Hybrid approaches for the DNER task at BioCreative V: the INRA/LIMSI system

Louise Deleger, Cyril Grouin, and Robert Bossy

INRA, MaIAGE, Jouy-en-Josas, France
{louise.deleger,robert.bossy}@jouy.inra.fr
LIMSI-CNRS, Orsay, France
cyril.grouin@limsi.fr

Abstract. *This paper describes the INRA/LIMSI system designed for the DNER task at BioCreative. The system relies on hybrid approaches (machine-learning and linguistic rules) to recognize and normalize disease names in scientific abstracts. Our best configuration achieved a precision of 0.8305, a recall of 0.8355 and a F-measure of 0.8330. Nevertheless, this system is quite slow and average response time (per document) was of 11,240 msec using this configuration. We also present the experiments we made to maximize recall and to reduce the response time of our web service.*

Key words: Disease name recognition, disease name normalization, CRF, rule-based methods

1 Introduction

In this paper, we present the system we designed to participate in the DNER task [11, 5] of the 2015 BioCreative challenge. Our method relies on two steps: first, a machine-learning approach based on a CRF framework to identify disease names, and second, a rule-based approach to normalize those names to MeSH concepts. As per the challenge requirements, our system was provided as a web-service. We designed several configurations in order to maximize disease name recognition performance (distinct CRF models to maximize either precision or recall) and response time (full and reduced versions of the normalization method).

2 Systems description and methods

2.1 Identification of disease names

In order to identify disease names, we designed a process based on the WAPITI [4] implementation of the CRF framework [3]. The following features were used:

- Lexical features: the token itself;
- Typographical features: token length in characters, typographic case of the token, presence of punctuation marks and digits in the token;

- Morpho-syntactic features: part-of-speech tag, lemma and stem of the token; morpho-syntactic information was provided by the TreeTagger POS tagger [9]. While we obtained better results using GENIA Tagger, this tool is rather slow to load the models needed to process the texts given as input. Since a rapid response time was also a requirement of the challenge, we decided to use the Tree Tagger system due to its shorter process time;
- Semantic features: presence in the token of specific prefixes and suffixes we manually identified as relevant cues for disease names,[1] information from the UMLS [7] for each token (CUI, semantic type and semantic group), information from the CTD lexicon, and identification of trigger words[2] we found useful in the context of disease names;
- Distributional analysis: the cluster ID of each token. We performed an automatic unsupervised clustering of all tokens from the training corpus into 1,000 clusters, based on the context in which each token occurs, using Liang's implementation [6] of the Brown algorithm [1];

For some features (token, typographic case), we also defined bigrams of features and contextual features (i.e., features for previous and next token). We did not perform any cross-validation to design our model. However, automatic feature selection was performed through the $l1$ regularization.

Design of experiments In order to tune our models, we split the corpus into two sub-corpora: a training corpus composed of 300 files and a test corpus composed of the remaining 198 files.

We designed two CRF models, based on the above-mentionned features:

- a model to maximize precision, created on the whole training corpus;
- a model to maximize recall: we reduced the size of the training corpus by keeping tagged tokens and two tokens surrounding each annotated token.[3]

In the first model, all tokens from our training corpus are used to build the model. Using all tokens gives a maximum of contexts to the CRF system, including tokens that must not be annotated. This constitutes the model to maximize precision. In the second model, we keep all annotated tokens but we reduce the number of unannotated tokens. As a consequence, the proportions of annotated entities is higher than it would be in the real corpus (and in the test corpus). Using a limited amount of tokens allows us to bias the CRF towards recall. This constitutes our model to maximize recall.

Table 1 shows the number and percentage of annotated and unannotated tokens in each category, when using the full training corpus and when using the

[1] Following prefixes and suffixes were used: *dys-, thrombo-, -ias, -iasis, -icity, -noma, -penia, -phrenia, -tension, -thenia, -titis, -uria.*

[2] Trigger words used were: *disease(s), failure(s), hemorrhage, infarction(s), nausea, pain(s), syndrome(s), toxicity.*

[3] We obtained our best results with this model using 2 tokens surrounding annotated tokens. We also tested contexts from 5 to 100 tokens surrounding annotated tokens by rounds of 5 (i.e., 5 tokens, 10 tokens, 15 tokens, etc.).

reduced training corpus. Even though the task only addressed the recognition of disease names, our models are trained to identify both chemical and disease names. We noticed our models perform better when processing several categories instead of only one.

