# Exploring Word Embedding for Drug Name Recognition

**Isabel Segura-Bedmar, Víctor Suárez-Paniagua, Paloma Martínez**
Computer Science Department
University Carlos III of Madrid, Spain
{isegura,vspaniag,pmf}@inf.uc3m.es

## Abstract

This paper describes a machine learning-based approach that uses word embedding features to recognize drug names from biomedical texts. As a starting point, we developed a baseline system based on Conditional Random Field (CRF) trained with standard features used in current Named Entity Recognition (NER) systems. Then, the system was extended to incorporate new features, such as word vectors and word clusters generated by the Word2Vec tool and a lexicon feature from the DINTO ontology. We trained the Word2vec tool over two different corpus: Wikipedia and MedLine. Our main goal is to study the effectiveness of using word embeddings as features to improve performance on our baseline system, as well as to analyze whether the DINTO ontology could be a valuable complementary data source integrated in a machine learning NER system. To evaluate our approach and compare it with previous work, we conducted a series of experiments on the dataset of SemEval-2013 Task 9.1 Drug Name Recognition.

## 1 Introduction

The automatic recognition of biomedical entities from scientific texts can markedly reduce the time that experts spend populating biomedical knowledge bases and annotating papers and patents. Furthermore, Named Entity Recognition (NER) is a crucial component for many Natural Language Processing (NLP) systems such as relation extraction, text classification or sentiment analysis systems, among many others.

Conditional Random Fields (CRF) often show best results in the recognition of drugs and chemical names (Krallinger et al., 2015a; Segura Bedmar et al., 2013). So far the most popular features for CRF-based NER systems concern syntactic and semantic properties of words (such as tokens, part-of-speech (POS) tags, lemmas, orthographic and lexicon features, among others). In this work, we develop a system based on a CRF to recognize drug mentions occurring in the DDI corpus (Herrero-Zazo et al., 2013)[1]. It consists of two different datasets: DDI-DrugBank (792 texts selected from the DrugBank database) and DDI-MedLine (233 MedLine abstracts on the subject of DDIs). This corpus will allow us to compare our system to the participating systems in the SemEval-2013 Task 9.1 DrugNER Task.

One of the goals of this paper is to study whether the DINTO ontology[2] (Herrero Zazo, 2015) can provide valuable information for this task. As far as we know, DINTO is the first ontology providing a comprehensive and accurate representation of drug-drug interactions (DDI) knowledge. The DINTO ontology contains a total of 25,809 classes, in particular 8,786 drugs and 11,555 DDIs. Several domain resources such as the CheBI ontology (Degtyarenko et al., 2008), the DrugBank database (Wishart et al., 2006) or the OAE ontology (He et al., 2014) have been reused to create DINTO. Furthermore, it was designed to be used by the computer science community working on the DDI domain. A detailed description of the DINTO ontology can be found in Herrero-Zazo's PhD thesis (Herrero Zazo, 2015).

As the main contribution, this work explores the effectiveness of new features for the Drug NER task, in particular, word clusters and word vectors generated using the Word2Vec tool (Mikolov et al., 2013a), a word embedding model based on a neural network (NN). We hypothesize that the use

[1] http://labda.inf.uc3m.es/ddicorpus
[2] http://www.obofoundry.org/cgi-bin/detail.cgi?id=DINTO



*Proceedings of the Sixth International Workshop on Health Text Mining and Information Analysis (Louhi)*, pages 64–72,
Lisbon, Portugal, 17 September 2015. 

of word embedding features would allow us to accurately detect even those drugs that are not in the training set or in the DINTO ontology. A word embedding is a function to map words to high-dimensional vectors. At present, NN is one of the most used learning techniques for generating word embeddings (Mikolov et al., 2013b). The essential assumption of word embedding is that semantically close words will have similar vectors. Word embeddings have shown promising results in NLP tasks, such as named entity recognition, sentiment analysis or parsing (Turian et al., 2010; Socher et al., 2013a; Socher et al., 2013b). However, to the best of our knowledge, this technique has hardly ever been exploited in drug name recognition (Liu et al., 2015).

