.8715
TaggerOne-Ensemble	0.8439	**0.9297**	**0.8847**
tmChem model 1	**0.8731**	0.8765	0.8748

4 Discussion

We first note that the official results are generally higher than the results on our internal holdout set. We note that while TaggerOne is intended to work well for any biomedical entity type, its performance is nearly as strong as tmChem, which is specifically dedicated to chemical NER. We see that using TaggerOne "out of the box" – without setting or optimizing hyperparameters – results in performance that approaches the optimal configuration. Alternately, adding Brown clusters improved performance for both the holdout and test sets. Adversarial training helped in the holdout set, but slightly hurt in the test set, possibly due to the difference in the holdout and test sets causing the adversarial training hyperparameter to be set to a suboptimal value. The ensemble provided the highest performance. We found adversarial training to help significantly with the ensemble configuration in our preliminary experiments (data not

shown); achieving this performance required the adversarial training hyperparameter to be set to a relatively high value. The strong performance by tmChem is primarily due to high precision.

5 Conclusion

We have explored several methods of improving the robustness of predictions for chemical named entity recognition in patents. We have shown that improving the feature representation for out-of-vocabulary words (via Brown clusters) improves performance. Adversarial training improved performance on the holdout set and may be worth exploring further. The highest performances were obtained by the dedicated tool for chemical NER, tmChem, and the ensemble approach with TaggerOne.

6 Acknowledgment

The authors thank the organizers of the BioCreative V.5 CHEMDNER task and the BeCalm team. We also thank the authors of the individual open source systems used in this work. This research is funded by the National Institutes of Health Intramural Research Program, National Library of Medicine.

REFERENCES

1. Krallinger, M., et al.: CHEMDNER: The drugs and chemical names extraction challenge. Journal of cheminformatics 7, S1 (2015)
2. Krallinger, M., et al.: The CHEMDNER corpus of chemicals and drugs and its annotation principles. Journal of cheminformatics 7, S2 (2015)
3. Krallinger, M., et al.: Overview of the CHEMDNER patents task. Fifth BioCreative Challenge Evaluation Workshop, pp. 63-75, Seville, Spain (2015)
4. Leaman, R., Wei, C.H., Lu, Z.: tmChem: a high performance approach for chemical named entity recognition and normalization. Journal of cheminformatics 7, S3 (2015)
5. Lafferty, J.D., McCallum, A., Pereira, F.: Conditional random fields: Probabilistic Models for Segmenting and Labeling Sequence Data. Proceedings of the International Conference on Machine Learning, pp. 282-289 (2001)
6. Settles, B.: Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets. International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA), pp. 104-107, Geneva, Switzerland (2004)

7. Leaman, R., Gonzalez, G.: BANNER: an executable survey of advances in biomedical named entity recognition. Pac. Symp. Biocomput. 652-663 (2008)
8. Leaman, R., Wei, C.H., Zou, C., Lu, Z.: Mining chemical patents with an ensemble of open systems. Database (Oxford) 2016, (2016)
9. Sutton, C., Sindelar, M., McCallum, A.: Reducing Weight Undertraining in Structured Discriminative Learning. Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL, pp. 89-95 (2006)
10. Hinton, G., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Improving neural networks by preventing co-adaptation of feature detectors. CoRR, pp. abs/1207.0580 (2012)
11. Søgaard, A.: Part-of-speech tagging with antagonistic adversaries. Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pp. 640–644, Sofia, Bulgaria (2013)
12. Leaman, R., Lu, Z.: TaggerOne: Joint Named Entity Recognition and Normalization with Semi-Markov Models. Bioinformatics 32, 2839-2846 (2016)
13. Cohen, W.W., Sarawagi, S.: Exploiting Dictionaries in Named Entity Extraction: Combining Semi-Markov Extractions Processes and Data Integration Methods. 10th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, pp. 89-98. ACM, Seattle, Washington, USA (2004)
14. Altun, Y., Hofmann, T., Tsochantarisis, I.: Support Vector Machine Learning for Independent and Structured Output Spaces. In: Bakir, G., Hofmann, T., Scholkopf, B., Smola, A.J., Taskar, B., Vishwanathan, S.V.N. (eds.) Predicting Structured Data. The MIT Press (2007)
15. Bai, B., et al.: Learning to rank with (a lot of) word features. Inform. Retrieval 13, 291-314 (2010)
16. Leaman, R., Doğan, R.I., Lu, Z.: DNorm: Disease name normalization with pairwise learning-to-rank. Bioinformatics 29, 2909-2917 (2013)
17. Crammer, K., Singer, Y.: Ultraconservative Online Algorithms for Multiclass Problems. J Mach Learn Res 3, 951-991 (2003)
18. Brown, P., deSouza, P., Mercer, R., Pietra, V., Lai, J.: Class-Based n-gram Models of Natural Language. Computational Linguistics 18, 467-479 (1992)
19. Turian, J., Ratinov, L., Bengio, Y.: Word representations: A simple and general method for semi-supervised learning. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pp. 384–394, Uppsala, Sweden (2010)
20. Mikolov, T., Yih, W.-t., Zweig, G.: Linguistic Regularities in Continuous Space Word Representations. Proceedings of the 2013 Conference of the NAACL-HLT, pp. 746-751, Atlanta, Georgia, USA (2013)
21. Munkhdalai, T., Li, M., Batsuren, K., Ryu, H.: BANNER-CHEMDNER: Incorporating Domain Knowledge in Chemical and Drug Named Entity Recognition. Fourth BioCreative Challenge Evaluation Workshop, vol. 2, pp. 135-139 (2013)
22. Lu, Y., Ji, D., Yao, X., Wei, X., Liang, X.: CHEMDNER system with mixed conditional random fields and multi-scale word clustering. Journal of cheminformatics 7, S4 (2015)
23. Li, J., et al.: BioCreative V CDR task corpus: a resource for chemical disease relation extraction. Database (Oxford) 2016, (2016)