UColorado_SOM: Extraction of Drug-Drug Interactions from BioMedical Text using Knowledge-rich and Knowledge-poor Features

Negacy D. Hailu   **Lawrence E. Hunter**   **K. Bretonnel Cohen**

`negacy.hailu@ucdenver.edu`   `larry.hunter@ucdenver.edu`   `kevin.cohen@gmail.com`

University of Colorado, Anschutz Medical Campus

Abstract

In this paper, we present our approach to SemEval-2013 Task 9.2. It is a feature rich classification using LIBSVM for Drug-Drug Interactions detection in the BioMedical domain. The features are extracted considering morphosyntactic, lexical and semantic concepts. Tools like openDMAP and TEES are used to extract semantic concepts from the corpus. The best F-score that we got for Drug-Drug Interaction (DDI) detection is 50% and 61% and the best F-score for DDI detection and classification is 34% and 48% for test and development data respectively.

Keywords: text mining, event extraction, machine learning, feature extraction.

1 Introduction

Our approach to the Semeval 2013 drug-drug interaction task explored the potential for integrating knowledge-based approaches with supervised machine learning. In practice, most supervised machine learning systems are actually hybrids of machine learning and some knowledge-based approach. However, the integration between the two is typically quite loose, with the knowledge-based approach being realized either as heuristic pre-processing or post-processing of the results. The work reported here is an attempt to make a tighter coupling between knowledge-based methods and machine learning. In particular, we took the approach of using knowledge-based methods for feature extraction.

2 Methodology

In this challenge we approach the Drug-Drug interaction task 9.2 as a binary classification problem. A pair of drugs is interacting if there is some kind of influence between the two. Our approach for Drug-Drug interaction extraction 2013 mainly makes use of domain specific morphosyntactic, lexical and semantic features between paired drugs.

We applied Machine Learning classification techniques in order to determine whether a pair of drugs within a biomedical text is interacting or not. For a training set of labeled instances $(\mathbf{X}_i, y_i) = 1, 2, ..., l$ where $\mathbf{X}_i \in R^n$ and $\mathbf{y} \in \{1, -1\}^l$, the support vector machines (SVMs) optimization problem is defined as(Boser et al., 1992) (Cortes and Vapnik , 1995):

$$\hat{\alpha} = \arg \max_{\alpha, w, b} \left(\frac{1}{2} W^T W + C \sum_{i=1}^{l} \alpha_i \right) \quad (1)$$

such that $y_i \left(W^T \phi(X_i) + b \right) > 1 - \alpha_i,$
$$\alpha_i \geq 0.$$

2.1 Materials

The corpus is provided from two data sources. There are 572 documents describing drug-drug interactions from the DrugBank database and 142 abstracts on the subject of drug-drug interactions from MedLine (Isabel et.al., 2011). We prepared datasets for the entire corpus. Each instance in the dataset is a set of paired drugs. In our dataset, there are 27787 instances. 93.57% of them are from Drugbank database and the remaining are from MedLine abstracts. DDI shared task 2013 is not only interaction detection but the challenge also includes detec-

tion of the type of interaction. In our approach, we treated each interaction type as one class.

2.2   Methods

LIBSVM is a library for support vector machines (LIBSVM, 2011). We used this tool for classifying the dataset. Basically, the problem is a multiclass classification problem. We applied the concept of one-vs-all multi-class classification technique to handle the multiple classes.

2.3   Feature Extraction

The features that we extracted for this challenge can be categorized into three types:

2.3.1   Morphosyntactic Features

- **Distance feature:** this is distance between paired drugs in number of words. The intuition here is that the closer two drugs are, the more chance that they might be interacting. Since this feature takes word count as its value, the text is split within white space when counting number of words. Punctuation marks are not considered when counting words.

- **Part-Of-Speech tags:** we chose the GENIA dependency parser for parsing the corpus for two reasons.

