Feature Selection for Drug-Drug Interaction Detection Using Machine-Learning Based Approaches

Anne-Lyse Minard[1,2], Lamia Makour[1],
Anne-Laure Ligozat[1,3], and Brigitte Grau[1,3]

[1] LIMSI-CNRS, BP 133, 91403 Orsay Cedex, France
[2] Université Paris-Sud 11, Orsay, France
[3] ENSIIE, Évry, France
firstname.lastname@limsi.fr

Abstract. This paper describes the systems developed for the DDI Extraction challenge. The systems use machine learning methods and are based on SVM by using LIBSVM and SVMPerf tools. Classical features and corpus-specific features are used, and they are selected according to their F-score. The best system obtained an F-measure of 0.5965.

Keywords: relation extraction, machine-learning methods, feature selection, drug-drug interaction, LIBSVM, SVMPerf

1 Introduction

In this paper [4], we present our participation to DDI Extraction challenge. The task was to detect if two drugs in the same sentence are in interaction or not. For example in (1) there is an interaction between *HUMORSOL* and *succinylcholine*, and between *HUMORSOL* and *anticholinesterase agents*, but not between *succinylcholine* and *anticholinesterase agents*.

(1) Possible drug interactions of HUMORSOL with succinylcholine or with other anticholinesterase agents.

The high number of features relevant to recognize the presence of an interaction between drugs in sentence, conducts us to propose systems based on machine-learning methods. We chose classifiers based on SVM because they are used in state-of-art systems for relation extraction. We tested two classifiers: LIBSVM [Chang and Lin2001] and SVMPerf [Joachims2005]. We thought that SVMPerf could improve the classification of the not well represented class, i.e. the interaction class (only 10% of drugs pairs are in interaction), because it gives more tolerance of false positives for the under-represented class. We also worked on feature selection in order to keep the most relevant features. In a first section,

[4] This work has been partially supported by OSEO under the Quaero program.

Proceedings of the 1st Challenge task on Drug-Drug Interaction Extraction (DDIExtraction 2011), pages 43–50, Huelva, Spain, September 2011.

we briefly describe the corpus and the knowledge it enables us to compute based on recurrent relations between same drugs. Then we describe our solution that makes use of LIBSVM and the studies we have done concerning first feature selection to improve the classification made by LIBSVM and second the use of another classifier SVMPerf. We then show the results obtained by our systems.

2 Corpus

2.1 Description

For the challenge we disposed of two corpora composed of biomedical texts collected from the DrugBank database and annotated with drugs [Segura-Bedmar et al.2011]. The development corpus was annotated with drug-drug interactions, and the evaluation corpus was annotated with drugs. We chose to use the corpora in the Unified format. The development corpus is composed of 435 files, which contain 23,827 candidate pairs of drugs including 2,402 drug-drug interactions. The evaluation corpus contains 144 files and 7,026 candidate pairs containing 755 interactions. We split the development corpus into training (1,606 interactions) and test (796 interactions) sub-corpora for the development of our models.

2.2 Knowledge Extracted from the Corpus

For each pair of entities in the development corpus, we searched if this pair is often found in interaction or never in interaction in the corpus. The results of this study are shown in table 1. Between brackets, we indicate the number of pairs that appear at least twice. For example, there are 91 pairs of drugs that always interact and appear more than twice in the corpus.

Table 1. Number of pairs in the development corpus

	training corpus
# entities couple	14,096
# never interact	12,163 (2,706)
# always interact	1,047 (91)
# interact and not	886

These results are kept in a knowledge base that will be combined with the results of the machine-learning method (see 5.1). We can see that the most relevant information coming from this kind of knowledge concerns the absence of interaction.

3 Classification with LIBSVM

We first applied LIBSVM with the features described in [Minard et al.2011] for the i2b2 2010 task about relation extraction. We wanted to verify their relevance

for this task. The system we developed use classical features ([Zhou et al.2005], [Roberts et al.2008]). We added to them some features related to the writing style of the corpus and some domain knowledge. For each pair of drugs all the features are extracted. If there are four drugs in the same sentence, we considered six pairs of drugs. In this section, we describe the sets of features and the classifier.