Category	Full corpus	Reduced corpus
Disease	4,254 (6.01%)	4,254 (16.81%)
Chemical	4,104 (5.80%)	4,104 (16.22%)
No category	62,429 (88.19%)	16,948 (66.97%)
Total	70,787 (100.00%)	25,306 (100.00%)

Table 1. Number and percentage of annotated and unannotated tokens in the training corpus (full vs. reduced corpus)

Table 2 shows results obtained on our test sub-corpus using the two CRF models we designed.

Category	Model Precision			Model Recall		
	P	R	F	P	R	F
Chemical	**0.903**	0.764	**0.828**	0.669	**0.895**	0.765
Disease	**0.872**	0.745	**0.804**	0.610	**0.786**	0.687
Overall	**0.889**	0.756	**0.817**	0.643	**0.846**	0.731

Table 2. Results achieved on the test sub-corpus depending on the CRF model used (to maximize precision vs. to maximize recall), in terms of precision (P), recall (R) and F-measure (F). Top results are in bold

2.2 Disease name normalization

Once disease entities are recognized by our CRF model, we normalize them to MeSH concepts using a rule-based approch based on syntactic and lexical characteristics. We use and extend the ToMap method [2], which relies on the internal structure of entities and map them based on their syntactic heads, the underlying assumption being that the head is the most informative component of an entity.

Our approach can be described as follows. In a preliminary step, we look for exact matches between an extracted entity and an entity from the thesaurus. We also added annotations from the training and development corpora to the lexicon. In the event of multiple matches (i.e. when several concepts are denoted by the same names in the ontology), we choose the concept whose entry (preferred) term matches the extracted entity (rather than a match with a synonym).

If no exact match is found, then the entity syntactic head is matched against the heads of MeSH terms. If there is a match, the entity is assigned the concept(s) with matching head(s). If not, the default tag "-1" is used.

Some heads may be too generic to be useful to select the right concept. In this case, entities are categorized using the modifier(s) instead. For instance, the word disorder is not informative and the entity mental disorder will be normalized using the modifier mental. These non-discriminant heads are dependent on the knowledge source used for the normalization task and have to be provided to the system. In our experiments, the list was manually designed based on an analysis of the syntactic heads of entities from the training corpus.

Different concepts may share a same syntactic head, which means that there can be multiple candidates for a given textual entity. A Jaccard index is computed between the entity and each of the MeSH candidates, and the concept with the highest score is chosen. When multiple candidates are given the same score, the method looks for lexical inclusion (when a term is fully included in another) between a candidate term and an extracted entity, as it is indicative of a hyperonymy relation. If no candidate matches this criterium, the choice is made at random.

When matching candidates, we take into account certain types of variation: *inflectional variation* by matching either the inflected form or the lemmatized form of the entities ; *orthographical variation* by using a list of spelling variants from the UMLS Specialist Lexicon [8]; *derivational variation* by using lists from the UMLS Specialist Lexicon and by performing stemming (implementation of Porter's algorithm); and *abbreviations* by using the Ab3P tool [10] to obtain the full form of abbreviations.

Because systems participating in the challenge were required to have a rapid response time, we implemented two versions of our normalization method. The first one uses all of the features described above. The second one is a reduced version that uses only stemming and abbreviation recognition to deal with variation (this allows us to reduce the use of lists which usually slow down the process as they need to be loaded).

2.3 Configurations used in the challenge

We tested three different configurations of our system in the evaluation phase of the challenge:

- #1: CRF model maximizing precision and the reduced version of our normalization method
- #2: CRF model maximizing precision and the full normalization method
- #3: CRF model maximizing recall and the full normalization method

2.4 Results

End-to-end performance Table 3 gives the end-to-end performance (i.e., disease name identification and normalization) of our system on the official test corpus of the challenge, in terms of precision, recall and F-measure.

Configurations #1 and #2, based on the CRF model to maximize precision, yielded similar performance (F-measures of 0.8315 and 0.8330, respectively) while configuration #3, based on the CRF model to maximize recall, although achieving a higher recall (0.8662 vs. 0.8526 and 0.8355), resulted in a significant drop in F-measure (0.7135), due to a low precision (0.6066).

Configuration #2, based on the full normalization method, shows the best balance between precision and recall (0.8305 and 0.8355), while configuration #1, based on a reduced version of the normalization method, has a higher recall than precision (0.8526 vs. 0.8114).