In fact, our work is the first to explore the word embedding potential using the whole word2vec vector for drug name recognition. In contrast to (Liu et al., 2015), we also train the word embedding features (word clusters and word vectors) using the latest wikipedia dump[3], which contains more than 3 billion words, as well as the 2013 release of MedLine[4], which they used for genereting their word representations. This release contains approximately one million words, being thus much smaller than the Wikipedia collection. While MedLine is a biomedical literature database, Wikipedia covers many different domains of knowledge. However, we believe that the larger the dataset used for training the Word2Vec models, the better word embeddings should be obtained. Thus, we would like to compare the effectiveness of word embeddings features trained on a specific domain corpus, such as MedLine, to those trained on a larger collection, such as Wikipedia.

Another key difference of our work with (Liu et al., 2015) is that while they only gave results for the whole DDI corpus, we analyze and discuss the effect of the DINTO and word2vec features on each one of the datasets: DDI-DrugBank and DDI-MedLine. This analysis is necessary in order to know what features are more efficient on each dataset. MedLine abstracts are very different from DrugBank texts. While abstracts are mainly addressed to scientists in life sciences, texts from DDI-DrugBank are written in a language understandable to patients.

The paper is organized as follows. In the next section, we introduce the two main shared tasks for drug name recognition task organized so far: the BioCreative IV ChemdNER task and the drugNER subtask of the SemEval-2013 DDIExtraction challenge. Section 3 describes the datasets used and the experiments performed. The experimental results are presented and discussed in Section 4. We conclude in Section 5 with a summary of our findings and some directions for future work.

## 2 State of the art

### 2.1 CHEMDNER task

The BioCreative IV CHEMDNER (Chemical compound and drug name recognition) task was devoted to NER focusing on detecting chemical entity mentions. Twenty-six teams participated in this task and as a result a corpus containing 10,000 PubMed abstracts annotated with 84,355 chemistry and chemical entity mentions was generated (Krallinger et al., 2015b). An overview of the task as well as of the main relevant characteristics of participating systems is given in (Krallinger et al., 2015a).

Participating systems used three approaches to recognize chemical entity mentions: (a) supervised machine learning techniques (used by 17 systems). CRF was the most used technique followed by Support Vector Machines (SVM) and logistic regression. These systems used different types of features: word level features (such as ngrams, numerical items and digits, word length, part-of-speech, among others), lookup features extracted from dictionaries and gazetteers and document features (for example, coocurrences of mentions); (b) rule-based approaches are used in two systems in the form of lexical patterns that implements the IUPAC nomenclature guidelines to detect formulas or specific sequences of compounds (this strategy requires a high understanding of chemical naming standards as well as annotation guidelines) and (c) dictionary-based approaches are integrated in four systems where domain-specific resources (such as CheBI[5], PubChem[6] or DrugBank[7]) and gazetteers are expanded with lexical variations to improve recall scores taking into account that a post-processing task of removing and pruning lexical entries is required. Only three

[3]http://dumps.wikimedia.org/

[4]http://www.nlm.nih.gov/databases/journal.html

[5]https://www.ebi.ac.uk/chebi/

[6]https://pubchem.ncbi.nlm.nih.gov

[7]http://www.drugbank.ca

systems tried a hybrid approach combining machine learning and rule-based strategies. Analyzing the runs submitted by participating teams, it is important to highlight that the top ranked system (Leaman et al., 2015) (87,39% of F-score) implemented a hybrid approach that combines a CRF model, a set of patterns to identify special types of mentions and gazetteers. This score is very close to the inter human annotator agreement (IAA) in this task (91%).

### 2.2 SemEval-2013 DrugNER task

The DDIExtraction Shared Task 2013 (Segura Bedmar et al., 2013; Segura-Bedmar et al., 2014) is the second edition of the DDIExtraction Shared Task series, a community-wide effort to promote the implementation and comparative assessment of NLP techniques in the field of the Pharmacovigilance domain. To attain this aim, two main tasks were proposed: the recognition of pharmacological substances (DrugNER task) and the detection and classification of drug-drug interactions (DDI task) from biomedical texts. Four types of pharmacological substances were defined: *drug* (generic drug names), *brand* (branded drug names), *group* (drug group names) and *drug-n* (active substances not approved for human use). The results of the participating systems were evaluated according to four evaluation criteria: strict (which demands exact boundary and entity type matching), exact (which only demans exact boundary matching), partial (which only demands partial boundary matching) and type (which demands partial boundary and entity type matching).

