Automatic Drug-Drug Interaction Detection: A Machine Learning Approach With Maximal Frequent Sequence Extraction

Sandra Garcia-Blasco[1], Santiago M. Mola-Velasco[1], Roxana Danger[2], and Paolo Rosso[3]

[1] bitsnbrains S.L. and Universidad Politécnica de Valencia, Spain – {*sandra.garcia,santiago.mola*}*@bitsnbrains.net*
[2] Imperial College London, UK – *rdanger@imperial.ac.uk*
[3] NLE Lab. - ELiRF, DSIC, Universidad Politécnica de Valencia, Spain – *prosso@dsic.upv.es*

Abstract. A Drug-Drug Interaction (DDI) occurs when the effects of a drug are modified by the presence of other drugs. DDIExtraction2011 proposes a first challenge task, Drug-Drug Interaction Extraction, to compare different techniques for DDI extraction and to set a benchmark that will enable future systems to be tested. The goal of the competition is for every pair of drugs in a sentence, decide whether an interaction is being described or not. We built a system based on machine learning based on bag of words and pattern extraction. Bag of words and other drug-level and character-level have been proven to have a high discriminative power for detecting DDI, while pattern extraction provided a moderated improvement indicating a good line for further research.

1 Introduction

A Drug-Drug Interaction (DDI) occurs when the effects of a drug are modified by the presence of other drugs. The consequences of a DDI may be very harmful for the patient's health, therefore it is very important that health-care professionals keep their databases up-to-date with respect to new DDI reported in the literature.

DDIExtraction2011 proposes a first challenge task, DDI Extraction, to compare different techniques for DDI extraction and to set a benchmark that will enable future systems to be tested. The goal of the competition is for every pair of drugs in a sentence, decide whether an interaction is being described or not. The corpus used was the DrugDDI corpus [1]. Two formats of the corpus were provided, MMTx format and Unified format. Our system uses Unified format, which only contains labels for drugs. Table 1 shows the corpus statistics[4].

The paper is structured as follows: Section 2 overviews related work. Section 3 describes the system used as well as its features. In section 4 we discuss the evaluation and results and in Section 5 we draw some conclusions.

[4] These statistics cover only documents and sentences that contain, at least, one drug pair.

Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction (DDIExtraction 2011), pages 51–58, Huelva, Spain, September 2011.

Table 1. DrugDDI corpus statistics.

	Training	Test	Total
Documents	399	134	533
Sentences	2812	965	3777
Pairs of drugs	23827	7026	30853
Interactions	2397	755	3152

2 Related Work

Even though the problem of DDI extraction is relatively new, some authors have already presented approximations to solve it. In [2], the author presents two approximations to face the problem: a hybrid approach, combining shallow parsing and matching of patterns described by a pharmacist; and an approximation based on kernel methods that obtained better results that the hybrid approach, reaching 55% precision and 84% recall.

In [3] the authors propose a first approximation for DDI detection based on automatically determining the patterns that identify DDI from a training set. The patterns extracted were *Maximal Frequent Sequences* (MFS), based on [4]. In this work, the identified MFS were used to determine whether a sentence contains or not a description of a DDI, without identifying the pair of interacting drugs. MFS have been useful in different tasks such as text summarization [5], measuring text similarities [6] and authorship attribution [7]. MFS will also be part of our approximation, and will be defined further on.

Protein-Protein Interaction (PPI) extraction is an area of research very similar to DDI extraction that has received a bigger attention from the scientific comunity. The BioCreative III Workshop hosted two tasks of PPI document classification and interaction extraction [8]. Some of the features present in a wide range of participants were bag-of-words, bigrams, co-occurrences and character ngrams. This kind of features will have a key role in our system. In [9] the authors use patterns as one of their main features to extract PPI. In [10], the authors use a hybrid approach with clustering and machine learning classification using Support Vector Machines (SVM).

3 Our System

We built a system based on machine learning[5], therefore we had to define a feature set to estimate the model. Each sample is one possible interaction, this is, each unique combination of two drugs appearing in a sentence of the corpus. Given the small size of the corpus and the difficulty of properly estimating the model, it was necessary to represent the features in a reduced space.

The first step was to preprocess the corpus. For doing so, each sentence was tokenized[6] with standard English tokenization rules (e.g. split by spaces, removal

[5] We used RapidMiner for every classification and clustering model. Available at `http://rapid-i.com/`.

[6] The tokenization was performed with Apache Lucene. Available at `http://lucene.apache.org`.

of apostrophes, conversion to lower case, removal of punctuation marks) with the following particularities:

- Each group of tokens that represent a drug were replaced by *#drug#*.
- Numbers were replaced by *_num_*.
- Stop words were not removed.
- Stemming was applied[7].
- Percentage symbols were preserved as independent tokens.