- **Dependency parser related features:** we construct the dependency tree using the GENIA dependency parser. Two features are extracted from the tree:

  - Presence of interaction word in the path from the target drug node to the root of the tree.
  - Distance from one target drug name to another one in the tree.

2.3.2   Lexical Features

  - **Bigrams:** a sequence of bigrams is extracted for input text.

2.3.3   Semantic Features

  - **Interaction words:** we collected the top 100 words that indicate drug-drug interaction. The presence of these words is

one feature for our system. The words are checked before and after each target drug. Such words include: increase, decrease, inhibit, interaction, reduce, affect.

- **Presence of preposition within target drugs:** the text within the target drugs is tested to see if it has preposition or not. If the text has a preposition, the value is 1 otherwise it will have zero value.

- **Presence of other drugs within target drugs:** firstly, we collect all drug names into a list. The text within the target drugs is searched for the drug names and the value for this feature will have the number of hits.

- **Concept from OpenDMAP:** OpenDMAP is an ontology-driven, rule-based concept analysis and information extraction system (Hunter et.al., 2008). We used openDMAP to extract drug-drug interaction concepts from the DDI2013 corpus. We extracted pattern based features using OpenDMAP only if OpenDMAP recognizes target drugs.

3   Dataset Preparation

The challenge provided datasets from Drug-Bank database and MedLine abstracts. We split the dataset into 20% development data and 80% training data. Table1 shows the percentage of positive instances in the dataset.

DDI interaction	14.47%
Interaction type effect	6.07%
Interaction type advise	2.97%
Interaction type mechanism	4.75%
Interaction type int	0.68%

Table 1:   positive instances for the different class types

The data is not balanced, as shown in table 1. We penalized the negative classes during training in order to balance the data.

In section 4 we present results for three runs. Run1 includes the basic features which are described in section 2.3. In Run2 we included feature values made available by TEES (Björne

et.al., 2011). In addition to the features in the first two runs, in Run3 the list of interaction words were considered individually as features. In this run, weight penalty and different optimized LIBSVM parameters were considered.

4 Results

Table 2 shows the results for DDI detection only, for both development and test data. The best F1 score is 50% for test data and 61% for development data.

		Runs		
		1	2	3
test data	precision	0.37	0.38	0.4
	recall	0.73	0.75	0.64
	F1	0.49	0.5	0.49
development data	precision	0.28	0.82	0.62
	recall	0.78	0.46	0.59
	F1	0.41	0.59	0.61

Table 2: Partial Evaluation: only detection of DDI

Table 3 shows results for DDI detection and classification. The best F1 score is 34% for test data and 48% for development data.

		Runs		
		1	2	3
test data	precision	0.16	0.25	0.27
	recall	0.32	0.5	0.44
	F1	0.21	0.33	0.34
development data	precision	0.13	0.59	0.49
	recall	0.37	0.33	0.46
	F1	0.2	0.42	0.48

Table 3: Detection and classification of DDI

And finally, the scores for the individual DDI type for the best run are shown in table 4. Apparently, Run3 outperforms in all the scores as can be seen in tables 2 through 4.

		Run3		
		precision	recall	F1
test data	mechanism	0.39	0.29	0.33
	effect	0.21	0.63	0.31
	advise	0.45	0.39	0.42
	int	0.4	0.28	0.334
development data	mechanism	0.5	0.29	0.37
	effect	0.44	0.61	0.51
	advise	0.72	0.46	0.56
	int	0.08	0.1	0.09

Table 4: Best scores for DDI type, Run3

5 Discussion

Generally speaking, the performance of our system is better for DDI detection regardless of their types compared to classifying what kind of DDI they are.

Among the three runs that we submitted for the challenge, Run3 outperforms in all the scores as can be seen in tables 2 through 4 for the following reasons:

- weight penalty techniques are applied in Run3
- optimal cost and gamma parameters are selected while training for Run3
- Bag of interaction words are considered as individual features. This specially increases scores for detecting the individual DDI types.