A total of 6 teams participated in the DrugNER subtask. The reader can find the full ranking information in (Segura Bedmar et al., 2013). In general, the results on the DDI-DrugBank dataset were much better than those obtained on the DDI-MedLine dataset. While DDI-DrugBank texts focus on the description of drugs and their interactions, the main topic of DDI-MedLine texts would not necessarily be on DDIs. Coupled with this, it is not always trivial to distinguish between substances that should be classified as pharmacological substances and those that should not. This is due to the ambiguity of some pharmacological terms. For example, *insulin* is a hormone produced by the pancreas, but can also be synthesized in the laboratory and used as drug to treat insulin-dependent diabetes mellitus. The participating systems should be able to determine if the text is describing a substance originated within the organism or, on the contrary, it describes a process in which the substance is used for a specific purpose and thus should be identified as pharmacological substance.

The best results were achieved by the WBI team (Rocktäschel et al., 2013) with a CRF algorithm. The system employed a domain-independent feature set along with features generated from the output of ChemSpot (Rocktäschel et al., 2012), an existing chemical named entity recognition tool, as well as a collection of domain-specific resources. Its model was trained on the training dataset as well as on entities of the test dataset for the DDI task. In the detection subtask (which only requires exact boundary matching), this system achieved an F1 of 90% on the DDI-DrugBank dataset and an F1 of 78% on DDI-MedLine. As expected, the results of the classification subtask (strict evaluation) were worse, showing an F1 of 87.8% on DDI-DrugBank and 58.1% on DDI-MedLine.

## 3 Method

This section describes the datasets and settings used in our experiments.

### 3.1 Datasets

The major contribution of DDIExtraction was to provide a benchmark corpus, the DDI corpus. The corpus was manually annotated with a total of 18,502 pharmacological substances and 5,028 DDIs. It consists of two different datasets: DDI-DrugBank (792 texts selected from the DrugBank database) and DDI-MedLine (233 MedLine abstracts on the subject of DDIs). A detailed description of the DDI corpus can be found in (Herrero-Zazo et al., 2013).

The corpus was split in order to build the datasets for the training and evaluation of the different participating systems. Approximately 77% of the DDI corpus documents were randomly selected for the training dataset and the remaining was used for the test dataset. The training dataset is the same for both subtasks since it contains entity and DDI annotations. The test dataset for the DrugNER task was formed by discarding documents which contained DDI annotations. Entity annotations were removed from this dataset to be used by participants. The remaining docu-

ments (that is, those containing some interactions) were used to create the test dataset for the DDI task. Since entity annotations are not removed from these documents, the test dataset for the DDI task can also be used as additional training data for the DrugNER task.

Table1 shows the basic statistics on the training and test datasets for the DrugNER task.

### 3.2 Experiments

As it stated in the previous section, most successful approaches for drug name recognition have used machine learning algorithms such as CRFs trained with linguistic features (tokens, lemmas or POS tags, among others) and semantic features from domain resources such as ontologies or dictionaries. Encouraged by the good results of the CRF-based methods, we propose a system based on CRF and also explore word embedding features provided by the Word2vec tool. In particular, we used a python binding[8] to CRFsuite (Okazaki, 2007).

CRF performs the NER task as a classification task on each token, determining whether it is an entity or not. To represent the class of each token, we used the BIO tagging scheme. According to this scheme, each token is tagged as either beginning entity token (B), inside entity token (I) or outside token (O). For the detection subtask (exact criterion), we only considered three classes: B-ENTITY, I-ENTITY and O. However, since we had to classify four different types (drug, brand, group and drug-n), we used nine different classes for the classification task.

As a first stage, we developed a baseline system using a CRF algorithm in which each token is represented with the following features:

- The context window of three tokens to its right and to its left in the sentence. The context window also includes the current token.
- POS tags and lemmas in the context window are also considered.
- An orthography feature which can take the following values: upperInitial (the token begins with an uppercase letter and the rest are lowercase), allCaps (all its letters are uppercase), lowerCase (all its letters are lowercase) and mixedCaps (the token contains any mixture of upper and lowercase letters).
- A feature representing the type of token: word, number, symbol or punctuation.

As one of our goals is to study the contribution of DINTO in the task, in a second stage, we also considered a binary feature that indicated whether the current token was found in the DINTO ontology.

Figure 1 shows a pipeline of GATE components used to process the texts and to obtain the feature set used to train the CRF model. There are five main processing modules: sentence splitter, tokenizer, POS tagger, morphological analyzer and the Gate onto root gazetteer, which links text to the DINTO ontology. The ontology is processed to produce a flexible gazetteer taking into account alternative morphological forms of the instances of the ontology.