In the following subsections, we will describe the different features used in the system.

3.1 Bag of Words

From the set of all words appearing in the preprocessed corpus, we discarded those with a frequency lower than 3 and stop words. With the resulting set of words, we generated a dataset where each sample was a possible interaction in the corpus and each feature was the presence or not of each word between the two drugs of the potential interaction. Using this dataset, every word was ranked using information gain ratio with respect to the label [8]. Then, every word with an information gain ratio lower than 0.0001 was discarded. The presence of each of the remaining words was a feature in the final dataset. Finally, 1,010 words were kept.

Samples of words with a high gain ratio are: *exceed*, *add*, *solubl*, *amphetamin*, *below*, *lowest*, *second*, *defici*, *occurr*, *stimul* and *acceler*.

3.2 Word Categories

In biomedical literature complex sentences are used very frequently. MFS and bag of words are not able to capture relations that are far apart inside a sentence. To somehow reflect the structure of the sentence, we defined some word categories. This way, we can have some information about dependent and independent clauses, coordinate and subordinate structures, etc. Some of these categories were also included in [2]. We added two categories that include absolute terms and quantifiers, as well as a category for negations. Table 2 enumerates the words included in each category.

For each word category we defined two features. One indicating how many times the words in the category appeared in the sentence, and the other indicating how many times they appeared between the two drugs of the potential interaction.

[7] The stemming algorithm used was Snowball for English. Available at `http://snowball.tartarus.org`.

[8] Information Gain Ratio was calculated using Weka. Available at `http://www.cs.waikato.ac.nz/ml/weka/`.

Table 2. Word Categories.

Category	Words included
Subordinate	*after, although, as, because, before, if, since, though, unless, until, whatever, when, whenever, whether, while.*
Independent markers	*however, moreover, furthermore, consequently, nevertheless, therefore.*
Appositions	*like, including, e.g., i.e.*
Coordinators	*for, and, nor, but, or, yet, so.*
Absolute	*never, always.*
Quantifiers	*higher, lower.*
Negations	*no, not.*

3.3 Maximal Frequent Sequences

Similar to bag of words, we used sequences of words as features. For this, we used Maximal Frequent Sequences (MFS).[9] Following [4], a sequence is defined as an ordered list of elements, in this case, words. A *sequence* is *maximal* if it is not a subsequence of any other, this is, if it does not appear in any other sequence in the same order. Given a collection of sentences, a *sequence* is β*-frequent* if it appears in at least β sentences, where β is the defined frequency threshold. *MFS* are all the sequences that are β*-frequent* and maximal.

We extracted all the MFS from the training corpus, with a β of 10 minimum length of 2. Given the size of the corpus, sometimes very long MFS have no capability to generalize knowledge because they sometimes represent full sentences, instead of patterns that should be frequent in a kind of sentence. To avoid this, we restricted the MFS to a maximum length of 7 words. With this, we obtained 1.010 patterns. In order to reduce the feature space we calculated clusters of MFS.

Clusters were calculated with the Kernel K-Means algorithm [11], using radial kernel, with respect to the relative frequency of each MFS in the following contexts: a) sentences, b) sentences containing an interaction, c) MFS appearing between two drugs, c) MFS appearing before the first drug of an interaction and d) MFS appearing after the last drug of an interaction. Clustering helped to avoid pattern redundancy. This was necessary because some patterns could be considered equivalent since they only differed in one or a few words not relevant in the context of DDI. We obtained 274 clusters. Each of this clusters is a feature of the final dataset which is set to 1 if, at least, one of the MFS of the cluster matches with the potential interaction. The matching algorithm is shown in Algorithm 1.

3.4 Token and Char Level Features

At the token and char level, several features were defined. We must recall that, during preprocessing, every token or group of tokens labeled as drugs where replaced by the token *#drug#*. Table 3 describes this subset of features. Each one of these features appears twice in the final dataset, once computed on the

[9] We used a proprietary library by bitsnbrains, `http://bitsnbrains.net`.

Algorithm 1: MFS matching algorithm.