The best F-score that we got for DDI detection is 61% for development data and 50% for test data as shown in Table 2. The reason why scores are better for DDI detection is that our approach is feature rich DDI detection and we believe that our features mainly target detecting DDIs. A further addition of features that distinguishes the DDI types will hopefully improve the scores for DDI classification. On the other hand, it has been observed that scores are lower for test data compared to development data. And the reason for this is due to optimization parameters that we heuristically chose during training are possibly favoring to development data than to test data. Another possible reason could be overfitting.

As shown in section 4, the knowledge-based lexical features produced our best run. The semantic parser made a smaller contribution to performance, almost certainly because of low coverage- - -historically, in past shared tasks on information extraction, its behavior has been characterized by very high precision but low recall.

5.1 Error Analysis

Table 5 shows false positive predictions collected from the results for Run3. In FP-1, the system predicts detecting the first pair (etanercept and anakinra) correctly and then classifying as type *effect* but it failed to determine whether **etanercept** is interacting with

interleukin-1 antagonist. A close examination of this sentence shows that the last two drugs are separated by parentheses and in fact the last drug is a further explanation of the second one. The system couldn't distinguish this concept — rather it is treating all the three drugs separately and both pairs i.e. (etanercept, anakinra) and (etanercept, interleukin-1 antagonist) are predicted the same. This is happening due the syntactic nature of the text. One possible way to avoid such confusion is to expand the sentence. In other words, we believe initial data clean up might improve the performance of the system. Avoiding punctuation marks such as parenthesis for this case and other delimiters and representing them in words if possible might improve the performance of the classifier.

It is also observed that there is poor prediction for pairs of drugs that have negation. The two examples, i.e. FP-2 and FP-3 in table 5 are wrongly predicted because there is no feature that handles negation in the system.

FP-1	Concurrent administration of **etanercept** (another TNF -blocking agent) and anakinra (an **interleukin-1 antagonist**) has been associated with an increased risk of serious infections, and increased risk of neutropenia and no additional benefit compared to these m edicinal products alone.
FP-2	When used in external subcutaneous infusion pumps for insulin, **NovoLog** should not be mixed with any other **insulins** or diluent.
FP-3	With the exception of **albuterol**, there are no formal studies fully evaluating the interaction effects of **ATROVENT** Inhalation Aerosol and these drugs with respect to effectiveness.

Table 5: False positive samples. In this table false positive DDIs are in bold font.

False negative predictions have a negative effect on the recall evaluation parameter. In table 6 we show false negative predictions and their possible analysis for the development data. A close analysis of FN-1 and FN-2 shows that both sentences have a comma between the paired drugs. From a linguistic point of view, the punctuation mark comma can be used to separate interdependent clauses. Representing this dependency as a feature might help to

avoid false negatives. FN-3 are a bit different and it apprears that there is much knowledge that can be extracted from the given text which is in number format. Currently, the features that we have don't extract information written in numbers. Also, the list of interaction words doesn't include words like **administered, administration** though words like **co-administration, coadministered** are included. A further development of the list of interaction words will avoid such false predictions.

FN-1	Anticholinergic agents: Although **ipratropium bromide** is minimally absorbed into the systemic circulation, there is some potential for an additive interaction with concomitantly used **anticholinergic medications**.
FN-2	Lymphocytopenia has been reported in patients receiving **CAMPTOSAR**, and it is possible that the administration of **dexamethasone** as antiemetic prophylaxis may have enhanced the likelihood of this effect.
FN-3	**Betaseron** administration to three cancer patients over a dose range of 0.025 mg to 2.2 mg led to a dose-dependent inhibition of **antipyrine** elimination.14 The effect of alternate-day administration of 0.25 mg of Betaseron on drug metabolism in MS patients is unknown.

Table 6: False negative samples. In this table false negative DDIs are in bold format.