3.1 Features

We first defined a lot of features, and then with the training and test corpus we did several tests and we kept only the most relevant combination of features for this task. In this section we described the features kept for the detection of interaction.

3.1.1 Coordination

To reduce the complexity of sentences we processed sentences before feature extraction to delete entities (tagged as drug) in coordination with one of the two candidate drugs. We added three features: the number of deleted entities, the coordination words that are the triggers of the deletion (*or*, *and*, a comma), and a feature which indicates that the sentence was reduced. This reduction is applied on 33% pairs of drugs in the training corpus.

3.1.2 Surface Features

The surface features take into account the position of the two drugs in the sentence.

- **Distance** (i.e. number of words [5]) between the two drugs: in the development corpus 88% of drugs in interaction are separated by 1 to 20 words. The value of this feature is a number, and not one or zero like other features.
- **Presence of other concepts** between the two entities: for 82% of the entity pairs in relation in the development corpus there are no other drugs between them.

3.1.3 Lexical Features

The lexical features are composed by the words of the contexts of the two entities, including verbs and prepositions which often express interaction.

- **The words and stems** [6] which constitute the entities. The stems are used to group inflectional and derivational variations altogether.
- **The stems of the three words** at the left and right contexts of candidate entities. After several tests we chose a window of three words; with bigger or smaller windows, precision lightly increases but recall decreases.

[5] The words include also the punctuation signs.

[6] We use the PERL module lingua::stem to obtain the stem of the word: http://snowhare.com/utilities/modules/lingua-stem/.

- **The stems of the words** between candidate concepts, to consider all the words between concepts; the most important information for the classification is located here.
- **The stems of the verbs** in the three words at the left and right of candidate concepts and between them. The verb is often the trigger of the relation: for example in (2) the interaction is expressed by *interact*.

  (2) Beta-adrenergic blocking agents may also **interact** with sympathomimetics.
- **The prepositions** between candidate concepts, for example *with* in (3).

  (3) d-amphetamine **with** desipramine or protriptyline and possibly other tricyclics cause striking and sustained increases in the concentration of d-amphetamine in the brain;

3.1.4 Morpho-Syntactic Features

This features take into account syntactic information for expressing relations.

- **Morpho-syntactic tags** of the three words at the left and right of candidate entities: the tags come from the TreeTagger [Schmid1994].
- **Presence of a preposition** between the two entities, regardless of which preposition it is.
- **Presence of a punctuation sign** between candidate entities, if it is the only "word".
- **Path length in the constituency tree** between the two entities: the constituency trees are produced by the Charniak/McClosky parser [McClosky2010].
- **Lowest common ancestor** of the two entities in the constituency tree.

Figure 1 represents the constituency tree for example (2). The length of the path between *Beta-adrenergic blocking agents* and *sympathomimetics* is 9 and the common ancestor is *S*.

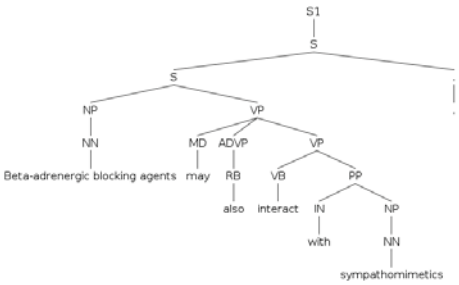

Fig. 1. Example of a constituency tree

3.1.5 Semantic Features

In order to generalize information given by some terms, we also give to the classifier their semantic types.

- **Semantic type (from the UMLS)** of the two entities. In the example (2) the entity *sympathomimetics* has the semantic type *pharmacologic substance*.
- **VerbNet classes** [7] (an expansion of Levin's classes) of the verbs in the three words at the left and right of candidate concepts and between them. For example *increase* is member of the same class as *enhance*, *improve*, etc.
- **Relation between the two drugs in the UMLS**: in the development corpus 57 kinds of relation are found. There is a relation in the UMLS for 5% of drugs pairs in the development corpus. For example, in the UMLS there is a relation *tradename of* between *Procainamide* and *Pronestyl* (4), so the two entities cannot be in interaction.
  (4) - Procainamide (e.g., Pronestyl) or

3.1.6 Corpus-Specific Features

These kinds of features are specific to the DDI corpus.

- A feature indicates **if one of the two drugs is the most frequent drug in the file**. Each file is about one particular drug, so most of the interaction described in the file is between it and another drug.
  A lot of sentences begin with a drug and a semi-colon, like sentence (5). A feature encodes **if one of the two drugs is the same as the first drug** in the sentence.
  (5) **Valproate**: Tiagabine causes a slight decrease (about 10%) in steady-state **valproate** concentrations.
- A feature is set **if one of the two entities is refered to by the term "drug"**: in the training corpus 520 entities are "drug". In this case the expression of the relation can be different (6).
  (6) Interactions between Betaseron and other **drugs** have not been fully evaluated.

3.2 Classifier

We used the LIBSVM tool with a RBF kernel. c and gamma parameters were chosen by the tool *grid.py* with the train corpus for test: c was set at 2 and gamma at 0.0078125. For each class we determined a weight on the parameter c to force the system to classify in the class of interaction. We did tests to choose the value of the weight: for the class of non-interaction the weight is 2 and for the interaction class the weight is 9.

4 Studies from LIBSVM results

This first system obtained 0.56 F-measure on the test corpus. We then made studies on two axes. As the number of features is great, we studied how to reduce it in order to improve the classification. We also studied the application of another classifier which could give more tolerance to false positive to improve the performance of prediction with unbalanced data.

[7] http://verbs.colorado.edu/∼mpalmer/projects/verbnet.html

4.1 Feature Selection

We did a selection of features thanks to the F-score of each feature computed as in [Chen and Lin2006] on the training corpus, prior to the training of the classifier. Given a data set X with m classes, X^k the set of instances in class k, and $|X^k| = l_k, k = 1, ..., m$. Assume $\bar{x}_j^k$ and $\bar{x}_j$ are the average of the j th feature in X^k and X, respectively. The Fisher score of the jth feature of this data set is defined as:

$$\hat{F}(j) = \frac{S_B(j)}{S_W(j)},$$

where

$$S_B(j) = \sum_{k=1}^{m} l_k(\bar{x}_j^k - \bar{x}_j)^2, S_W(j) = \sum_{k=1}^{m} \sum_{x \in X^k} (x_j - \bar{x}_j^k)^2$$

We used the tool *fselect.py*, provided with the LIBSVM library. We defined different thresholds under which we deleted the features. We classified the features in four classes: the semantic class, the morpho-syntactic class, the lexical class and a class with the other features (syntactic, surface, corpus-specific and coordination features). We did tests with different combinations of thresholds for each features class. The best combination of thresholds is described in table 2. This improvement lead to an F-measure of 0.59 on the test corpus. On the full training corpus, we have 368 fewer features after selection, i.e. a total of 9741 features.

Table 2. Best combination of thresholds for feature selection

Semantic class	0.001
Morpho-syntactic class	0.000001
Lexical class	0
Other	0.000004

4.2 SVMPerf

We also tested the SVMPerf tool with a linear kernel. This tool is faster than LIBSVM and optimizes different measures of performance like F1-score or ROC-Area in binary classification. This last measure (ROCArea) allows to choose between different training models. The model is optimal if ROCArea=1, which is the probability to affect the right class to each instance. After training, we changed the value of the threshold b from 1.5 to 1.2. This value was the optimal threshold between the different values that we tested; it increases the performance of prediction with more tolerance of false positives. The c parameter was set at 20 after test of several values with the training corpus.

5 Experimentations and Results

In this section we describe the particularity of each developed system, and finally we give the results obtained at DDI Extraction 2011.

5.1 Experimentations

1. LIMSI-CNRS_4: LIBSVM (baseline)
   This system is the baseline described in 3.
2. LIMSI-CNRS_2: LIBSVM + feature selection
   This system uses lIBSVM with feature selection.
3. LIMSI-CNRS_3: LIBSVM + feature selection (bis)
   This system is the same as the previous one, but the c and gamma parameters differ. The parameters are calculated on the development corpus. The c parameter was set at 2048 and the gamma parameter at 0.0001220703125.
4. LIMSI-CNRS_1: LIBSVM + feature selection + knowledge
   This system is based on LIBSVM. After the classification we combined the prediction of the classifier and the knowledge (cf. section 2.2) in case that their decisions differ. The combination is done as follows: for the class of non-interaction, if the couple exists in the knowledge base and the decision value provided by the classifier is lower than 0.1, the resulting class is the class of the knowledge base. For the interaction class, we keep the class of the knowledge base when the classifier decision value is lower than -0.5.
5. LIMSI-CNRS_5: LIBSVM + SVMPerf (+ feature selection)
   We combine the performance of SVMPerf and LIBSVM by comparing the decision values from each tool. If the two decision values are lower than 0.5, we use the LIBSVM prediction, otherwise we use the prediction with the highest decision value.

5.2 Results and Discussion

The results of the different runs are presented in table 3. The best F-measure is 0.5965 and was obtained by the system which used LIBSVM and combined the prediction of the classfier with the knowledge about pairs of drugs in the training corpus. This F-measure is not significantly different with the F-measure obtained by the system which used LIBSVM without using the knowledge about pairs of drugs in the corpus. So the use of information about the presence or not of the pairs of drugs in the training corpus is not useful for the identification of drugs interaction because the intersection of drugs pairs in the development and evaluation corpus is small (cf. Table 4). There are only 15 pairs that are always in interaction in the development corpus and the evaluation corpus. The best improvement is given by feature selection: without feature selection the system obtained an F-measure of 0.57 and with feature selection of 0.59. However, we can notice that the combination of the two classifiers improve precision.

6 Conclusion

For the DDI Extraction challenge, we developed several methods based on SVM. We showed that a selection of features according to their F-measure improve interaction detection. Reducing the number of features leads to a 0.02 increase of the F-measure. We also showed that SVMPerf is not as efficient as libSVM for this task on this kind of unbalanced data.

Table 3. Results

	Precision	**Recall**	**F-measure**
LIBSVM (baseline)	0.5487	0.6119	0.5786
LIBSVM + feature selection	0.5498	**0.6503**	0.5959
LIBSVM + feature selection (bis)	0.4522	0.5139	0.4811
LIBSVM + feature selection + knowledge	0.5518	0.6490	**0.5965**
LIBSVM (+ feature selection) and SVMPerf	**0.5856**	0.4940	0.5359

Table 4. Intersection between the pairs in the development and the evaluation corpus

		development corpus			
		# never interact	# always interact	# not in development corpus	total
evaluation corpus	# never interact	1,323	100	2,929	4,352
	# always interact	25	15	329	369
	# not in evaluation corpus	10,772	1,008		
	total	12,120	1,123		

References

[Chang and Lin2001] Chih-Chung Chang and Chih-Jen Lin, 2001. *LIBSVM: a library for support vector machines.* Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[Chen and Lin2006] Y. W. Chen and C. J. Lin, 2006. *Combining SVMs with various feature selection strategies.* Springer.

[Joachims2005] Thorsten Joachims. 2005. A support vector method for multivariate performance measures. In *Proceedings of the 22nd international conference on Machine learning*, ICML '05, pages 377–384, New York, NY, USA. ACM.

[McClosky2010] David McClosky. 2010. Any Domain Parsing: Automatic Domain Adaptation for Natural Language Parsing. *PHD Thesis, Department of Computer Science, Brown University.*

[Minard et al.2011] Anne-Lyse Minard, Anne-Laure Ligozat, Asma Ben Abacha, Delphine Bernhard, Bruno Cartoni, Louise Delger, Brigitte Grau, Sophie Rosset, Pierre Zweigenbaum, and Cyril Grouin. 2011. Hybrid methods for improving information access in clinical documents: Concept, assertion, and relation identification.

[Roberts et al.2008] Angus Roberts, Robert Gaizauskas, and Mark Hepple. 2008. Extracting clinical relationships from patient narratives. In *BioNLP2008: Current Trends in Biomedical Natural Language Processing*, pages 10–18.

[Schmid1994] Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, pages 44–49.

[Segura-Bedmar et al.2011] Isabel Segura-Bedmar, Paloma Martinez, and Cesar de Pablo-Sanchez. 2011. Using a shallow linguistic kernel for drug-drug interaction extraction. *Journal of Biomedical Informatics*, In Press, Corrected Proof:–.

[Zhou et al.2005] GuoDong Zhou, Jian Su, Jie Zhang, and Min Zhang. 2005. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 427–434.