Word Embedding Clustering for Disease Named Entity Recognition

Víctor Suárez-Paniagua, Isabel Segura-Bedmar, and Paloma Martínez

Computer Science Department
University Carlos III of Madrid, Spain
{vspaniag,isegura,pmf}@inf.uc3m.es

Abstract. This paper reports the use of a machine learning-based approach with word embedding features for the Disease Named Entity Recognition and Normalization subtask of the BioCreative V Chemical-Disease Relation (CDR) challenge task. Firstly, we developed a feature extraction phase with standard features used in current Named Entity Recognition (NER) systems. Then, we compared the use of the word vectors and the word clusters generated by the Word2Vec tool to add the best of both in the feature set. For this purpose, we trained the Word2Vec models over Wikipedia and MedLine as corpora. Our results suggest that the use of word clusters improves 28% in F-score in disease mention recognition and increases precision almost 49% in the normalization task over the baseline system provided by the organizers.

Key words: Disease Named Entity Recognition, Machine Learning, Conditional Random Field, Word Embedding, Clustering

1 Introduction

Named Entity Recognition (NER) is a main task of Natural Language Processing (NLP) that finds and classifies terms in texts into categories. In particular, Disease Named Entity Recognition and Normalization (DNER) is an important task which reduces the time that experts spend populating biomedical knowledge bases and annotating papers and patents. Also, it is considered a previous step for relation extraction such as automatic Chemical-Disease Relations (CDR) extraction.

The BioCreative V Chemical-Disease Relation (CDR) is a challenge task for the recognition of the diseases as well as the extraction of their relations with chemical substances in the texts [1]. Two subtasks have been organized for these two purposes. Our team only provided results for the first task.

Conditional Random Fields (CRF) is a machine learning model that is used for sequence prediction. This technique performs NER task as a classification task on each token, determining whether it is an entity or not. In general, CRF obtains one of the best results in the recognition of drugs and chemical names [3, 4]. Due to this fact, we decided to used it to detect diseases entities in texts for the DNER task.

Recently, word embeddings have been used in different NLP task such as named entity recognition or parsing [7, 9]. Our hypotesis is that the incorporation of word embedding features into a CRF can improve the DNER system. Thus, we trained word embeddings using the Word2Vec tool [5] with Wikipedia and MedLine corpora, and then they were integrated in the feature set of each word.

2 Method

In this section, we describe the dataset used in our evaluation and the experiments realized by our system.

Datasets

The CDR task provided a new corpus for the evaluation. This dataset contains pieces of text with disease annotations selected from CTD-Pfizer set [2]. It is divided into training set and development set, each one with 500 articles. However, the testing phase uses raw PubMed abstracts for the extraction of the disease mentions and their normalized MeSH ids.

Experiments

We developed a system based on CRF which uses lexical features and word embedding features provided by the Word2Vec tool. In particular, we used a python binding[1] to CRFsuite [6]. In the feature extraction phase each token is represented with the following features:

- The context window of three tokens to its right and to its left in the sentence, including the current token.
- POS tags and lemmas in the context window are also considered.
- A feature representing the type of token: word, number, symbol or punctuation.
- An orthography feature which can take the following values: upperInitial (the token begins with an uppercase letter and the rest are lowercase), allCaps (all its letters are uppercase), lowerCase (all its letters are lowercase) and mixedCaps (the token contains any mixture of upper and lowercase letters).
- A binary feature that indicates whether the current token was found in a gazetteer of diseases, provided by the DNorm tool.
- A feature representing the long word shape of the current token. This feature is defined by mapping any uppercase letter, lowercase letter, digit, and other characters to X x 0 and O respectively. For example, the long word shape of "d-glycericacidemia" is xOxxxxxxxxxxxxxxx.
- A feature representing the brief word class of the current token. In this feature, consecutive uppercase letters, lowercase letters, digits, and other characters map to X, x, 0, and O, respectively. For example, the brief word shape of "d-glycericacidemia" is xOx.

