

FBK-irst : A Multi-Phase Kernel Based Approach for Drug-Drug Interaction Detection and Classification that Exploits Linguistic Information

Md. Faisal Mahbub Chowdhury^{†‡} and Alberto Lavelli[‡]

[‡]Fondazione Bruno Kessler (FBK-irst), Italy

[†]University of Trento, Italy

fmchowdhury@gmail.com, lavelli@fbk.eu

Abstract

This paper presents the multi-phase relation extraction (RE) approach which was used for the DDI Extraction task of SemEval 2013. As a preliminary step, the proposed approach indirectly (and automatically) exploits the scope of negation cues and the semantic roles of involved entities for reducing the skewness in the training data as well as discarding possible negative instances from the test data. Then, a state-of-the-art hybrid kernel is used to train a classifier which is later applied on the instances of the test data not filtered out by the previous step. The official results of the task show that our approach yields an F-score of 0.80 for DDI detection and an F-score of 0.65 for DDI detection and classification. Our system obtained significantly higher results than all the other participating teams in this shared task and has been ranked 1st.

1 Introduction

Drug-drug interaction (DDI) is a condition when one drug influences the level or activity of another. The extraction of DDIs has significant importance for public health safety. It was reported that about 2.2 million people in USA, age 57 to 85, were taking potentially dangerous combinations of drugs (Landa, 2009). Another report mentioned that deaths from accidental drug interactions rose by 68 percent between 1999 and 2004 (Payne, 2007). The DDIExtraction 2011 and DDIExtraction 2013 shared tasks underline the importance of DDI extraction.

The DDIExtraction 2013 task concerns the recognition of drugs and the extraction of drug-drug in-

teractions from biomedical literature. The dataset of the shared task is composed by texts from the Drug-Bank database as well as MedLine abstracts in order to deal with different type of texts and language styles. Participants were asked to not only extract DDIs but also classify them into one of four predefined classes: advise, effect, mechanism and int. A detailed description of the task settings and data can be found in Segura-Bedmar et al. (2013).

The system that we used in this shared task combines various techniques proposed in our recent research activities for relation extraction (RE) (Chowdhury and Lavelli, 2012a; Chowdhury and Lavelli, 2012b; Chowdhury and Lavelli, 2013).¹

2 DDI Detection

Our system performs DDI detection and classification in two separate steps. In this section, we explain how DDI detection (i.e. whether two drug mentions participate in a DDI) is accomplished. DDI classification will be described in Section 3.

There are three phases for DDI detection: (i) discard less informative sentences, (ii) discard less informative instances, and (iii) train the system (a single model regardless of DDI types) on the remaining training instances and identify possible DDIs from the remaining test instances. These phases are described below.

2.1 Exploiting the scope of negations for sentence filtering

Negation is a linguistic phenomenon where a *negation cue* (e.g. *not*) can alter the meaning of a partic-

¹Available in <https://github.com/fmchowdhury/HyREX>.

ular text segment or of a fact. This text segment (or fact) is said to be inside the *scope of such negation (cue)*. In one of our recent papers (Chowdhury and Lavelli, 2013), we proposed how to exploit the scope of negations for RE. We hypothesize that a classifier trained solely on features related to the scope of negations can be used to pro-actively filter groups of instances which are less informative and mostly negative.

To be more precise, we propose to train a classifier (which will be applied before using the kernel based RE classifier mentioned in Section 2.3) that would check whether all the target entity mentions inside a sentence along with possible relation clues (or trigger words), if any, fall (directly or indirectly) under the scope of a negation cue. If such a sentence is found, then it would be identified as less informative and discarded (i.e. the candidate mention pairs inside such sentence would not be considered). During training (and testing), we group the instances by sentences. *Any sentence that contains at least one relation of interest is considered by the less informative sentence (LIS) classifier as a positive (training/test) instance.* The remaining sentences are considered as negative instances.

We use a number of features related to negation scopes to train a binary SVM classifier that filters out less informative sentences. These features are basically contextual and shallow linguistic features. Due to space limitation, we do not report these features here. Interested readers are referred to Chowdhury and Lavelli (2013).

