Two Different Machine Learning Techniques for Drug-Drug Interaction Extraction

Md. Faisal Mahbub Chowdhury[2,3], Asma Ben Abacha[1],
Alberto Lavelli[2], and Pierre Zweigenbaum[1]

[1] LIMSI-CNRS, BP 133 - F-91403 Orsay Cedex, France
[2] HLT Research Unit, Fondazione Bruno Kessler (FBK), Trento, Italy
[3] Department of Information Eng. and Computer Science, University of Trento, Italy
chowdhury@fbk.eu, abacha@limsi.fr, lavelli@fbk.eu, pz@limsi.fr

Abstract. Detection of drug-drug interaction (DDI) is an important task for both patient safety and efficient health care management. In this paper, we explore the combination of two different machine-learning approaches to extract DDI: *(i)* a feature-based method using a SVM classifier with a set of features extracted from texts, and *(ii)* a kernel-based method combining 3 different kernels. Experiments conducted on the DDIExtraction2011 challenge corpus (unified format) show that our method is effective in extracting DDIs with 0.6398 F_1.

Keywords: Drug-Drug Interaction, machine learning, feature-based method, kernel-based method, tree kernel, shallow linguistic kernel.

1 Introduction

The *drug-drug interaction (DDI)* is a condition when one drug influences the level or activity of another. Detection of DDI is crucial for both patient safety and efficient health care management.

The objective of the *DDIExtraction2011 challenge*[4] was to identify the state of the art for automatically extracting DDI from biomedical articles. We participated in this challenge with a system combining two different machine learning methods to extract DDI: a feature-based method and a kernel-based one. The first approach uses a SVM classifier with a set of lexical, morphosyntactic and semantic features (e.g. trigger words, negation) extracted from texts. The second method uses a kernel which is a composition of a *mildly extended dependency tree (MEDT)* kernel [3], a *phrase structure tree (PST)* kernel [9], and a *shallow linguistic (SL)* kernel [5]. We obtained 0.6398 F-measure on the unified format of the challenge corpus.

In the rest of the paper, we first discuss related works (Section 2). In Section 3, we briefly discuss the dataset. Then in Section 4, we describe the feature-based system. Following that, in Section 5, the kernel-based system is presented. Evaluation results are discussed in Section 6. Finally, we summarize our work and discuss some future directions (Section 7).

[4] http://labda.inf.uc3m.es/DDIExtraction2011/

Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction (DDIExtraction 2011), pages 19–26,
Huelva, Spain, September 2011.

2 Related Work

Several approaches have been applied to biological relation extraction (e.g. protein-protein interaction). Song et al. [13] propose a protein-protein interaction (PPI) extraction technique called PPISpotter by combining an active learning technique with semi-supervised SVMs to extract protein-protein interaction. Chen et al. [2] propose a PPI Pair Extractor (PPIEor), a SVM for binary classification which uses a linear kernel and a rich set of features based on linguistic analysis, contextual words, interaction words, interaction patterns and specific domain information. Li et al. [8] use an ensemble kernel to extract the PPI information. This ensemble kernel is composed with feature-based kernel and structure-based kernel using the parse tree of a sentences containing at least two protein names.

Much less approaches have focused on the extraction of DDIs compared to biological relation extraction. Recently, Segura-Bedmar et al. [11] presented a hybrid linguistic approach to DDI extraction that combines shallow parsing and syntactic simplification with pattern matching. The lexical patterns achieve 67.30% precision and 14.07% recall. With the inclusion of appositions and coordinate structures they obtained 25.70% recall and 48.69% precision. In another study, Segura-Bedmar et al. [12] used shallow linguistic (SL) kernel [5] and reported as much as an F_1 score of 0.6001.

3 Dataset

The DDIExtraction2011 challenge task required the automatic identification of DDIs from biomedical articles. Only the intra-sentential DDIs (i.e. DDIs within single sentence boundaries) are considered. The challenge corpus [12] is divided into training and evaluation dataset. Initially released training data consist of *435* abstracts and *4,267* sentences, and were annotated with *2,402* DDIs. During the evaluation phase, a dataset containing *144* abstracts and *1,539* sentences was provided to the participants as the evaluation data. Both datasets contain drug annotations, but only the training dataset has DDI annotations.