[1] http://python-crfsuite.readthedocs.org/en/latest/

For this task, a pipeline in GATE [8] was created with five main processing modules: sentence splitter, tokenizer, POS tagger and morphological analyzer. In addition, we trained the Word2Vec tool using the Wikipedia and MedLine corpora in order to obtain the word embeddings for each token. Moreover, the tool is able to provide clusters using a k-means algorithms. Our goal is to compare the performance of using word vector in contrast to word clusters in the feature set. On the one hand, the word vector for the current token was tried with different dimensions of vector (50, 100, 150 and 200). On the other hand, the word clusters were computed with different k values (100, 200, 300 and 400).

To choose the best values for our system, we perform different experiments on the development dataset. These experiments are described in the Table 3. Table 4 shows that the use of DNorm seems to achieve a light improvement of 1% in F1 on the baseline system. However, the combination of Word2Vec features with DNorm does not seem to overcome the system using Word2Vec features alone. That is, DNorm could be ignored from the system. Furthermore, the system that uses Word2Vec features trained on MedLine without the DNorm gazetteer provides better results than those trained on Wikipedia. Moreover, if the system does not use the DNorm gazetteer, clusters provide better results than vectors. In particular, the best results are achieved using Word2Vec clusters (k = 200 or k = 300) with MedLine. Apart from that, the DNorm only seems to help when DNorm is combined with Word2Vec wikipedia vectors.

Results

Three different configurations were sent as the three runs for the DNER task, thereby we chose the three best results on the development dataset. Thus, each run uses the MedLine data to train the Word2Vec, and three different k values for the clustering (Run 1 uses 200, Run 2 uses 300 and Run 3 uses 100). None of them uses the DNorm gazetteer.

	TP	**FP**	**FN**	**P**	**R**	**F1**
Baseline	1341	1799	647	42.71%	67.46%	52.30%
Run 1	708	66	1280	**91.47%**	**35.61%**	**51.27%**
Run 2	703	69	1285	91.06%	35.36%	50.94%
Run 3	703	67	1285	91.30%	35.36%	50.98%

Table 1. CDR baseline and results of normalization evaluate on the test set (500 documents).

Table 1 shows the final results of the DNER task for the different runs of our system. We can see that the best F-score is obtained by the run 1, which is also the best on the development dataset, and improves 0.5% over the others runs. Run 2 and 3 obtain very close results. This may suggests that it is not suitable to use higher k values than 100. Comparing with the baseline system provided

by the organizers, the F-score is 1% lower, but the results in precision and recall are very different. While our system provides a large improvement in precision over the baseline system (almost 49%), the difference in Recall is the opposite, since our system gets around 32% less. The main source of this decrease is the high number of False Negatives (FN), probably, this is due to the lack of an accurate normalization system. For this reason, we analyze the results obtained in the disease mention recognition phase (see Table 2). As expected, the results in this evaluation are more similar to the experiments on the development dataset, achieving 77% in F-score in all runs. Thus, our system increases 28% with respect to the baseline system, mostly, because the number of False Negatives is 5 times fewer.

	TP	**FP**	**FN**	**P**	**R**	**F1**
Baseline	2652	3835	1772	40.88%	59.95%	48.61%
Run 1	3223	718	1201	81.78%	**72.85%**	**77.06%**
Run 2	3207	747	1217	81.11%	72.49%	76.56%
Run 3	3210	711	1214	**81.87%**	72.56%	76.93%

Table 2. CDR baseline and results of disease mention recognition evaluate on the test set (500 documents).

Conclusion

This paper describes the system developed by the UC3M team for the DNER task of the BioCreative V CDR challenge task. The system uses a CRF-based algorithm combining the standard features for the NER task as well as word embedding features obtained using the Word2Vec tool. We study the performance of the different features and we conclude that DNorm feature could be discarted, using MedLine instead of Wikipedia enhance the results and the k-means mode is better than the whole vector without DNorm. The final results shows that word embedding clustering features achieve an improvement in precision for the normalization task, specifically an increase of almost 49% over the baseline system provided by the organizers. However, it is slightly less in F-score because our approach only uses a list of diseases and their MeSH ids provided by the DNorm tool. On the contrary, in the recognition task, our system overcomes 28% due to the low number of False Positives.