The main hypothesis of this work is that the incorporating of word embeddings as features into a CRF model could help to recognize unseen or very rare drug mentions in the training set. Thus, we train word embeddings using the Word2vec tool. Word2vec only requires a large corpus of sentences as input dataset in order to generate word vectors by training a NN language model. The NN model is able to learn from the different contexts in which a word appears and then to compute its representation as a vector. In this study, Word2Vec tool was trained on two different corpora. As first option, we used the latest wikipedia dump[9], which contains more than 3 billion words. Then, we used the Word2Vec model trained on Wikipedia to obtain the word vectors for all tokens in the DDI corpus.

Based on distributional hypothesis (Harris, 1954), similar words will have similar vectors because they occur in similar contexts. The word vector for the current token was considered as a new feature into an our CRF system. We tried with different dimensions of vectors (50, 100 and 200) (see Table 3). It should be noted that these word representations could be very valuable input, not only for named entity recognition, but also in many other NLP tasks (POS tagging, word name disambiguation, lexical simplification, etc).

Another important advantage of the Word2vec tool is that contains a utility to compute word clusters using a k-means clustering algorithm. Thus, we also used word cluster as a new feature to represent the current token in our CRF-based system.

[8] http://python-crfsuite.readthedocs.org/en/latest/

[9] http://dumps.wikimedia.org/

| | | Training + Test for DDI task | Test for DrugNER task |
|---|---|---|---|
| **DDI-DrugBank** | **documents** | 730 | 54 |
| | **sentences** | 6648 | 145 |
| | **drug** | 9715 | 180 |
| | **group** | 3832 | 65 |
| | **brand** | 1770 | 53 |
| | **drug_n** | 124 | 5 |
| **DDI-MedLine** | **documents** | 175 | 58 |
| | **sentences** | 1627 | 520 |
| | **drug** | 1574 | 171 |
| | **group** | 234 | 90 |
| | **brand** | 36 | 6 |
| | **drug_n** | 520 | 115 |

Table 1: Statistics on the training and test dataset for the DrugNER task.

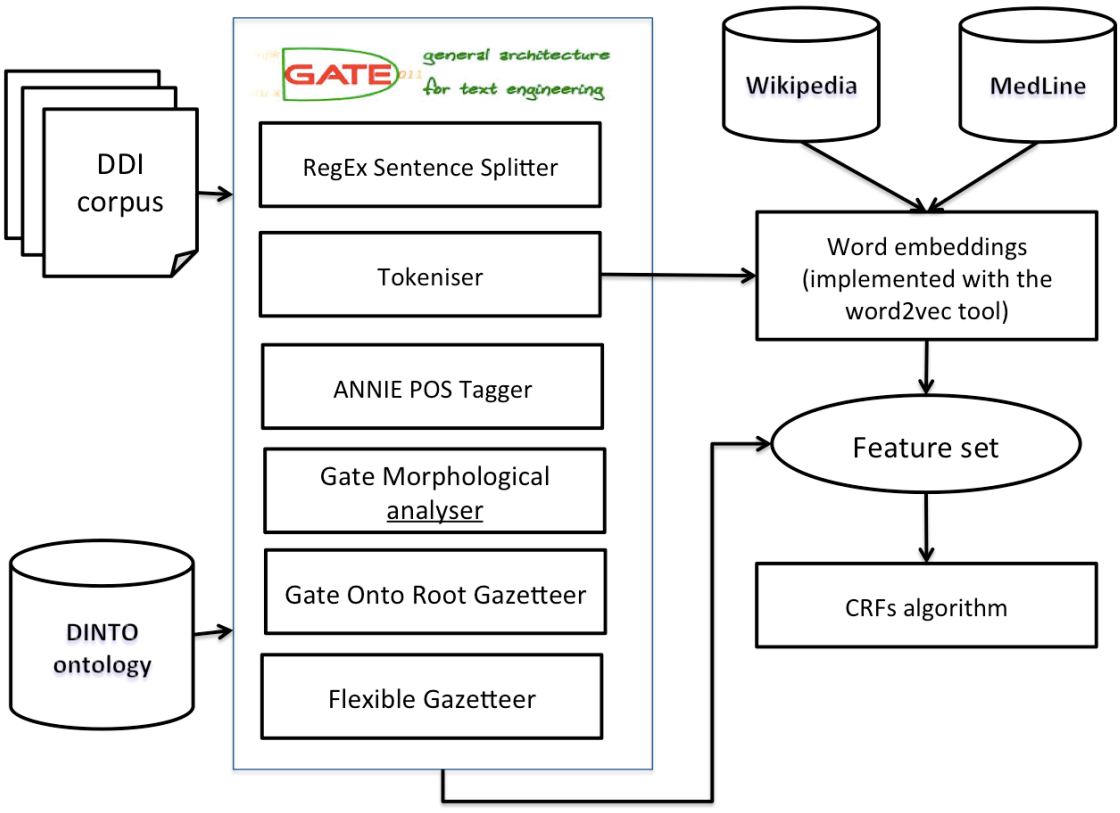


Figure 1: System architecture and pipelines for CRF machine learning-based Drug NER.

Word clusters represent words at a higher level abstraction that may help to recognize even those drug mentions that are not observed in the training set. We performed experiments for different values of k in the k-means (50, 150 and 500). All experiments are summarized in Table 2.

## 4 Evaluation

Table 3 shows the results for the different settings studied for the detection subtask (exact criterion) and for the classification subtask (strict criterion). The scores correspond to the micro-average values, which were calculated with regarding all classes (B- and I-) of each corresponding subtask.

The following subsections present and discuss the results for each dataset: DDI-DrugBank and DDI-MedLine.

### 4.1 Results on DDI-DrugBank

#### 4.1.1 Detection subtask

The use of a lexicon feature from DINTO achieved an increase in both precision and recall (and consequently, an improvement of 1% in F1 score).

The results suggest that Word2vec features can potentially lead to improved detection performance. In general, the use of word clusters showed a significant increase in recall values (from 84% to 89%), and hence a gain of 3% in F1. However, word clusters did not seem significantly to alter overall precision values. As expected, word cluster is an effective feature to improve the coverage of the system.

Our initial hypothesis was that Word2vec features trained on MedLine should provide better results because these texts are focused on the biomedical domain, however the results demonstrated that word clusters from Wikipedia, in general, had a better performance than those from MedLine. This may be due to the size of the Wikipedia corpus is significantly larger than the release of Medline used in this work. Therefore, Wikipedia is the best option to train our Word2Vec models in our current settings, though Wikipedia cover a vast array of subjects, not necessarily related to the biomedical domain.

Word cluster features trained on MedLine always seem to provide the same scores, that is, there is no difference between to use a cluster which was calculated using k=50, k=150 or k=500. Word clusters trained on Wikipedia produced better results when the number of clusters is larger. More experiments are necessary to confirm or deny these results. In general, word clusters performed better than word vectors.

To sum up, the results suggest that word clusters are the most influential features for the detection subtask, achieving an improvement of 4% in recall over the baseline system.

#### 4.1.2 Classification subtask

Regarding the results of the classification task on the DDI-DrugBank dataset, the use of Word2vec features did not necessarily give better results than the baseline system and might even be worse (see Table 3). The best F1 (75%) was obtained by five different strategies (see Table 3): baseline, word clusters (k=50) on Wikipedia, word clusters (k=50, k=500) on MedLine and word vectors (d=50) on MedLine.

Similarly, DINTO did not overcome the baseline system yet. Therefore, while the experiments on the detection task show that the use of DINTO and Word2vec features could help to improve the performance, this positive effect does not seem to be present for the classification task.

### 4.2 Results on DDI-MedLine

#### 4.2.1 Detection subtask

The use of DINTO led to an increase in precision, achieving 10% over the baseline system, and an increase of 3% in recall. Thus, F1-score went up from 61% to 66%.

Word cluster features generated from Wikipedia provided a significant improvement of 6% in recall, but with worse precision than the combination of baseline with DINTO. As was the case on DDI-DrugBank, lower improvements were obtained by the word clusters trained on MedLine. Moreover, word clusters seemed to perform better than word vectors. On the other hand, word vectors trained on MedLine showed precision values very close to those obtained by the baseline system with DINTO.