The objective of the classifier is to decide whether all target entity mentions as well as any possible evidence inside the corresponding sentence fall under the scope of a negation cue in such a way that the sentence is unlikely to contain the relation of interest (e.g. DDI). If the classifier finds such a sentence, then it is assigned the negative class label. At present, we focus only on the first occurrence of the negation cues “no”, “n’t” or “not”. These cues usually occur more frequently and generally have larger negation scope than other negation cues.

The LIS classifier is trained using a linear SVM classifier. Its hyper-parameters are tuned during training for obtaining maximum recall. In this way we minimize the number of false negatives (i.e. sentences that contain relations but are wrongly filtered

out). Once the classifier is trained using the training data, we apply it on both the training and test data. However, if the recall of the LIS classifier is found to be below a *threshold value* (we set it to 70.0) during cross validation on the training data of a corpus, it is not used for sentence filtering on such corpus.

Any (training/test) sentence that is classified as negative is considered as a less informative sentence and is filtered out. In other words, such a sentence is not considered for RE. However, it should be noted that, if such a sentence is a test sentence and it contains positive RE instances, then *all these filtered positive RE instances are automatically considered as false negatives during the calculation of RE performance.*

We rule out sentences (i.e. we consider them neither positive nor negative instances for training the classifier that filters less informative sentences) during both training and testing if any of the following conditions holds:

- The sentence contains less than two target entity mentions (such sentence would not contain the relation of interest anyway).
- It has any of the following phrases – “not recommended”, “should not be” or “must not be”.²
- There is no “no”, “n’t” or “not” in the sentence.
- No target entity mention appears in the sentence after “no”, “n’t” or “not”.

2.2 Discarding instances using semantic roles and contextual evidence

For identifying less informative negative instances, we exploit static (i.e. already known, heuristically motivated) and dynamic (i.e. automatically collected from the data) knowledge which has been proposed in Chowdhury and Lavelli (2012b). This knowledge is described by the following criteria:

- **C1:** If each of the two entity mentions (of a candidate pair) has *anti-positive governors* (see Section 2.2.1) with respect to the type of the relation, then they are not likely to be in a given relation.

²These expressions often provide clues that one of the drug entity mentions negatively influences the level of activity of the other.

- **C2:** If two entity mentions in a sentence refer to the same entity, then it is unlikely that they would have a relation between themselves.
- **C3:** If a mention is the abbreviation of another mention (i.e. they refer to the same entity), then they are unlikely to be in a relation.

Criteria C2 and C3 (static knowledge) are quite intuitive. For criterion C1, we construct on the fly a list of *anti-positive governors* (dynamic knowledge) taken from the training data and use them for detecting pairs that are unlikely to be in relation. As for criterion C2, we simply check whether two mentions have the same name and there is more than one character between them. For criterion C3, we look for any expression of the form “Entity1 (Entity2)” and consider “Entity2” as an abbreviation or alias of “Entity1”.

The above criteria are used to filter instances from both training and test data. *Any positive test instance filtered out by these criteria is automatically considered as a false negative during the calculation of RE performance.*

2.2.1 Anti-positive governors

The semantic roles of the entity mentions may indirectly contribute either to relate or not to relate them in a particular relation type (e.g. PPI) in the corresponding context. To put it differently, the semantic roles of two mentions in the same context could provide an indication whether the relation of interest does *not* hold between them. Interestingly, the word on which a certain entity mention is (syntactically) dependent (along with the dependency type) could often provide a clue of the semantic role of such mention in the corresponding sentence.

Our goal is to automatically identify the words (if any) that tend to prevent mentions, which are directly dependent on those words, from participating in a certain relation of interest with any other mention in the same sentence. We call such words *anti-positive governors* and assume that they could be exploited to identify negative instances (i.e. negative entity mention pairs) in advance. Interested readers are referred to Chowdhury and Lavelli (2012b) for example and description of how anti-positive governors are automatically collected from the training data.