In future work, we need to perform additional experiments to fine-tuned the dimensions of vector, the k values and the CRF's parameters through cross-validation on the training set to choose the best configuration. Given that word embedding features increase the precision, we should try another tool such as the Global Vectors for Word Representation (GloVe)[2] to compare the performance.

[2] http://nlp.stanford.edu/projects/glove/

In addition, we believe that the high number of False Negative is due to that we only work in the recognition task and not in the normalization task. Moreover, we believe that the inclusion of additional semantic features from biomedical resources (such as DrugBank, CheBI, ChemIDPlus, the ATC classification system, etc) are essential in order to improve performance for the normalization subtask. Thus, the next stage will be to add some of these resources to resolve this lack.

Acknowledgments. This work was supported by TrendMiner project [FP7-ICT287863] and by eGovernAbility-Access project (TIN2014-52665-C2-2-R).

References

1. Wei CH, Peng Y, Leaman R, et al. (2015) Overview of the BioCreative V Chemical Disease Relation (CDR) Task, in Proceedings of the fifth BioCreative challenge evaluation workshop, Sevilla, Spain.
2. Davis, Allan Peter, Wiegers, Thomas C., Roberts, Phoebe M., King, Benjamin L., Lay, Jean M., Lennon-Hopkins, Kelley, Sciaky, Daniela, Johnson, Robin, Keating, Heather, Greene, Nigel, Hernandez, Robert, McConnell, Kevin J., Enayetallah, Ahmed E. & Mattingly, Carolyn J. (2013). A CTDPfizer collaboration: manual curation of 88 000 scientific articles text mined for drugdisease and drugphenotype interactions. Database, 2013.
3. Martin Krallinger, Florian Leitner, Obdulia Rabal, Miguel Vazquez, Julen Oyarzabal, and Alfonso Valencia. 2015. Chemdner: The drugs and chemical names extraction challenge. J Cheminform, 7(Suppl 1):S1.
4. Isabel Segura Bedmar, Paloma Martínez, and María Herrero Zazo. 2013. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts. In SemEval-2013: Semantic Evaluation Exercises Workshop, pages 341350. Association for Computational Linguistics.
5. Mikolov, Tomas, Chen, Kai, Corrado, Greg & Dean, Jeffrey (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
6. Naoaki Okazaki, CRFsuite: a fast implementation of Conditional Random Fields (CRFs), http://www.chokkan.org/software/crfsuite/, 2007.
7. Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In Proceedings of the 48th annual meeting of the association for compu- tational linguistics, pages 384394. Association for Computational Linguistics.
8. H. Cunningham, V. Tablan, A. Roberts, K. Bontcheva (2013) Getting More Out of Biomedical Documents with GATE's Full Lifecycle Open Source Text Analytics. PLoS Comput Biol 9(2): e1002854. doi:10.1371/journal.pcbi.1002854 - http://tinyurl.com/gate-life-sci/
9. Richard Socher, John Bauer, Christopher D Manning, and Andrew Y Ng. 2013. Parsing with compositional vector grammars. In In Proceedings of the ACL conference. Citeseer.

System	Feature set
CRF	standard feature set
CRFD	baseline + DNorm feature
CDR DevelopmentSet-W2VMD-dD	CRF's features + word vectors of dimension d from Word2Vec trained on MedLine
CDR DevelopmentSet-W2VMD-kK	CRF's features + word cluster from Word2Vec trained with k centroids on MedLine
CDR DevelopmentSet-W2VWIKI-dD	CRF's features + word vectors of dimension d from Word2Vec trained on Wikipedia
CDR DevelopmentSet-W2VWIKI-kK	CRF's features + word cluster from Word2Vec trained with k centroids on Wikipedia
CDR DevelopmentSet-DNorm-W2VMD-dD	CRFD's features + word vectors of dimension d from Word2Vec trained on MedLine
CDR DevelopmentSet-DNorm-W2VMD-kK	CRFD's features + word cluster from Word2Vec trained with k centroids on MedLine
CDR DevelopmentSet-DNorm-W2VWIKI-dD	CRFD's features + word vectors of dimension d from Word2Vec trained on Wikipedia
CDR DevelopmentSet-DNorm-W2VWIKI-kK	CRFD's features + word cluster from Word2Vec trained with k centroids on Wikipedia

Table 3. List of experiments.