6 Conclusion

Our approach to Extraction of Drug-Drug Interactions from BioMedical Texts task 9.2 is a feature rich SVM classification. The performance on detecting Drug-Drug interactions is encouraging but it is a bit lower when it comes to further classfying the type of the interaction. As described in section 5.1, addition of features such as negation will reduce false positive prediction and this will increase precision score. Further development of the list of interaction words is also a important task to handle the different forms of words that could indicate an interaction type. We have also observed that pattern-based semantic features are not well extracted in our system.

687

References

Segura-Bedmar, I., Martínez, P, Herrero-Zazo, M. SemEval-2013 Task 9: Extraction of Drug-Drug Interactions from Biomedical Texts. *In Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013)*

Chang, Chih-Chung and Lin, Chih-Jen 2011. *LIBSVM: A library for support vector machines*, volume 2. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`

Hunter, L, Z Lu, J Firby, WA Baumgartner, Jr., HL Johnson, PV Ogren, KB Cohen. . OpenDMAP: An open-source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-type-specific gene expression. *BMC Bioinformatics 2008, 9:78.*

Jari Björne, Filip Ginter, Juho Heimonen, Antti Airola, Tapio Pahikkala and Tapio Salakoski. 2011. *TEES: Event Extraction Software.* Software available at `http://jbjorne.github.com/TEES/`

Isabel Segura-Bedmar, Paloma Martínez, Cesar de Pablo-sachnez Using a shallow linguistic kernel for drug-drug interaction Extraction. 2011. *Journal of Biomedical Informatics*, 44(5):789-804..

Boser, Bernhard E. and Guyon, Isabelle M. and Vapnik, Vladimir N. A training algorithm for optimal margin classifiers. *Proceedings of the fifth annual workshop on Computational learning theory.*

C. Cortes and V. Vapnik. *Support-vector network.* Machine Learning, 20:273-297

Isabel Segura-Bedmar, Paloma Martínez, and Daniel Sánchez-Cisneros The 1st DDIExtraction-2011 challenge task: Extraction of Drug-Drug Interactions from biomedical texts *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Philippe Thomas, Mariana Neves, Illes Solt, Domonkos Tikk, and Ulf Leser. Relation Extraction for Drug-Drug Interactions using Ensemble Learning *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Md. Faisal Mahbub Chowdhury, Asma Ben Abacha, Alberto Lavelli, and Pierre Zweigenbau. Two Different Machine Learning Techniques for Drug-drug Interaction Extraction *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Md. Faisal Mahbub Chowdhury, and Alberto Lavelli. Drug-drug Interaction Extraction Using Composite Kernels *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Jari Björne, Antti Airola, Tapio Pahikkala, and Tapio Salakoski Drug-Drug Interaction Extraction with RLS and SVM Classiffers *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

JAnne-Lyse Minard, Anne-Laure Ligozat, Brigitte Grau, and Lamia Makour Feature selection for Drug-Drug Interaction detection using machine-learning based approaches *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Sandra Garcia-Blasco, Santiago M. Mola-Velasco, Roxana Danger, and Paolo Rosso Automatic Drug-Drug Interaction Detection: A Machine Learning Approach With Maximal Frequent Sequence Extraction *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Jacinto Mata Vázquez, Ramón Santano, Daniel Blanco, Marcos Lucero, and Manuel J. Maña López A machine learning approach to extract drugdrug interactions in an unbalanced cataset *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Stefania Rubrichi, Matteo Gabetta, Riccardo Bellazzi, Cristiana Larizza, and Silvana Quaglini Drug-Drug Interactions Discovery Based on CRFs SVMs and Rule-Based Methods *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Man Lan, Jiang Zhao, Kezun Zhang, Honglei Shi, and Jingli Cai An experimental exploration of drug-drug interaction extraction from biomedical texts *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*

Shreyas Karnik, Abhinita Subhadarshini, Zhiping Wang, Luis Rocha and Lang Li Extraction of drug-drug interactions using all paths graph kernel *Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction*