# Identifying interactions between chemical entities in biomedical text

**Andre Lamurias[1],[*], João D. Ferreira[1] and Francisco M. Couto[1]**

[1]LaSIGE, Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa, 1749-016, Lisboa, Portugal

### Summary

Interactions between chemical compounds described in biomedical text can be of great importance to drug discovery and design, as well as pharmacovigilance. We developed a novel system, "Identifying Interactions between Chemical Entities" (IICE), to identify chemical interactions described in text. Kernel-based Support Vector Machines first identify the interactions and then an ensemble classifier validates and classifies the type of each interaction. This relation extraction module was evaluated with the corpus released for the DDI Extraction task of SemEval 2013, obtaining results comparable to state-of-the-art methods for this type of task. We integrated this module with our chemical named entity recognition module and made the whole system available as a web tool at `www.lasige.di.fc.ul.pt/webtools/iice`.

## 1 Introduction

One of the major sources of current scientific knowledge is scientific literature, in form of patents, articles or other types of communication. Interactions discovered between chemical compounds are often described in scientific articles [1]. However, the number of documents that a researcher has to retrieve, read and understand to find something useful increases everyday, making this a very time-consuming task. Furthermore, the available drug interaction databases are uneven and unable to correctly identify the interactions with the highest clinical importance [2].

A simple interaction between two chemical compounds may be mentioned in the abstract or full text of a document that may not appear relevant with a simple search by keywords. The interactions found in biomedical documents can be used to validate the results of new research or even to find potentially new interactions between two chemical compounds that interact with a third chemical compound. For example, [3] found that dietary fish oils might benefit patients with Raynaud's syndrome, by connecting the information present in two different sets of articles that did not cite each other. This inference had been confirmed independently by others in clinical trials [4].

Biomedical text mining tasks have attracted the interest of the scientific community in recent years. The DDI extraction task of SemEval 2013 [5] consisted in recognizing drug names

---

[*]To whom correspondence should be addressed. Email: alamurias@lasige.di.fc.ul.pt

mentioned in biomedical texts, and then identifying the interactions described therein. To independently evaluate each step (drug name recognition and drug-drug interactions), the dataset for the Relation Extraction (RE) task was already annotated with chemical entities. These tasks provide not only a gold standard corpus for developing new systems, but also metrics that can be used to compare their performance.

In this paper we introduce a novel system for extracting chemical interactions described in text, named "Identifying Interaction between Chemical Entities" (IICE). We focused on symmetrical binary relations between two chemical compounds mentioned in the same sentence. Even though each participant may have a different role in the relation, we do not consider the order of the entities in the pair as relevant for our system, i.e., if A interacts with B, we assume that B also interacts with A. Examples of interactions that can be identified by our system are the effect that one drug has on the metabolism of another chemical compound, or how one drug should not be administered while a patient is already under the effect of a different drug.

IICE is based on two non-linear kernel-based Support Vector Machine (SVM) classifiers and an ensemble classifier, which combines the results of the SVMs with ontological knowledge to produce a final interaction classification. We combined this system with our Named Entity Recognition (NER) module [6], so that raw text can be analyzed without any prior annotations. We then used the DDI extraction task as a case study for the RE module, achieving results similar to the top performing systems of the DDI extraction task.

The proposed system is available via a web tool, at `www.lasige.di.fc.ul.pt/webtools/iice`. This tool can process a biomedical text by first identifying the chemical entities mentioned in it and then the chemical interactions it describes. It can be potentially used by researchers or database curators who wish to automatically annotate biomedical documents with chemical entities and interactions.

The rest of this paper is organized as follows: Section 2 describes the resources and the general methods we used to develop and evaluate our system, Section 3 presents the results obtained by different approaches used for each module of IICE Section 4 describes the main sources of errors and how our system can be improved and Section 5 presents our main conclusions.

## 2 Methods

### 2.1 Community challenges

There have been several domain challenges to evaluate the performance of biomedical text mining systems. We participated in two of theses challenges with our system: the DDI Extraction task of SemEval 2013, and the CHEMDNER task of BioCreative IV [7]. The two challenges had tasks to identify chemical entities mentioned on biomedical texts. The DDI Extraction task had a second subtask of drug-drug interaction identification, on which we did not participate originally. However, we used the gold standard and the results obtained by the other teams to evaluate the RE module of our system.

## 2.2    Related work

We applied our "Identifying Chemical Entities" (ICE) [8] framework to identify chemical names mentioned in a given text and to map them to a ChEBI identifier.

ICE is based on Conditional Random Fields (CRF) classifiers [9], a machine learning method that can be used to label a sequence of tokens, differentiating between regular words and entities of interest. The main advantage of this framework is its integration with the Chemical Entities of Biological Interest (ChEBI) ontology [10].

We used the MALLET [11] implementation of CRFs with the default parameters. Since MALLET does not, by default, report the probability associated with the calculated most probable sequences, we adapted its source code to include it in the output. This probability was used as a confidence score for each prediction, making it possible to filter predictions with low confidence.

To train models and classify new text, it is necessary to tokenize the text and generate features from word tokens. One of the features that was added in the train set is the final label of the token, i.e. the label we wish to predict. This label is one of "Not Chemical", "Single", "Start", "Middle" or "End", where the last three were included to allow for the detection of chemical entities composed of more than one token. We have used a specifically adapted word tokenizer for chemical text from an open source project [12].

The system we used to participate on the DDI Extraction and CHEMDNER tasks implemented only four features extracted from each tokens:

**Stem:** Stem of the word, determined with the Porter stemming algorithm;
**Prefix and Suffix size 3:** The first and last three characters of a word token;
**Number:** Boolean that indicates if the token contains digits.

We employed an adaptation of FiGO, a lexical similarity method [13], to perform the search for the ChEBI term that most likely corresponds to each identified entity. This yields a similarity score between the word tokens and the ChEBI term, which we call ChEBI mapping score. Terms recognized that cannot be mapped to ChEBI are treated as errors by the CRF classifiers and ignored from this point on. Then, semantic similarity is applied to find the similarity between each ChEBI entity and all the other ChEBI entities detected on the text. Together, the mapping and similarity scores enabled us to filter most false positives and achieve high precision values [14, 15].

## 2.3    Datasets used

The CHEMDNER corpus consists of 10,000 MEDLINE titles and abstracts, partitioned randomly in three: training, development and test [7]. Each annotation consists of the article identifier, type of text (title or abstract), start and end indices, the text string and the type of the Chemical Entity Mention (CEM) which could be one of: trivial, formula, systematic, abbreviation, family and multiple. There was no limit for the number of words that could refer to a CEM but due to the annotation format, the sequence of words had to be continuous. There was

a total of 59,004 annotations on the training and development sets, which consisted in 7,000 abstracts. The test set was composed of 3,000 abstracts and 25,351 annotations. This dataset was used to develop and evaluate the NER module of our system.

Table 1: **Examples of interactions from the DDI corpus. The entities that constitute the interaction are highlighted.**

| DDI type | Sentence |
|---|---|
| advise | Administration of a higher dose of **indinavir** should be considered when coadministering with **megestrol acetate**. |
| effect | When administered concomitantly with **ProAmatine**, **cardiac glycosides** may enhance or precipitate bradycardia, A.V. |
| mechanism | In vivo, the plasma clearance of **ropivacaine** was reduced by 70% during coadministration of **fluvoxamine** (25 mg bid for 2 days), a selective and potent CYP1A2 inhibitor. |
| int | **Trilostane** may interact with **aminoglutethimide** or mitotane (causing too great a decrease in adrenal function). |

The DDI corpus was originally released for task 9 of SemEval 2013, which consisted in extracting drug-drug interactions from biomedical texts [16]. This corpus is composed of 792 texts from the DrugBank database and 233 MEDLINE abstracts, and was partitioned in two sets by its authors: train and test. Each document is annotated with drug names and drug-drug interactions (see Table 1). There was a total of 18,502 chemical entities and 5,028 interactions in this dataset. This dataset was used to develop and evaluate both modules of our system, NER and RE.

Since each data set provided separate train and test sets, we trained the classifiers with the documents from each train set, and evaluated on the test set. The results obtained can then be compared with those obtained by the teams that participated on each competition. The training and development sets of the CHEMDNER corpus were merged into one training set and then we generated one dataset for each type of CEM considered on the CHEMDNER and DDI corpus. With this step, we expected to identify more correct chemical entities since we were including the results of classifiers focused on just one type of CEM. We also employed this step for the different types of DDIs on the DDI corpus.

## 2.4 Chemical named entity recognition module

After participating in the CHEMDNER and DDI Extraction tasks, we have improved the performance of the CRF classifiers by expanding the feature set [6]. This expanded feature set was able to recognize more chemical entities, even with a validation process tuned for high precision. We studied the effect of adding each new feature at a time, while always keeping the four original features constant. These new features are based on orthographic and morphological properties of the words used to represent the entity, inspired by other CRF-based chemical NER systems [17, 18, 19, 20]. Thus, we integrated the following features:

**Prefix and Suffix sizes 1, 2 and 4:** The first and last $n$ characters of a word token.

**Greek symbol:** Boolean that indicates if the token contains Greek symbols.

**Non-alphanumeric character:** Boolean that indicates if the token contains non-alphanumeric symbols.

**Case pattern:** "Lower" if all characters are lower case, "Upper" if all characters are upper case, "Title" if only the first character is upper case and "Mixed" if none of the others apply.

**Word shape:** Normalized form of the token by replacing numbers with "0", uppercase letters with "A", lowercase letters with "a" and other characters with "x".

**Simple word shape:** Simplified version of the word shape feature where consecutive equal symbols are merged.

**Periodic Table element:** Boolean that indicates if the token matches a periodic table symbols or name.

**Amino acid:** Boolean that indicates if the token matches a 3 letter aminoacid code.

An example of some of these features being applied to word tokens is shown on Table 2.