	TP	**FP**	**FN**	**P**	**R**	**F1**
CRF	4230	1118	2206	79%	66%	72%
CRFD	4380	1136	2056	79%	68%	73%
CDR_DevelopmentSet-W2VMD-50D	4255	1103	2181	79%	66%	72%
CDR_DevelopmentSet-W2VMD-100D	4246	1114	2190	79%	66%	72%
CDR_DevelopmentSet-W2VMD-150D	4272	1147	2164	79%	66%	72%
CDR_DevelopmentSet-W2VMD-200D	4268	1075	2168	80%	66%	72%
CDR_DevelopmentSet-W2VMD-100K	4410	1146	2026	79%	69%	74%
CDR_DevelopmentSet-W2VMD-200K	4423	1103	2013	80%	69%	74%
CDR_DevelopmentSet-W2VMD-300K	4419	1105	2017	80%	69%	74%
CDR_DevelopmentSet-W2VMD-400K	4361	1113	2075	80%	68%	73%
CDR_DevelopmentSet-W2VWIKI-50D	4285	1133	2151	79%	67%	72%
CDR_DevelopmentSet-W2VWIKI-100D	4263	1099	2173	80%	66%	72%
CDR_DevelopmentSet-W2VWIKI-150D	4263	1140	2173	79%	66%	72%
CDR_DevelopmentSet-W2VWIKI-200D	4272	1082	2164	80%	66%	72%
CDR_DevelopmentSet-W2VWIKI-100K	4357	1211	2079	78%	68%	73%
CDR_DevelopmentSet-W2VWIKI-200K	4364	1170	2072	79%	68%	73%
CDR_DevelopmentSet-W2VWIKI-300K	4344	1172	2092	79%	67%	73%
CDR_DevelopmentSet-W2VWIKI-400K	4356	1117	2080	80%	68%	73%
CDR_DevelopmentSet-DNorm-W2VMD-50D	4396	1123	2040	80%	68%	74%
CDR_DevelopmentSet-DNorm-W2VMD-100D	4367	1113	2069	80%	68%	73%
CDR_DevelopmentSet-DNorm-W2VMD-150D	4410	1113	2026	80%	69%	74%
CDR_DevelopmentSet-DNorm-W2VMD-200D	4423	1145	2013	79%	69%	74%
CDR_DevelopmentSet-DNorm-W2VMD-100K	3680	900	2756	80%	57%	67%
CDR_DevelopmentSet-DNorm-W2VMD-200K	3767	909	2669	81%	59%	68%
CDR_DevelopmentSet-DNorm-W2VMD-300K	3687	934	2749	80%	57%	67%
CDR_DevelopmentSet-DNorm-W2VMD-400K	3731	897	2705	81%	58%	67%
CDR_DevelopmentSet-DNorm-W2VWIKI-50D	4430	1126	2006	80%	69%	74%
CDR_DevelopmentSet-DNorm-W2VWIKI-100D	4424	1129	2012	80%	69%	74%
CDR_DevelopmentSet-DNorm-W2VWIKI-150D	4404	1185	2032	79%	68%	73%
CDR_DevelopmentSet-DNorm-W2VWIKI-200D	4412	1149	2024	79%	69%	74%
CDR_DevelopmentSet-DNorm-W2VWIKI-100K	3397	892	3039	79%	53%	63%
CDR_DevelopmentSet-DNorm-W2VWIKI-200K	3489	913	2947	79%	54%	64%
CDR_DevelopmentSet-DNorm-W2VWIKI-300K	3501	880	2935	80%	54%	65%
CDR_DevelopmentSet-DNorm-W2VWIKI-400K	3569	937	2867	79%	55%	65%

Table 4. Experimental results on CDR_Dev dataset (Runs in bold).