These datasets are made available in two formats: the so-called *unified* format and the *MMTx* format. The unified format contains only the tokenized sentences, while the MMTx format contains the tokenized sentences along with POS tag for each token.

We used the unified format data. In both training and evaluation datasets, there are some missing special symbols, perhaps due to encoding problems. The position of these symbols can be identified by the presence of the question mark *"?"* symbol. For example:

> *<sentence id="DrugDDI.d554.s14" origId="s14" text="Ergotamine or dihydroergotamine***?** *acute ergot toxicity characterized by severe peripheral vasospasm and dysesthesia.">*

4 Feature-based Machine Learning Method

In this approach, the problem is modeled as a supervised binary classification task. We used a SVM classifier to decide whether a candidate DDI pair is an authentic DDI or not. We used the LibSVM tool [1] to test different SVM techniques (nu-SVC, linear kernel, etc.) and the script grid.py, provided by LibSVM, to find the best C and gamma parameters. We obtained the best results by using a C-SVC SVM with the Radial Basis kernel function with the following SVM parameters: c=1.0, g=0.0078125 and the set of features described in sections 4.1 and 4.2.

4.1 Features for DDI Extraction

We choose the following feature set to describe each candidate DDI pair (D1,D2):

- **Word Features.** Include Words of D1, words of D2, words between D1 and D2 and their number, 3 words before D1, 3 words after D2 and lemmas of all these words.
- **Morphosyntactic Features.** Include Part-of-speech (POS) tags of each drug words (D1 and D2), POS of the previous 3 and next 3 words. We use TreeTagger [5] to obtain lemmas and POS tags.
- **Other Features.** Include, among others, verbs between D1 and D2 and their number, first verb before D1 and first verb after D2.

4.2 Advanced features

In order to improve the performance of our system, we also incorporated some more advanced features related to this task. We used lists of interacting drugs, constructed by extracting drug couples that are related by an interaction in the training corpus. We defined a feature to represent the fact that candidate drug couples are declared in this list.

However, such lists are not sufficient to identify an interaction between new drug pairs. We also worked on detecting keywords expressing such relations in the training sentences. The following examples of positive (1,2) and negative (3) sentences show some of the keywords or trigger words that may indicate an interaction relationship.

1. *The oral bioavailability of enoxacin is* ***reduced*** *by 60% with* ***coadministration*** *of ranitidine.*
2. *Etonogestrel may* ***interact*** *with the following medications: acetaminophen (Tylenol) ...*
3. *There have been* ***no*** *formal studies of the* ***interaction*** *of Levulan Kerastick for Topical Solution with any other drugs ...*

To exploit these pieces of semantic information, we defined the following features:

[5] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger

- **Trigger words.** This category of features indicates whether a specific trigger word occurs in the sentence (e.g. induce, inhibit). The trigger words were collected manually from the training corpus.
- **Negation.** This category of features indicates if a negation is detected (e.g. not, no) at a limited distance of characters before, between and after the two considered drugs.

5 Kernel-based Machine Learning Method

In this approach, the DDI extraction task was addressed using a system that exploits kernel-based method. Initially, the data had been pre-processed to obtain relevant information of the tokens of the sentences.

5.1 Data pre-processing

We used the Stanford parser[6] [7] for tokenization, POS-tagging and parsing of the sentences. Having *"?"* in the middle of a sentence causes parsing errors since the syntactic parser often misleadingly considers it as a sentence ending sign. So, we replace all *"?"* with *"@"*. To reduce tokenization errors, if a drug name does not contain an empty space character immediately before and after its boundaries, we inserted blank space characters in those positions inside the corresponding sentence. The SPECIALIST lexicon tool[7] was used to normalize tokens to avoid spelling variations and also to provide lemmas. The dependency relations produced by the parser were used to create dependency parse trees for corresponding sentences.