**Table 2: Example of some of the new features and the corresponding label, derived from a sentence fragment (PMID 23194825).**

| Token | Prefix 4 | Suffix 4 | Case pattern | Word shape | Label |
|-------|----------|----------|--------------|------------|-------|
| Cells | Cell | ells | titlecase | Aaaaa | Not Chemical |
| exposed | expo | osed | lowercase | aaaaaaa | Not Chemical |
| to | to | to | lowercase | aa | Not Chemical |
| $\alpha$-MeDA | $\alpha$-Me | MeDA | mixed | xxAaAA | Chemical |
| showed | show | owed | lowercase | aaaaaa | Not Chemical |
| an | an | an | lowercase | aa | Not Chemical |
| increase | incr | ease | lowercase | aaaaaaaa | Not Chemical |
| in | in | in | lowercase | aa | Not Chemical |
| intracellular | intr | ular | lowercase | aaaaaaaaaaaaa | Not Chemical |
| glutathione | glut | ione | lowercase | aaaaaaaaaaa | Chemical |
| ( | ( | ( | - | x | Not Chemical |
| GSH | GSH | GSH | uppercase | AAA | Chemical |
| ) | ) | ) | - | x | Not Chemical |
| levels | leve | vels | lowercase | aaaaaa | Not Chemical |

After running the system with only one new feature each time, we were able to compare the effect of each one on the results. Then, we selected the features that achieved a higher precision, recall and F-measure, creating three sets of features for each metric and a fourth set with all the features tested.

### 2.4.1 Validation processes

As the CRF classifiers are susceptible to recognition errors, we implemented different validation processes that are able to filter out false positives, improving the precision of our system. The three main approaches we employed were: (a) excluding recognized terms with a score lower than a fixed threshold, (b) excluding recognized terms with a combination of multiple

scores lower than a fixed threshold, and (c) training a Random Forest classifier with the scores obtained to classify each term as a true chemical entity or a recognition error.

The scores we computed for each term were the CRF confidence score, previously described, and the ChEBI mapping and SSM scores (see section 2.2). We used the maximum SSM value for each entity as a feature for filtering false positives and as part of a combined score. This value was shown to be crucial to achieve high precision results [21].

To enhance this validation process we introduced a new method to measure semantic similarity based on a concept of relevance, similar to how the h-index measures the impact of the work of a researcher [22]. This concept was applied to the simUI [23] and simGIC [24] measures, so that only ancestors with higher relevance are considered in the computation of the measure. A ChEBI term has index $h$ if $h$ of its $Np$ children have at least $h$ children each and the other $(Np - h)$ children have less than $h$ children each. Then, we consider only ancestors with an index higher than $\alpha$ for the computation of the semantic similarity with the $simUI_\alpha$ and $simGIC_\alpha$ measures. After comparing the results obtained with different $\alpha$ values, the $simGIC_4$ measure provided the best performance at filtering false positives.

## 2.5 Relation extraction module

IICE considers a pair of entities mentioned in the same sentence as a potential interaction, and each element of the pair is considered a candidate. Each pair of entities is classified as a true or false interaction and labeled with one of the DDI types considered in the DDI corpus. This module is also able to bypass the NER module and identify interactions in text that is already annotated.

### 2.5.1 Pre-processing

As a pre-processing step on our RE module, we run the input text through Stanford CoreNLP to extract additional information provided by this tool:

- Part-of-speech (POS) tagging [25].
- Parse tree [26].
- Co-reference resolution [27]: the co-reference annotator is used to replace implicit references to a chemical entity by the representative words. This way, the structure of the sentence is simpler and easier to understand for a classifier. Co-reference resolution was considered to be one of the main source of errors in this task [28].
- Named entity recognition [29]: used to detect numbers, percentages and dates, which can improve the recall since drug interactions are often described with dosages and temporal references [28].

Figure 1 provides an example of our pre-processing method and the type of data generated, which is then used as input for the SVM classifiers.

The names of the chemical entities are replaced in the text by an identifier unique for each sentence. When classifying a pair, we replace the identifiers of the two candidate entities by a