Input: mfs, $sentence$, $drug1index$, $drug2index$
Output: $match$
$startThreshold \leftarrow 0$
$endThreshold \leftarrow 0$
if *"#drug#"* $\in mfs$ **then**
    $startThreshold \leftarrow$ First index of "#drug#" in mfs
    $endThreshold \leftarrow length(mfs) -$ last index of "#drug#" in mfs
$startIndex \leftarrow drug1index - startThreshold$
if $startIndex < 0$ **then**
    $startIndex \leftarrow 0$
$endIndex \leftarrow drug2index + endThreshold$
if $endIndex > length(sentence)$ **then**
    $endIndex \leftarrow length(sentence)$
$textBetweenDrugs \leftarrow$ Substring of $sentence$ from index $startIndex$ to $endIndex$
if mfs *is subsequence of* $textBetweenDrugs$ **then**
    $match \leftarrow 1$
else
    $match \leftarrow 0$

whole sentence and once computed only in the text between the two drugs of the potential interaction.

Table 3. Token and char level features.

Feature	Description
Tokens	Number of tokens.
Token *#drug#*	Number of times the *#drug#* token appears.
Chars	Number of chars.
Commas	Number of commas.
Semicolons	Number of semicolons.
Colons	Number of colons.
Percentages	Number of times the character % appears.

3.5 Drug Level Features

With the features defined so far, we have not taken into account the two drugs of the potential interaction. We believe this is important in order to have more information when deciding wether if they interact or not.

For each document, we calculated the *main drug* as the drug after which the document was named, this is, the name of the article of the DrugBank database where the text was extracted from. In the case of scientific articles, the main drug would be calculated as the drug or drug names appearing in the title of the article, if any. Also for each document, we calculated the *most frequent drug* as the token labeled as drug that appeared more times in the document.

We noticed that, sometimes, drugs are referred to using their trade names. To ensure good treatment of drugs in the drug level features, we replaced each trade name with the original drug name[10]. Table 4 describes the drug level features.

Table 4. Drug level features for candidate interactions (CI)

Feature	Description
Main drug	True if one of of the two drugs in the CI is the document name.
Most frequent drug	True if one of the two drugs in the CI is the most frequent drug in the document.
Cross reference	True if, at least, one of the two drugs in the CI is *drug*, *medication* or *medicine*.
Alcohol	True if, at least, one of the two drugs in the CI is *alcohol* or *ethanol*.
Is same drug	True if both drugs in the CI are the same.

3.6 Classification Model

During preliminary research, we explored the performance of a wide range of classification models, notably Support Vector Machines, Decision Trees and multiple ensemble classifiers such as Bagging, MetaCost and Random Forests [12]. Our best choice was Random Forest with 100 iterations and 100 attributes per iteration.

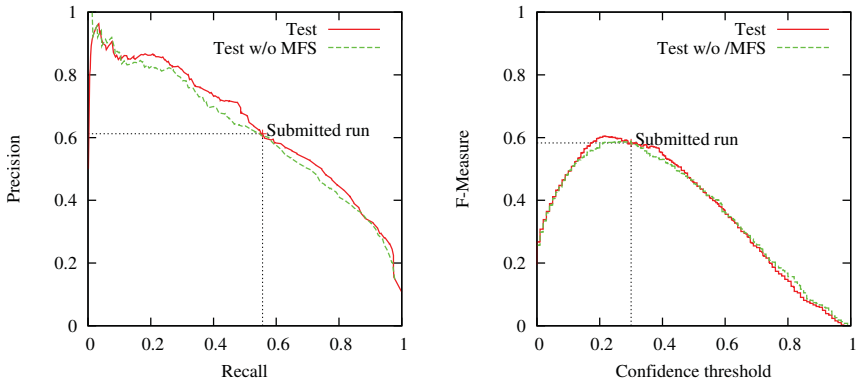
4 Evaluation

We evaluated our model with standard performance measures for binary classification: Precision (P), Recall (R) and F-Measure (F). For each label, our model outputs a confidence value. In order to decide the label, we define a confidence threshold above which the decision will be positive and below which it will be negative. A quick way to visualize every possible set up of the system is the PR curve, where P and R are ploted for different confidence thresholds. Analogously, we can plot F-Measure and confidence thresholds to visualize the optimum threshold with respect F-Measure. AUC-PR is defined as the area under the PR curve. AUC-PR is a very stable measure to compare binary classification models.

We are evaluating the performance of our system for the test set, with and without MFS. Figure 1 shows PR and F curves for both settings. The PR curves are convex, which makes the decision of an optimum threshold much easier and less risky. Table 5 shows Precision, Recall, F-Measure, AUC-PR, precision at recall 0.8 and recall at precision 0.8 for test with MFS.

[10] Trade names were extracted from the KEGG DRUG database, from the Kyoto Encyclopedia of Genes and Genomes. Available at `http://www.genome.jp/kegg/drug/`

MFS improve moderately the performance of the system, increasing about 0.02 in AUC-PR. We expected more influence of MFS. Patterns were extracted using all sentences, even the ones that did not include any drug interaction. We believe that this could have reduced the performance.