Configuration	TP	FP	FN	P	R	F
#1 CRF-precision, reduced normalization	1,695	394	293	0.8114	0.8526	0.8315
#2 CRF-precision, full normalization	1,661	339	327	**0.8305**	0.8355	**0.8330**
#3 CRF-recall, full normalization	1,722	1,117	266	0.6066	**0.8662**	0.7135

Table 3. Results obtained during the evaluation phase of the challenge. Best results are in bold. P = precision, R = recall, F = F-measure

Web-service response time Table 4 shows the response time (msec) of our web-service for each configuration used. As expected, the reduced version of our normalization method allows us to obtained a lower response time (5,213 msec in configuration #1) than the full version (11,240 and 11,627 msec in configurations #2 and #3), with a response time half longer.

Configuration	Average response time per document
#1 CRF-precision, reduced normalization	5,213
#2 CRF-precision, full normalization	11,240
#3 CRF-recall, full normalization	11,627

Table 4. Web-service response time (msec)

3 Discussion

The official results we obtained during the evaluation phase of the challenge are in line with results obtained during the training stage on the disease name identification. Even though the CRF model maximizing recall achieved the best recall (0.8662), its precision—due to a high number of false positives (1,117 vs. 1,988 disease names to be found)—is too low (0.6066) for this model to be useful. Additionally, the CRF model maximizing precision obtained recalls only slightly lower (0.8526 and 0.8355) than the one obtained with the CRF model maximizing recall. As a consequence, our attempts at maximizing recall were unsuccessful.

Concerning the normalization method, the full version (#1 and #3) performed better than the reduced version (#2). Nevertheless, this full version takes twice more time (11,240 msec, #2) as the reduced version (5,213 msec, #1). Since the initial recommendation in this challenge was a time response of 10,000 msec (upgraded to 30,000 msec for the test phase), our configurations #2 and #3 are slightly longer than expected.

As GENIA tagger performed better than the Tree Tagger in our preliminary experiments, a potential direction for future work would consist in using information from the GENIA tagger in the disease name identification, combined with the reduced version of our normalization method (configuration #1). We estimate such a configuration would improve the disease name identification, while providing an acceptable response time.

4 Conclusion

In this paper, we described the INRA/LIMSI system designed for the DNER task at BioCreative 2015. Our system relies on hybrid approaches (machine-learning and linguistic rules) to recognize and normalize disease names in scientific abstracts. Our best configuration achieved a precision of 0.8305, a recall of 0.8355 and a F-measure of 0.8330. Nevertheless, this system is quite slow and average response time per document was of 11,240 msec using this configuration.

References

1. Brown, P.F., Della Pietra, V.J., de Souza, P.V., Lai, J.C., Mercer, R.L.: Class-based n-gram models of natural language. Computational Linguistics 18(4), 467–79 (1992)
2. Golik, W., Warnier, P., Nédellec, C.: Corpus-based extension of termino-ontology by linguistic analysis: a use case in biomedical event extraction. In: Proceedings of the 9th International Conference on Terminology and Artificial Intelligence (2011)
3. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. In: Proc of ICML. pp. 282–9. Williamstown, MA (2001)
4. Lavergne, T., Cappé, O., Yvon, F.: Practical very large scale CRFs. In: Proc of ACL. pp. 504–13. Uppsala, Sweden (July 2010)
5. Li, J., Sun, Y., Johnson, R., Sciaky, D., Wei, C., Leaman, R., Davis, A., Mattingly, C., Wiegers, T., Lu, Z.: Annotating chemicals, diseases, and their interactions in biomedical literature. In: Proc of fifth BioCreative challenge evaluation workshop. Sevilla, Spain (2015)
6. Liang, P.: Semi-supervised learning for natural language. Master's thesis, Massachusetts Institute of Technology (2005)
7. Lindberg, D.A., Humphreys, B.L., McRay, A.T.: The Unified Medical Language System. Methods Inf Med 32(4), 281–91 (1993)
8. McCray, A.T., Srinivasan, S., Brown, A.C.: Lexical methods for managing variation in biomedical terminologies. In: Proceedings of the Annual Symposium on Computer Applications in Medical Care (1994)
9. Schmid, H.: Probabilistic part-of-speech tagging using decision trees. In: Proc of International Conference on New Methods in Language (1994)
10. Sohn, S., Comeau, D.C., W., K., Wilbur, W.J.: Abbreviation definition identification based on automatic precision estimates. BMC bioinformatics 9(1), 402 (2008)
11. Wei, C.H., Peng, Y., Leaman, R., Li, Z.: Overview of the BioCreative V chemical disease relation (CDR) task. In: Proceedings of the fifth BioCreative challenge evaluation workshop. Sevilla, Spain (2015)