5.2 System description

Our system uses a composite kernel K_{SMP} which combines multiple tree and feature based kernels. It is defined as follows:

$$K_{SMP}(R_1, R_2) = K_{SL}(R_1, R_2) + w_1{*}K_{MEDT}(R_1, R_2) + w_2{*}K_{PST}(R_1, R_2)$$

where K_{SL}, K_{MEDT} and K_{PST} represent respectively shallow linguistic (SL) [5], mildly extended dependency tree (MEDT) [3] and PST [9] kernels, and w_i represents multiplicative constant(s). The values for all of the w_i used during our experiments are equal to 1.[8] The composite kernel is valid according to the kernel closure properties.

A dependency tree (DT) kernel, pioneered by Culotta et al. [4], is typically applied to the minimal or smallest common subtree of a dependency parse tree

[6] http://nlp.stanford.edu/software/lex-parser.shtml

[7] http://lexsrv3.nlm.nih.gov/SPECIALIST/index.html

[8] Due to time constraints, we have not been able to perform extensive parameter tuning. We are confident that tuning of the multiplicative constant(s) (i.e. w_i) might produce even better performance.

that includes a target pair of entities. Such subtree reduces unnecessary information by placing word(s) closer to its dependent(s) inside the tree and emphasizes local features of the corresponding relation. However, sometimes a minimal subtree might not contain important cue words or predicates. The MEDT kernel addresses this issue using some linguistically motivated expansions. We used the best settings for the MEDT kernel reported by Chowdhury et al. [3] for protein-protein interaction extraction.

The PST kernel is basically the path-enclosed tree (PET) proposed by Moschitti [9]. This tree kernel is based on the smallest common subtree of a phrase structure parse tree, which includes the two entities involved in a relation.

The SL kernel is perhaps the best feature based kernel used so far for biomedical RE tasks (e.g. PPI and DDI extraction). It is a combination of global context (GC) and local context (LC) kernels. The GC kernel exploits contextual information of the words occurring before, between and after the pair of entities (to be investigated for RE) in the corresponding sentence; while the LC kernel exploits contextual information surrounding individual entities.

The jSRE system[9] is the implementation of these kernels using the support vector machine (SVM) algorithm. It should be noted that, by default, the jSRE system uses the ratio of negative and positive examples as the value of the cost-ratio-factor[10] parameter during SVM training.

Segura-Bedmar et al. [12] used the jSRE system for DDI extraction on the same corpus (in the MMTx format) that has been used during the DDIExtraction2011 challenge. They experimented with various parameter settings, and reported as much as an F_1 score of 0.6001. We used the same parameter settings (n-gram=3, window-size=3) with which they obtained their best result.

To compute the feature vectors of SL kernel, we used the jSRE system. The tree kernels and composite kernel were computed using the SVM-LIGHT-TK toolkit[11] [10, 6]. Finally, the ratio of negative and positive examples has been used as the value of the cost-ratio-factor parameter.

6 Results

We split the original training data into two parts by documents. One part contains around 63% of documents (i.e. 276 docs) that have around 67% of the "true" DDI pairs (i.e. 1603). The remaining documents belong to the other part. Both of the systems used these splits.

The first part is used for tuning the systems, while the second part is used as a test corpus for performance evaluation. The results on this test corpus are shown in Table 1. As we can see, the union (on the positive DDIs) of the outputs of each approach is higher than the individual output of the systems. We also calculated results for the intersection (only common positive DDIs) of

[9] http://hlt.fbk.eu/en/technology/jSRE

[10] This parameter value is the one by which training errors on positive examples would outweight errors on negative examples.

[11] http://disi.unitn.it/moschitti/Tree-Kernel.htm

the outputs which decreased the outcome. It is also important to note that the feature-based method (FBM) provides higher precision while the kernel-based method (KBM) obtains higher recall.

	FBM	**KBM**	**Union**	**Intersection**
Precision	0.5910	0.4342	0.4218	0.6346
Recall	0.3640	0.5277	0.6083	0.2821
F_1 Score	0.4505	0.4764	**0.4982**	0.3906

Table 1. Experimental results when trained on 63% of the original training documents and tested on the remaining.

Table 2 shows the evaluation results for the proposed approaches on the final challenge's evaluation corpus. The union of outputs of the systems has produced an F_1 score of **0.6398** which is better than the individual results. The behaviour of precision and recall obtained by the two approaches is the same as observed on the initial corpus (better precision for the feature-based approach and better recall for the kernel-based approach), however, the F_1 score of the kernel-based approach is quite close (F_1 score of *0.6365*) to that of the union.

	FBM	**KBM**	**Union**
True Positive	319	513	532
False Positive	133	344	376
False Negative	436	242	223
True Negative	6138	5927	5895
Precision	**0.7058**	0.5986	0.5859
Recall	0.4225	0.6795	**0.7046**
F_1 Score	0.5286	0.6365	**0.6398**

Table 2. Evaluation results provided by the challenge organisers.

7 Conclusion

In this paper, we have proposed the combination of two different machine learning techniques, a feature-based method and a kernel-based one, to extract DDIs. The feature-based method uses a set of features extracted from texts, including lexical, morphosyntactic and semantic features. The kernel-based method does not use features explicitly, but rather use a kernel composition of MEDT, PST and SL kernels. We have combined these two machine learning techniques and presented a simple union system in the DDIExtraction2011 challenge which obtained encouraging results. We plan to test and add more features in our first method (e.g. UMLS semantic types), and to test the kernel-based method by assigning different weights to the individual kernels of the composite kernel. We also plan to perform further tests with other type of approaches like rule-based methods using manually constructed patterns. Another interesting future work can be to test other algorithms for the combination of different approaches (e.g. ensemble algorithms).

Acknowledgments

One of the authors, MFM Chowdhury, is supported by the PhD research grant of the project “eOnco - Pervasive knowledge and data management in cancer care”.
One of the authors, Asma Ben Abacha, is partially supported by OSEO under the Quaero program.

References

[1] Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm

[2] Chen, Y., Liu, F., Manderick, B.: Extract protein-protein interactions from the literature using support vector machines with feature selection. Biomedical Engineering, Trends, Research and Technologies (2011)

[3] Chowdhury, M.F.M., Lavelli, A., Moschitti, A.: A study on dependency tree kernels for automatic extraction of protein-protein interaction. In: Proceedings of BioNLP 2011 Workshop. pp. 124–133. Association for Computational Linguistics, Portland, Oregen, USA (June 2011)

[4] Culotta, A., Sorensen, J.: Dependency tree kernels for relation extraction. In: Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL ’04). Barcelona, Spain (2004)

[5] Giuliano, C., Lavelli, A., Romano, L.: Exploiting shallow linguistic information for relation extraction from biomedical literature. In: Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL’2006). pp. 401–408. Trento, Italy (2006)

[6] Joachims, T.: Making large-scale support vector machine learning practical. In: Advances in kernel methods: support vector learning, pp. 169–184. MIT Press, Cambridge, MA, USA (1999)

[7] Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics (ACL ’03). pp. 423–430. Association for Computational Linguistics, Sapporo, Japan (2003)

[8] Li, L., Ping, J., Huang, D.: Protein-protein interaction extraction from biomedical literatures based on a combined kernel. Journal of Information and Computational Science 7(5), 1065–1073 (2010)

[9] Moschitti, A.: A study on convolution kernels for shallow semantic parsing. In: Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL ’04). Barcelona, Spain (2004)

[10] Moschitti, A.: Efficient convolution kernels for dependency and constituent syntactic trees. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) Machine Learning: ECML 2006, Lecture Notes in Computer Science, vol. 4212, pp. 318–329. Springer Berlin / Heidelberg (2006)

[11] Segura-Bedmar, I., Martínez, P., Pablo-Sánchez, C.d.: Extracting drug-drug interactions from biomedical texts. BMC Bioinformatics 11(Suppl 5), 9 (2010)

[12] Segura-Bedmar, I., Martínez, P., Pablo-Sánchez, C.d.: Using a shallow linguistic kernel for drug-drug interaction extraction. Journal of Biomedical Informatics In Press, Corrected Proof, Available online (24 April, 2011)

[13] Song, M., Yu, H., Han, W.: Combining active learning and semi-supervised learning techniques to extract protein interaction sentences. In: International Workshop on Data Mining in Bioinformatics (2010)