#### 4.2.2 Classification subtask

Contrary to the evaluation on the DDI-DrugBank dataset, the use of DINTO increased the baseline precision by 8% and the baseline recall by 3%. Therefore, DINTO provide valuable information for the classification of drug entities in scientific texts. This may be due to DINTO incorporates information from several resources such as the ChEBI ontology, the DrugBank database and

the ATC classification system[10] (a drug classification system developed by WHO). Word clusters (k=500) achieved the best performance by increasing the recall (by 7%) and thus the F1 accordingly. However, word vectors do not seem to provide an improvement over the results achieved by DINTO.

Although our system does not provide better performance than the WBI system, the use of the DINTO feature show a significant improvement by 9% in precision over the WBI system, but with a sharp reduction in recall.

## 5 Conclusion

The main contribution of this paper is the incorporation of word embedding features into a CRF-based NER system for drug entities. In addition, we explore if the DINTO ontology can be a valuable resource for the task.

The results suggest that DINTO can lead to improve the performance over the detection subtask. Therefore, we can confirm that the DINTO ontology is a useful resource for the drug name recognition task from scientific texts. For this reason, we intend to continue studying on how to better use DINTO in order to increase the performance of the task. Moreover, we believe that the inclusion of additional semantic features from biomedical resources (such as DrugBank, CheBI, ChemIDPlus, the ATC classification system, Drugs@FD [11], etc) are essential in order to improve performance for the classification subtask.

As we foresaw in the initial hypothesis, Word2vec features achieve a marked improvement in recall for the detection task. Word cluster features trained on Wikipedia seem to provide the most satisfactory results. More experiments are necessary to determine the optimum number of clusters for the task. Although in general our results are not better than those achieved by the top system in the DrugNER task, we strongly believe the use of word embeddings for this task is worth further research.

Our experiments conducted on the DDI corpus allow us to compare our approach with the participating systems of the DrugNER task in the SemEval-2013 DDIExtraction challenge. In general, our system does not perform better than the top system (WBI) in this shared task. However, the results for the classification task on the DDI-MedLine dataset show that DINTO could be a valuable resource to improve precision.

The WBI system provided an F1 of 87.8% on DDI-DrugBank (which is very close to the IAA (0.91)), but performed worse on the DDI-MedLine dataset (showing an F1 of 58.1%). It stands to reason that this system could have already reached the maximum threshold results for the DDI-DrugBank dataset. On the other hand, there is much room for improvement on the DDI-MedLine dataset. The results reported in (Liu et al., 2015) are better than those provided by the WBI system. However, since the authors only provide results for the whole DDI corpus, we cannot know the performance of their system on each dataset and whether their system is able to overcome the WBI system on the DDI-MedLine dataset.

In future work, we will first train the Word2vec tool using a large set of MedLine abstracts. It could provide better results than those obtained from the Word2vec model trained on Wikipedia. Since MedLine is a biomedical literature database, Medline abstracts should provide better word representations for drug entities than those obtained from Wikipedia articles. We also plan to extend the experimentation to the ChemdNER corpus in order to compare our approach to the participating systems of the BioCreative IV CHEMDNER task. We also intend to carry out an error analysis to determine the main causes for wrong detection and classification.

Furthermore, we will still explore additional word embedding features for the drugNER task. In particular, we plan to generate vectors to represent, not only words, but also phrases because many biomedical concepts are multiwords. Additionally, the parameters of CRF algorithm will be fine-tuned through cross-validation on the training set for improving the classification results on the test set.

Finally, we would like to investigate the contribution of word embeddings for the relation extraction task, especially, the extraction of DDIs. We will also explore how the DINTO ontology can be used to improve the DDI extraction task. We strongly believe that this ontology could be a valuable resource for the research on Biomedical Information Extraction and would like to encourage the research community to use the DINTO ontology, which is available for research purposes at https://code.google.com/p/dinto/.

[10] http://www.whocc.no/atc/structure_and_principles/

[11] http://www.accessdata.fda.gov/scripts/cder/drugsatfda/

## Acknowledgments

This work was supported by TrendMiner project [FP7-ICT287863] and by eGovernAbility-Access project (TIN2014-52665-C2-2-R).

## References

Kirill Degtyarenko, Paula De Matos, Marcus Ennis, Janna Hastings, Martin Zbinden, Alan McNaught, Rafael Alcántara, Michael Darsow, Mickaël Guedj, and Michael Ashburner. 2008. Chebi: a database and ontology for chemical entities of biological interest. *Nucleic acids research*, 36(suppl 1):D344–D350.

Zellig S Harris. 1954. Distributional structure. *Word*.