generic string, and all the other chemical entities by a different generic string. This technique has been shown to improve the results of RE systems by ensuring the generality of the classifiers [30].
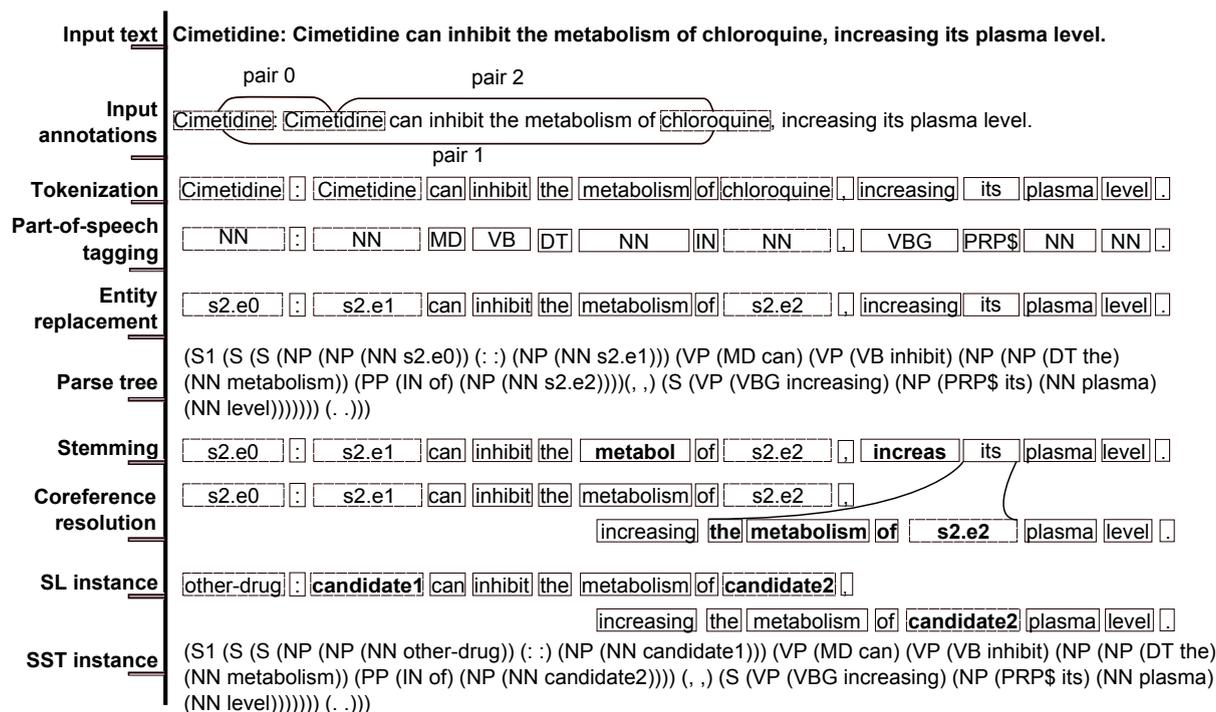


**Figure 1: Pre-processing transformations on the input text for the RE module.**

### 2.5.2  Machine learning for pair classification

Kernel-based methods have gained popularity in the RE field and were used by the teams that achieved the best results at the DDI Extraction task [31, 32]. This type of method is based on a kernel function $K : S \times S \rightarrow [0, \infty[$, which is used to express the similarity between two training instances $x$ and $y$:

$$K(x, y) = \langle f(x), f(y) \rangle$$

where $x, y \in S$ and $f$ is a function that maps an instance to a feature vector, which does not have to be stated explicitly. The kernel function implicitly calculates the dot-product of these feature vectors. This kernel can then be applied to linear machine learning methods, for example SVMs, converting them into non-linear methods. With this method, the focus is shifted from feature selection to kernel construction [33]. This is particularly useful for RE because the instances are not easily expressed by a feature vector.

We applied the Shallow Linguistic (SL) kernel, implemented by the jSRE tool [34] and the SubSet Tree kernel (SST), implemented by the SVM-Light-TK toolkit [35, 36] to classify each pair instance.

The SL kernel is composite kernel that takes into account both the local and global context of the pair elements. We followed the recommendations provided by Segura-Bedmar et al.[37],

on which this kernel was also applied to the DDI corpus, obtaining decent results which we intend to improve upon. Each training instance of this method is the whole sentence tokenized, where the two candidates are assigned a role of "Agent" and "Target". Whenever a candidate was mentioned more than once, by resolving co-references, we added an instance for each combination between the two pairs. This means that the example on Figure 1 would generate 5 instances: 3 explicit pairs and 2 more for the second reference to the s2.e2 entity. As the interactions we considered were symmetric, we always assigned the role of "Agent" to the first candidate and "Target" to the second. This kernel calculates the similarity between two instances by comparing the text, POS tags, stems and label of each token. As such, we used the tokenization, POS tagging and stemming rows from Figure 1 besides the SL instance line. The label of each token was given by the Stanford NER, which could be only "NUMBER", "DATE" or "PERCENTAGE". For every chemical entity, including the ones that did not constitute the pair, we assigned the label "DRUG".

The SST kernel is a tree kernel that calculates the similarity between two instances by computing the number of common subset trees between two trees. For this kernel, we use as input the smallest tree that contains both candidates (SST line of Figure 1) and the default parameters of the tool.

Both kernel method employed classify each pair as interacting or not. We trained one classifier for each kernel method and for each type of interaction, as well as using the whole corpus, resulting in a total of 10 classifiers (4 types of interaction + 1 with the whole corpus for each of the two kernel methods).

### 2.5.3   Ensemble classifier

Even though the results of the kernel classifiers can be used to directly classify the pairs, we implemented an ensemble SVM classifier, which uses as features the output of each RE classifier, along with a set of lexical and domain specific features. We used the SVM implementation of scikit-learn [38, 39] to train and test this classifier The feature set can be organized in three different groups: output of the kernel classifiers, ontological knowledge and presence of certain stems in the sentence.

The features derived from the kernel classifiers can only be 0 or 1, depending on if the pair was classified as interacting or not. For example, if the SL kernel classifier trained with the type "effect" identified the pair as a true interaction, the feature "SL effect" would be equal to 1 for this instance.