2.3 Hybrid Kernel based RE Classifier

As RE classifier we use the following hybrid kernel that has been proposed in Chowdhury and Lavelli (2013). It is defined as follows:

$$K_{Hybrid}(R_1, R_2) = K_{HF}(R_1, R_2) + K_{SL}(R_1, R_2) + w * K_{PET}(R_1, R_2)$$

where K_{HF} is a feature based kernel (Chowdhury and Lavelli, 2013) that uses a heterogeneous set of features, K_{SL} is the Shallow Linguistic (SL) kernel proposed by Giuliano et al. (2006), and K_{PET} stands for the Path-enclosed Tree (PET) kernel (Moschitti, 2004). w is a multiplicative constant that allows the hybrid kernel to assign more (or less) weight to the information obtained using tree structures depending on the corpus. We exploit the SVM-Light-TK toolkit (Moschitti, 2006; Joachims, 1999) for kernel computation. The parameters are tuned by doing 5-fold cross validation on the training data.

3 DDI Type Classification

The next step is to classify the extracted DDIs into different categories. We train 4 separate models for each of the DDI types (one Vs all) to predict the class label of the extracted DDIs. During this training, all the negative instances from the training data are removed. The filtering techniques described in Sections 2.1 and 2.2 are not used in this stage.

The extracted DDIs are assigned a default DDI class label. Once the above models are trained, they are applied on the extracted DDIs from the test data. The class label of the model which has the highest confidence score for an extracted DDI instance is assigned to such instance.

4 Data Pre-processing and Experimental Settings

The Charniak-Johnson reranking parser (Charniak and Johnson, 2005), along with a self-trained biomedical parsing model (McClosky, 2010), has been used for tokenization, POS-tagging and parsing of the sentences. Then the parse trees are processed by the Stanford parser (Klein and Manning, 2003) to obtain syntactic dependencies. The Stanford parser often skips some syntactic dependencies in output. We use the rules proposed in Chowdhury

and Lavelli (2012a) to recover some of such dependencies. We use the same techniques for unknown characters (if any) as described in Chowdhury and Lavelli (2011).

Our system uses the SVM-Light-TK toolkit³ (Moschitti, 2006; Joachims, 1999) for computation of the hybrid kernels. The ratio of negative and positive examples has been used as the value of the cost-ratio-factor parameter. The SL kernel is computed using the jsRE tool⁴.

The K_{HF} kernel can exploit non-target entities to extract important clues (Chowdhury and Lavelli, 2013). So, we use a publicly available state-of-the-art NER system called BioEnEx (Chowdhury and Lavelli, 2010) to automatically annotate both the training and the test data with disease mentions.

The DDIExtraction 2013 shared task data include two types of texts: texts taken from the DrugBank database and texts taken from MedLine abstracts. During training we used both types together.

5 Experimental Results

Table 1 shows the results of 5-fold cross validation for DDI detection on the training data. As we can see, the usage of the LIS and LII filtering techniques improves both precision and recall.

We submitted three runs for the DDIExtraction 2013 shared task. The only difference between the three runs concerns the default class label (i.e. the class chosen when none of the separate models assigns a class label to a predicted DDI). Such default class label is “int”, “effect” and “mechanism” for run 1, 2 and 3 respectively. According to the official results provided by the task organisers, our best result was obtained by run 2 (shown in Table 2).

According to the official results, the performance for “advise” is very low (F_1 0.29) in MedLine texts, while the performance for “int” is comparatively much higher (F_1 0.57) with respect to the one of the other DDI types. In comparison, the performance for “int” is much lower (F_1 0.55) in DrugBank texts with respect to the one of the other DDI types.

In MedLine test data, the number of “effect” (62) and “mechanism” (24) DDIs is much higher than that of “advise” (7) and “int” (2). On the other

³<http://disi.unitn.it/moschitti/Tree-Kernel.htm>

⁴<http://hlt.fbk.eu/en/technology/jsRE>

	P	R	F_1
K_{Hybrid}	0.66	0.80	0.72
LIS filtering + K_{Hybrid}	0.67	0.80	0.73
LIS filtering + LII filtering + K_{Hybrid}	0.68	0.82	0.74

Table 1: Comparison of results for DDI detection on the training data using 5-fold cross validation. Parameter tuning is not done during these experiments.

	P	R	F_1
All text			
DDI detection only	0.79	0.81	0.80
Detection and Classification	0.65	0.66	0.65
DrugBank text			
DDI detection only	0.82	0.84	0.83
Detection and Classification	0.67	0.69	0.68
MedLine text			
DDI detection only	0.56	0.51	0.53
Detection and Classification	0.42	0.38	0.40

Table 2: Official results of the best run (run 2) of our system in the DDIExtraction 2013 shared task.

hand, in DrugBank test data, the different DDIs are more evenly distributed – “effect” (298), “mechanism” (278), “advise” (214) and “int” (94).