Yongqun He, Sirarat Sarntivijai, Yu Lin, Zuoshuang Xiang, Abra Guo, Shelley Zhang, Desikan Jagannathan, Luca Toldo, Cui Tao, and Barry Smith. 2014. Oae: the ontology of adverse events. *J Biomed Semantics*, 5:29.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The DDI corpus: An annotated corpus with pharmacological substances and drug-drug interactions. *Journal of Biomedical Informatics*, 46(5):914–920.

María Herrero Zazo. 2015. *Semantic Resources in Pharmacovigilance: A Corpus and an Ontology for Drug-Drug Interactions*. Ph.D. thesis, Carlos III University of Madrid, 5.

Martin Krallinger, Florian Leitner, Obdulia Rabal, Miguel Vazquez, Julen Oyarzabal, and Alfonso Valencia. 2015a. Chemdner: The drugs and chemical names extraction challenge. *J Cheminform*, 7(Suppl 1):S1.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, et al. 2015b. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of cheminformatics*, 7(Suppl 1):S2.

Robert Leaman, Chih-Hsuan Wei, and Zhiyong Lu. 2015. tmchem: a high performance approach for chemical named entity recognition and normalization. *Journal of cheminformatics*, 7(supplement 1).

Shengyu Liu, Buzhou Tang, Qingcai Chen, Xiaolong Wang, and Xiaoming Fan. 2015. Feature engineering for drug name recognition in biomedical texts: Feature conjunction and feature selection. *Computational and mathematical methods in medicine*, 2015.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *ICLR 2013 Workshop Track*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Naoaki Okazaki. 2007. Crfsuite: a fast implementation of conditional random fields (crfs).

T. Rocktäschel, M. Weidlich, and U. Leser. 2012. Chemspot: a hybrid system for chemical named entity recognition. *Bioinformatics*, 28(12):1633–1640.

Tim Rocktäschel, Torsten Huber, Michael Weidlich, and Ulf Leser. 2013. Wbi-ner: The impact of domain-specific features on the performance of identifying and classifying mentions of drugs. In *Proceedings of the 7th International Workshop on Semantic Evaluation*, pages 356–363.

Isabel Segura Bedmar, Paloma Martínez, and María Herrero Zazo. 2013. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts. In *SemEval-2013: Semantic Evaluation Exercises Workshop*, pages 341–350. Association for Computational Linguistics.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2014. Lessons learnt from the ddiextraction-2013 shared task. *Journal of biomedical informatics*, 51:152–164.

Richard Socher, John Bauer, Christopher D Manning, and Andrew Y Ng. 2013a. Parsing with compositional vector grammars. In *In Proceedings of the ACL conference*. Citeseer.

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013b. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, volume 1631, page 1642. Citeseer.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics.

David S Wishart, Craig Knox, An Chi Guo, Savita Shrivastava, Murtaza Hassanali, Paul Stothard, Zhan Chang, and Jennifer Woolsey. 2006. Drugbank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic acids research*, 34(suppl 1):D668–D672.

| System | Feature set |
|---|---|
| CRF | standard feature set |
| CRFD | baseline + DINTO feature |
| CRFclusterK50Wiki | CRFD's features + word cluster from Word2Vec trained with k=50 on Wikipedia |
| CRFclusterK50MedLine | CRFD's features + word cluster from Word2Vec trained with k=50 on MedLine |
| CRFclusterK150Wiki | CRFD's features + word cluster from Word2Vec trained with k=150 on Wikipedia |
| CRFclusterK150MedLine | CRFD's features + word cluster from Word2Vec trained with k=150 on MedLine |
| CRFclusterK500Wiki | CRFD's features + word cluster from Word2Vec trained with k=500 on Wikipedia |
| CRFclusterK50MedLine | CRFD's features + word cluster from Word2Vec trained with k=500 on MedLine |
| CRFvec50Wiki | CRFD's features + word vectors of dimension 50 from Word2Vec trained on Wikipedia |
| CRFvec50MedLine | CRFD's features + word vectors of dimension 50 from Word2Vec trained on MedLine |
| CRFvec100Wiki | CRFD's features + word vectors of dimension 100 from Word2Vec trained on Wikipedia |
| CRFvec100MedLine | CRFD's features + word vectors of dimension 100 from Word2Vec trained on MedLine |
| CRFvec200Wiki | CRFD's features + word vectors of dimension 200 from Word2Vec trained on Wikipedia |
| CRFvec200MedLine | CRFD's features + word vectors of dimension 200 from Word2Vec trained on MedLine |

Table 2: List of experiments.

| | | Exact criterion | | | Strict criterion | | |
|---|---|---|---|---|---|---|---|
| | | **P** | **R** | **F1** | **P** | **R** | **F1** |
| **DDI-DrugBank** | **WBI** | 0.90 | 0.89 | 0.90 | 0.88 | 0.87 | 0.87 |
| | **CRF** | 0.70 | 0.85 | 0.77 | 0.69 | 0.82 | **0.75** |
| | **CRFD** | 0.72 | 0.84 | 0.77 | 0.68 | 0.81 | 0.74 |
| | **CRFclusterK50Wiki** | 0.72 | 0.89 | 0.79 | 0.68 | 0.83 | **0.75** |
| | **CRFclusterK150Wiki** | 0.73 | 0.89 | **0.80** | 0.68 | 0.83 | 0.74 |
| | **CRFclusterK500Wiki** | 0.72 | 0.89 | **0.80** | 0.68 | 0.83 | 0.74 |
| | **CRFclusterK50MedLine** | 0.72 | 0.86 | 0.79 | 0.69 | 0.82 | **0.75** |
| | **CRFclusterK150MedLine** | 0.72 | 0.86 | 0.79 | 0.68 | 0.82 | 0.74 |
| | **CRFclusterK500MedLine** | 0.72 | 0.86 | 0.79 | 0.69 | 0.82 | **0.75** |
| | **CRFvec50Wiki** | 0.71 | 0.84 | 0.77 | 0.69 | 0.81 | 0.74 |
| | **CRFvec100Wiki** | 0.72 | 0.84 | 0.77 | 0.69 | 0.81 | 0.74 |
| | **CRFvec200Wiki** | 0.72 | 0.85 | 0.78 | 0.68 | 0.80 | 0.74 |
| | **CRFvec50MedLine** | 0.72 | 0.84 | 0.78 | 0.69 | 0.82 | **0.75** |
| | **CRFvec100MedLine** | 0.73 | 0.86 | 0.79 | 0.68 | 0.81 | 0.74 |
| | **CRFvec200MedLine** | 0.73 | 0.85 | 0.79 | 0.68 | 0.80 | 0.74 |
| **DDI-MedLine** | **WBI** | 0.81 | 0.74 | 0.77 | 0.61 | 0.56 | 0.58 |
| | **CRF** | 0.69 | 0.54 | 0.61 | 0.62 | 0.44 | 0.52 |
| | **CRFD** | 0.79 | 0.57 | 0.66 | 0.70 | 0.47 | 0.56 |
| | **CRFclusterK50Wiki** | 0.74 | 0.63 | **0.68** | 0.66 | 0.48 | 0.56 |
| | **CRFclusterK150Wiki** | 0.73 | 0.63 | **0.68** | 0.67 | 0.49 | **0.57** |
| | **CRFclusterK500Wiki** | 0.72 | 0.64 | **0.68** | 0.65 | 0.51 | **0.57** |
| | **CRFclusterK50MedLine** | 0.74 | 0.59 | 0.66 | 0.64 | 0.46 | 0.53 |
| | **CRFclusterK150MedLine** | 0.75 | 0.63 | **0.68** | 0.66 | 0.49 | 0.56 |
| | **CRFclusterK500MedLine** | 0.73 | 0.62 | 0.67 | 0.67 | 0.49 | **0.57** |
| | **CRFvec50Wiki** | 0.77 | 0.57 | 0.66 | 0.68 | 0.47 | 0.56 |
| | **CRFvec100Wiki** | 0.78 | 0.56 | 0.66 | 0.66 | 0.46 | 0.54 |
| | **CRFvec200Wiki** | 0.77 | 0.57 | 0.66 | 0.68 | 0.46 | 0.55 |
| | **CRFvec50MedLine** | 0.79 | 0.57 | 0.66 | 0.66 | 0.45 | 0.54 |
| | **CRFvec100MedLine** | 0.81 | 0.57 | 0.66 | 0.69 | 0.46 | 0.55 |
| | **CRFvec200MedLine** | 0.78 | 0.57 | 0.66 | 0.68 | 0.46 | 0.55 |

Table 3: Experimental results.