(a) Precision-Recall curves (b) F-Measure curves

Fig. 1. PR and F curves for test with and without MFS.

Table 5. Performance measures for test with and without MFS.

	P	R	F	AUC-PR	P@R 0.8	R@P 0.8
Test	0.6122	0.5563	0.5829	0.6341	0.4309	0.3205
Test w/o MFS	0.6069	0.5563	0.5805	0.6142	0.4113	0.2808

5 Conclusions

We presented a system for DDI extraction based on bag-of-words and Maximal Frequent Sequences, as used for the DDIExtraction2011 competition. Our submission obtained a F-Measure of 0.5829 and a AUC-PR of 0.6341 for the test corpus. Our system can be set up to reach recall of 0.3205 with a precision of 0.8, or precision of 0.4309 and a recall 0.8. The use of MFS increased AUC-PR by 0.02.

One of the main problems we have encountered is the complexity of the language structures used in biomedical literature. Most of the sentence contained appositions, coordinators, etc. Therefore it was very difficult to reflect those structures using MFS. The reduced size of the corpus is also a serious limitation for our approach.

Our system should be improved by complementing it with other state-of-the-art techniques used in the PPI field that have not been explored yet during

our participation, such as character n-grams and co-occurrences. It could also be improved by extracting MFS with reduced restrictions and improving the clustering step.

Acknowledgments. This work has been done in the framework of the VLC/ CAMPUS Microcluster on Multimodal Interaction in Intelligent Systems. Contributions of first and second authors have been supported and partially funded by bitsnbrains S.L. Contribution of fourth author has been partially funded by the European Commission as part of the WIQEI IRSES project (grant no. 269180) within the FP 7 Marie Curie People Framework, by MICINN as part of the Text-Enterprise 2.0 project (TIN2009-13391-C04-03) within the Plan I+D+i. Computational resources for this research have been kindly provided by Daniel Kuehn from Data@UrService.

References

1. Segura-Bedmar, I., Martinez, P., de Pablo-Sanchez, C.: Using a shallow linguistic kernel for drug-drug interaction extraction. J. Biomed. Inform. In Press, Corrected Proof, Available online 24 April 2011, DOI 10.1016/j.jbi.2011.04.005.
2. Segura-Bedmar, I.: Application of Information Extraction techniques to pharmacological domain: Extracting drug-drug interactions. PhD thesis, UC3M, Madrid, Spain (April 2010)
3. García-Blasco, S., Danger, R., Rosso, P.: Drug-Drug Interaction Detection: A New Approach Based on Maximal Frequent Sequences. SEPLN **45** (2010) 263–266
4. Ahonen-Myka, H.: Discovery of Frequent Word Sequences in Text. In: Pattern Detection and Discovery. Volume 2447 of LNCS., London, UK, Springer (2002) 180–189
5. García, R.A.: Algoritmos para el descubrimiento de patrones secuenciales maximales. PhD thesis, INAOE, Mexico (September 2007)
6. García-Blasco, S.: Extracción de secuencias maximales de una colección de textos. Final degree project, UPV, Valencia, Spain (December 2009)
7. Coyotl-Morales, R.M., Villaseñor Pineda, L., Montes-y Gómez, M., Rosso, P.: Authorship Attribution using Word Sequences. In: Proc. 11th Iberoamerican Congress on Pattern Recognition, CIARP 2006. Volume 4225 of LNCS., Springer (2006) 844–853
8. Arighi, C., Cohen, K., et al., eds.: Proceedings of BioCreative III Workshop, Bethesda, MD USA (2010)
9. Sullivan, R., Miller, C., Tari, L., Baral, C., Gonzalez, G.: Efficient extraction of protein-protein interactions from full-text articles. IEEE/ACM Trans. Comput. Biology Bioinform. **7**(3) (2010) 481–494
10. Bui, Q.C., Katrenko, S., Sloot, P.M.A.: A hybrid approach to extract proteinprotein interactions. Bioinformatics **27**(2) (2011) 259–265 Code available at http://staff.science.uva.nl/˜bui/PPIs.zip.
11. Zhang, R., Rudnicky, A.I.: A Large Scale Clustering Scheme for Kernel K-Means. In: 16th Conference on Pattern Recognition. Volume 4., Los Alamitos, CA, USA, IEEE Computer Society (2002) 289–292
12. Breiman, L.: Random Forests. Machine Learning **45**(1) (2001) 5–32