Initially, it was not clear to us why our system (as well as other participants) achieves so much higher results on the DrugBank sentences in comparison to MedLine sentences. Statistics of the average number of words show that the length of the two types of training sentences are substantially similar (DrugBank : 21.2, MedLine : 22.3). It is true that the number of the training sentences for the former is almost 5.3 times higher than the latter. But it could not be the main reason for such high discrepancies.

So, we turned our attention to the presence of the cue words. In the 4,683 sentences of the DrugBank training set (which have at least one drug mention), we found that the words “increase” and “decrease” are present in 721 and 319 sentences respectively. While in the 877 sentences of the MedLine training set (which have at least one drug mention), we found that the same words are present in only 67 and 40 sentences respectively. In other words, the presence of these two important cue words in the

DrugBank sentences is twice more likely than that in the MedLine sentences. We assume similar observations might be also possible for other cue words. Hence, this is probably the main reason why the results are so much better on the DrugBank sentences.

6 Conclusion

In this paper, we have described a novel multi-phase RE approach that outperformed all the other participating teams in the DDI Detection and Classification task at SemEval 2013. The central component of the proposed approach is a state-of-the-art hybrid kernel. Our approach also indirectly (and automatically) exploits the scope of negation cues and the semantic roles of the involved entities.

Acknowledgments

This work is supported by the project “eOnco - Pervasive knowledge and data management in cancer care”. The authors would like to thank Alessandro Moschitti for his help in the use of SVM-Light-TK.

References

- E Charniak and M Johnson. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*.
- MFM Chowdhury and A Lavelli. 2010. Disease mention recognition with specific features. In *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*, pages 83–90, Uppsala, Sweden, July.
- MFM Chowdhury and A Lavelli. 2011. Drug-drug interaction extraction using composite kernels. In *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction (DDIExtraction 2011)*, pages 27–33, Huelva, Spain, September.
- MFM Chowdhury and A Lavelli. 2012a. Combining tree structures, flat features and patterns for biomedical relation extraction. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2012)*, pages 420–429, Avignon, France, April.
- MFM Chowdhury and A Lavelli. 2012b. Impact of Less Skewed Distributions on Efficiency and Effectiveness of Biomedical Relation Extraction. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, Mumbai, India, December.
- MFM Chowdhury and A Lavelli. 2013. Exploiting the Scope of Negations and Heterogeneous Features for Relation Extraction: A Case Study for Drug-Drug Interaction Extraction. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technology (NAACL 2013)*, Atlanta, USA, June.
- C Giuliano, A Lavelli, and L Romano. 2006. 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)*, pages 401–408.
- T Joachims. 1999. Making large-scale support vector machine learning practical. In *Advances in kernel methods: support vector learning*, pages 169–184. MIT Press, Cambridge, MA, USA.
- D Klein and C Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, pages 423–430, Sapporo, Japan.
- E Landau. 2009. Jackson’s death raises questions about drug interactions [Published in CNN; June 26, 2009]. <http://edition.cnn.com/2009/HEALTH/06/26/jackson.drug.interaction.caution/index.html>.
- D McClosky. 2010. *Any Domain Parsing: Automatic Domain Adaptation for Natural Language Parsing*. Ph.D. thesis, Department of Computer Science, Brown University.
- A Moschitti. 2004. 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.
- A Moschitti. 2006. Making tree kernels practical for natural language learning. In *Proceedings of 11th Conference of the European Chapter of the Association for computational Linguistics (EACL 2006)*, pages 113–120, Trento, Italy.
- JW Payne. 2007. A Dangerous Mix [Published in The Washington Post; February 27, 2007]. <http://www.washingtonpost.com/wp-dyn/content/article/2007/02/23/AR2007022301780.html>.
- I Segura-Bedmar, P Martínez, and M Herrero-Zazo. 2013. SemEval-2013 task 9: Extraction of drug-drug interactions from biomedical texts. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013)*, Atlanta, USA, June.