Since SSM values have been useful before for filtering false positives on the NER module, we use this information again for the ensemble classifier in this module. We used five different SSMs as features: Resnik, simUI, simGIC, $simUI_4$ and $simGIC_4$, which we had already implemented for the NER module. Moreover, three features based on DrugBank and ChEBI were added to improve the performance of the classifier:

- One candidate is a synonym of the other according to the ChEBI ontology
- Distance between the two candidates if one is an ascendant of the other in the ChEBI ontology (-1 otherwise)

- DrugBank entry for one candidate mentions the other candidate in the list of interactions

**Table 3: Feature set for the ensemble classifier, divided in three groups.**

| Kernel results | | Ontological | Presence of stems in the sentence | | |
|---|---|---|---|---|---|
| Kernel | DDI type | | | | |
| SL | all | Resnik | advanc | advic | affect |
| | effect | simUI | anaesthetis | augment | awar |
| | mechanism | simGIC | bound | care | coadminist |
| | advice | $simUI_4$ | combin | concentr | decreas |
| | int | $simGIC_4$ | effect | exagger | expos |
| SST | all | ChEBI synonym | inhibit | ioniz | lengthen |
| | effect | ChEBI Distance | mechan | metabol | not |
| | mechanism | DrugBank interactions | note | part | prevent |
| | advice | | reach | regul | short |
| | int | | should | warn | withdrawn |

As some terms are more commonly used than others when describing a type of interaction, we compiled a list of 32 stems that suggest the possibility of a DDIs, and added one binary feature for the presence of each word of this list. Finally, we also have another binary feature that has value 1 if the text of the two candidates is the same, since usually these pairs are not interactions. Table 3 shows a summary of the features used for this classifier

This classifier was trained to label each pair with one of the following labels: "mechanism", "effect", "advice", "int" (the four DDI types considered in the training data) or "no-ddi", corresponding to pairs that do not represent an interaction. Finally, we used the evaluator released by the organization of the DDI Extraction task to compute the standard precision, recall and F-measure values.

An overview of the architecture of our system is provided in Figure 2. The system can process raw text without any annotation, or text already annotated with chemical entities, which is what we did to evaluate the RE module, starting the input on the box "Annotated Text".

# 3　Results

**Table 4: Precision, Recall and F-measure estimates for NER using different validation processes, on the test set of the CHEMDNER and DDI corpus. Bold values are the maximum values in each column.**

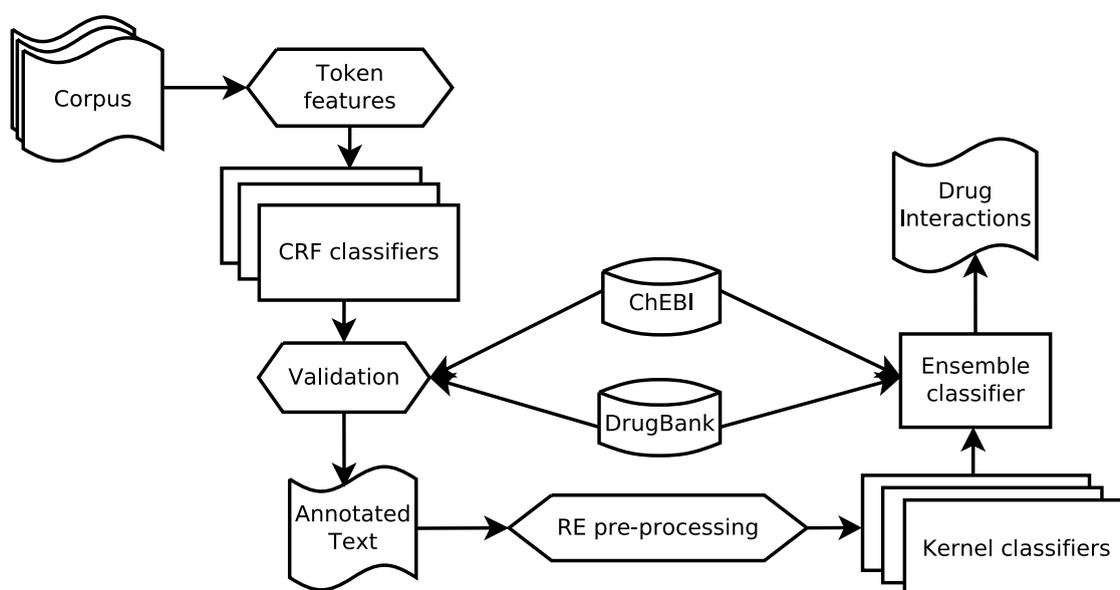| | CHEMDNER | | | DDI | | |
|---|---|---|---|---|---|---|
| | P | R | F | P | R | F |
| ICE 2013 | 87.80% | 65.20% | 74.80% | 82.80% | 73.90% | 78.10% |
| No validation | 58.18% | **80.93%** | 67.70% | 79.40% | **81.49%** | 80.43% |
| SSM | 77.36% | 46.64% | 58.20% | 86.56% | 31.92% | 46.65% |
| Combination | 68.96% | 33.13% | 44.76% | **91.25%** | 56.27% | 69.61% |
| Random Forest | **88.25%** | 70.31% | **78.26%** | 89.25% | 76.24% | **82.23%** |

**Figure 2: Overview of the system architecture.**

Table 4 shows the results we obtained using the CHEMDNER and DDI, for the entity recognition task. We considered true positives only the entities that matched exactly the offsets of the gold standard, and did not attempt to classify the type of entity, which was asked for the DDI Extraction task. In this table, ICE 2013 refers to the best results obtained previously with that corpus, for the respective competition, with our system. The validation processes in the rows of the table refer to the ones mention in the Methods section: "SSM" consists in filtering by one score, in this case, the SSM score; "Combination" consists in filtering by a combination of scores, in this case, the average of the three highest scores for each results; and "Random Forest" refers to the Random Forest classifier.

On the CHEMDNER corpus, the best F-measure was of 78.26%, using the Random Forests validation, which is an improvement over our previous best F-measure (74.80%). The best F-measure on the DDI Corpus was of 82.23%, also with the Random Forests validation. On the DDI corpus, the results were higher than on the CHEMDNER corpus. However, the previous version of our system also performed better on the DDI corpus.

**Table 5: Precision, Recall and F-measure estimates for RE the test set of the DDI corpus.**

|          | Task           | P      | R      | F      |
|----------|----------------|--------|--------|--------|
| Kernel   | Detection      | 70.32% | 79.37% | 74.57% |
|          | Classification | 49.95% | 56.38% | 52.98% |
| Ensemble | Detection      | 80.20% | 66.19% | 72.52% |
|          | Classification | 70.79% | 58.43% | 64.02% |

To evaluate the RE module, we compared the results obtained with only the kernel methods, to the results obtained using also the ensemble classifier. In the first case, we considered a true DDI any pair classified by at least one classifier. If it was classified by more than one type-specific classifier, or only by the whole-corpus classifier, we selected the type that was most frequent in the training data. The order of types, from most to least frequent, was: "effect",

"mechanism", "advice" and "int". Otherwise, the DDI type was the one of the classifier that identified that DDI.

Two types of task were evaluated: the detection task consisted in simply labeling each pair as a DDI or not, while the classification task consisted in classifying each pair with one type of DDI or none. Table 5 shows the results obtained by training the classifiers with the training set and then testing on the test set. The ensemble classifier improved the precision of results for the detection and classification tasks, and also the F-measure of the classification task. The best F-measure for the detection task was 74.57%, using only the kernel methods, and for the classification task it was 64.02%, using the ensemble classifier.

## 4　Discussion

We improved the results obtained previously for two different corpora, by expanding the feature set to include more general and specific features, and by optimizing the SSM score used to filter false positives. These changes improved the results by 3.46 percentage points on the CHEMDNER test set and 4.13 percentage points on the DDI test set. The results obtained are now closer to the top performing systems of the two competitions, particularly with the DDI corpus, where the difference in F-measure to the best run of that competition was 1.07 percentage points.

For the RE module, our assumption was that an ensemble of classifiers and features would provide better results than using only one machine learning methods. In fact, just by using the two kernel methods, we obtained an acceptable F-measure, since the recall was maximized with this strategy. The kernel results provide a baseline for the ensemble classifier.

Without the ensemble classifier, we were able to achieve higher recall values, since the positive pairs of two different classifiers were merged, but at the cost of lower precision. This classifier was able to generally increase the precision, particularly on the classification task. This task was more complex and, for this reason, the results were considerably lower: our highest F-measure for detection was 74.57% while for classification, it was 64.02%. However, the ensemble classifier was able to reduce the difference between the F-measure of detection and classification by 12.11 percentage points on the train set and 13.48 percentage points on the test set. The main factor for this reduction was the increase in precision by the ensemble classifier, which uses machine learning to label the pairs with a DDI type. The ensemble classifier improved both precision and recall of the classification task. While it is still 7.76 percentage points lower than the recall of the detection task, this is an improvement over the classification results of the kernel methods.

However, the main advantage of the ensemble classifier was that it assigned the DDI types with more precision than the rule we used for merging the results of the kernel methods. Hence, we were able to increase the precision by 20.84 percentage points for the classification task. The results obtained were close to the best team of the detection and classification tasks of the DDI Extraction challenge (F-measure of 80.0% and 65.1%, respectively). Even though we did not achieve better results than the top systems of these competitions, our system is almost

independent of external sources, using only the ChEBI ontology and DrugBank for domain knowledge.

The corpus used to train the kernel classifiers for RE was annotated only with drug entities and DDIs. Although this may limit the scope of interactions that are detected by our system, we also used another corpus for NER of chemical compounds. We believe that the kernel classifiers are also able to identify other types of chemical interactions if they are trained with appropriate corpora that contains those interaction types, even though this should be further tested. It should be noted that only the kernel based features of the ensemble classifier depend on the DDI corpus, and the other features are valid for general chemical interactions.

## 4.1   Error analysis

Analyzing the false negatives committed by our RE module, we verified that many were caused by coordinate structures that were not resolved correctly by the parser. When one entity interacts with another, and a list of examples for the second entity is provided, the system may not identify the interactions between the first entity and the examples. For instance, in the sentence "The induction dose requirements of DIPRIVAN Injectable Emulsion may be reduced in patients with intramuscular or intravenous premedication, particularly with narcotics (eg, morphine, meperidine, and fentanyl, etc.)", the system identified the interaction between "DIPRIVAN" and "narcotics", but not between "DIPRIVAN" and each of the narcotics mentioned. This problem could be mitigated by the use of ChEBI's class hierarchy, which states that morphine and fentanyl are subclasses of narcotics. Thus, an interaction between DIPRIVAN and narcotics implies the same type of interaction between the pairs (DIPRIVAN, morphine) and (DIPRIVAN, fentanyl) (note that the compound meperidine is not part of ChEBI).

Furthermore, the method we employed for resolving co-references is limited since it was not optimized for biomedical text, which can have complex sentence structures. In the sentence "It is reasonable to employ appropriate clinical monitoring when potent cytochrome P450 enzyme inducers, such as phenobarbital or rifampin, are co-administered with montelukast.", the system was unable to identify the interaction between "phenobarbital" and "montelukast". By employing for natural language models specific for biomedical text, this type of errors could be less frequent. Two models have been proposed, for co-reference resolution [40] and dependency parsing [41], which we intended to integrate on our system in the future.

We verified that 17 DDIs that the kernel methods were unable to identify were then correctly identified by the ensemble classifier using the domain and stem features. For example, the pair DDI-DrugBank.d585.s0.p2 of the DDI corpus, which is an interaction between "anticholinergic drugs" and "quinidine", was not identified by the kernel methods, possibly because of the complex structure of that sentence, which has 14 chemical entities. The ensemble classifier correctly identified this pair as an interaction of the type effect. For this reason, the ensemble classifier improves not only the precision of the type classification task, but also the recall.

## 5 Conclusion

We have developed IICE to analyze biomedical text using machine learning methods to identify chemical entities and interactions between them. The system can accept raw text, for example, a MEDLINE abstract, or text already annotated with chemical entities to identify the relations between them. IICE is accessible via a free web tool, accessible at `www.lasige.di.fc.ul.pt/webtools/iice`. The user can insert a text or PMID to be annotated in a text box. Various options are available, according to the methods described in this paper, which will have influence in the results obtained. It is also possible to bypass the NER module, assuming the chemical entities are inside '<entity>' tags.

Our module for extracting interactions between chemical entities is based on kernel methods for SVMs. We trained an ensemble classifier with features derived from the kernel and domain knowledge to classify each interaction with a specific type. This module achieved results close to the state-of-the-art for this type of task, while using only one main external source.

Since we used only two primary classifiers, we intend to implement other types of machine learning methods that could identify interactions missed by the ones we used, for example, using Conditional Random Fields and Structured Support Vector Machines [42]. Furthermore, the NER module could be improved by also providing a type for the chemical entities recognized, and the RE module could be improved by training classifier with a corpus annotated with a different type of chemical interactions.

## Acknowledgements

## References

[1] J. Aronson. Communicating information about drug interactions. *British Journal of Clinical Pharmacology*, 63(6):637–639, 2007.

[2] J. Abarca, D. C. Malone, E. P. Armstrong, A. J. Grizzle, P. D. Hansten, R. C. Van Bergen and R. B. Lipton. Concordance of severity ratings provided in four drug interaction compendia. *Journal of the American Pharmacists Association: JAPhA*, 44(2):136–141, 2003.

[3] D. R. Swanson. Medical literature as a potential source of new knowledge. *Bulletin of the Medical Library Association*, 78(1):29, 1990.

[4] R. A. DiGiacomo, J. M. Kremer and D. M. Shah. Fish-oil dietary supplementation in patients with Raynaud's phenomenon: a double-blind, controlled, prospective study. *The American Journal of Medicine*, 86(2):158–164, 1989.

[5] I. Segura-Bedmar, P. Martınez and M. Herrero-Zazo. SemEval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). *Proceedings of Semeval*, pages 341–350, 2013.

[6] A. Lamurias, J. Ferreira and F. M. Couto. Chemical named entity recognition: Improving recall using a comprehensive list of lexical features. In *8th International Conference on Practical Applications of Computational Biology & Bioinformatics (PACBB 2014)*, pages 253–260. Springer, 2014.

[7] M. Krallinger, F. Leitner, O. Rabal, M. Vazquez, J. Oyarzabal and A. Valencia. Overview of the chemical compound and drug name recognition (chemdner) task. In *BioCreative Challenge Evaluation Workshop*, volume 2, page 2. 2013.

[8] T. Grego, P. Pęzik, F. M. Couto and D. Rebholz-Schuhmann. Identification of chemical entities in patent documents. In *Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living*, pages 942–949. Springer, 2009.

[9] J. Lafferty, A. McCallum and F. C. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data, 2001.

[10] J. Hastings, P. de Matos, A. Dekker et al. The ChEBI reference database and ontology for biologically relevant chemistry: enhancements for 2013. *Nucleic Acids Research*, 41(D1):D456–D463, 2013.

[11] A. K. McCallum. Mallet: A machine learning for language toolkit, 2002.

[12] P. Corbett, C. Batchelor and S. Teufel. Annotation of chemical named entities. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, pages 57–64. Association for Computational Linguistics, 2007.

[13] F. M. Couto, M. J. Silva and P. M. Coutinho. Finding genomic ontology terms in text using evidence content. *BMC Bioinformatics*, 6(Suppl 1):S21, 2005.

[14] A. Lamurias, T. Grego and F. M. Couto. Chemical compound and drug name recognition using CRFs and semantic similarity based on chebi. In *BioCreative Challenge Evaluation Workshop*, volume 2, page 75. 2013.

[15] T. Grego, F. Pinto and F. M. Couto. LASIGE: using conditional random fields and chebi ontology. *Proceedings of SemEval*, pages 660–666, 2013.

[16] M. Herrero-Zazo, I. Bedmar, P. Martínez and T. Declerck. The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of Biomedical Informatics*, 46(5):914–920, 2013.

[17] T. Huber, T. Rocktäschel, M. Weidlich, P. Thomas and U. Leser. Extended feature set for chemical named entity recognition and indexing. In *BioCreative Challenge Evaluation Workshop vol. 2*, page 88. 2013.

[18] R. T. Batista-Navarro, R. Rak and S. Ananiadou. Chemistry-specific features and heuristics for developing a CRF-based chemical named entity recogniser. In *BioCreative Challenge Evaluation Workshop vol. 2*, page 55. 2013.

[19] A. Usié, J. Cruz, J. Comas, F. Solsona and R. Alves. A tool for the identification of chemical entities (CheNER-BioC). In *BioCreative Challenge Evaluation Workshop vol. 2*, page 66. 2013.

[20] D. Campos, S. Matos and J. L. Oliveira. Chemical name recognition with harmonized feature-rich conditional random fields. In *BioCreative Challenge Evaluation Workshop vol. 2*, page 82. 2013.

[21] T. Grego and F. M. Couto. Enhancement of chemical entity identification in text using semantic similarity validation. *PloS one*, 8(5):e62984, 2013.

[22] J. E. Hirsch. An index to quantify an individual's scientific research output. *Proceedings of the National academy of Sciences of the United States of America*, 102(46):16569, 2005.

[23] R. Gentleman. Visualizing and distances using GO, 2005. URL `http://www.bioconductor.org/docs/vignettes.html`.

[24] C. Pesquita, D. Faria, H. Bastos, A. Ferreira, A. Falcao and F. Couto. Metrics for GO based protein semantic similarity: a systematic evaluation. *BMC Bioinformatics*, 9(Suppl 5):S4, 2008.

[25] K. Toutanova and C. D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pages 63–70. Association for Computational Linguistics, 2000.

[26] D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430. Association for Computational Linguistics, 2003.

[27] H. Lee, A. Chang, Y. Peirsman, N. Chambers, M. Surdeanu and D. Jurafsky. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4):885–916, 2013.

[28] I. Segura-Bedmar, P. Martínez and M. Herrero-Zazo. Lessons learnt from the DDIExtraction-2013 shared task. *Journal of Biomedical Bnformatics*, 2014.

[29] J. R. Finkel, T. Grenager and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics, 2005.

[30] S. Pyysalo, R. Sætre, J. Tsujii and T. Salakoski. Why biomedical relation extraction results are incomparable and what to do about it. In *Proceedings of the Third International Symposium on Semantic Mining in Biomedicine (SMBM 2008). Turku*, pages 149–152. 2008.

[31] M. F. M. Chowdhury and A. Lavelli. FBK-irst: A multi-phase kernel based approach for drug-drug interaction detection and classification that exploits linguistic information. *Atlanta, Georgia, USA*, page 351, 2013.

[32] P. Thomas, M. Neves, T. Rocktäschel and U. Leser. WBI-DDI: Drug-drug interaction extraction using majority voting. In *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 628–635. Citeseer, 2013.

[33] D. Zelenko, C. Aone and A. Richardella. Kernel methods for relation extraction. *The Journal of Machine Learning Research*, 3:1083–1106, 2003.

[34] C. Giuliano, A. Lavelli and L. Romano. Exploiting shallow linguistic information for relation extraction from biomedical literature. In *EACL*, volume 18, pages 401–408. Citeseer, 2006.

[35] A. Moschitti. Making tree kernels practical for natural language learning. In *EACL*, pages 113–120. 2006.

[36] T. Joachims. Making large-scale support vector machine learning practical. In B. Schölkopf, C. J. C. Burges and A. J. Smola (editors), *Advances in Kernel Methods*, pages 169–184. MIT Press, Cambridge, MA, USA, 1999.

[37] I. Segura-Bedmar, P. Martinez and C. de Pablo-Sánchez. Using a shallow linguistic kernel for drug–drug interaction extraction. *Journal of Biomedical Informatics*, 44(5):789–804, 2011.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[39] C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[40] I. Segura-Bedmar, M. Crespo, C. de Pablo-Sánchez and P. Martínez. Resolving anaphoras for the extraction of drug-drug interactions in pharmacological documents. *BMC Bioinformatics*, 11(Suppl 2):S1, 2010.

[41] D. Mcclosky. *Any Domain Parsing: Automatic Domain Adaptation for Natural Language Parsing*. Ph.D. thesis, Brown University, Providence, RI, USA, 2010. AAI3430199.

[42] I. Tsochantaridis, T. Hofmann, T. Joachims and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the Twenty-First International Conference on Machine Learning*, page 104. ACM, 2004.