WBI-NER: The impact of domain-specific features on the performance of identifying and classifying mentions of drugs

Tim Rocktäschel **Torsten Huber** **Michael Weidlich** **Ulf Leser**
Humboldt-Universität zu Berlin
Knowledge Management in Bioinformatics
Unter den Linden 6
Berlin, 10099, Germany
{trocktae,thuber,weidlich,leser}@informatik.hu-berlin.de

Abstract

Named entity recognition (NER) systems are often based on machine learning techniques to reduce the labor-intensive development of hand-crafted extraction rules and domain-dependent dictionaries. Nevertheless, time-consuming feature engineering is often needed to achieve state-of-the-art performance. In this study, we investigate the impact of such domain-specific features on the performance of recognizing and classifying mentions of pharmacological substances. We compare the performance of a system based on general features, which have been successfully applied to a wide range of NER tasks, with a system that additionally uses features generated from the output of an existing chemical NER tool and a collection of domain-specific resources. We demonstrate that acceptable results can be achieved with the former system. Still, our experiments show that using domain-specific features outperforms this general approach. Our system ranked first in the SemEval-2013 Task 9.1: Recognition and classification of pharmacological substances.

1 Introduction

The accurate identification of drug mentions in text is an important prerequisite for many applications, including the retrieval of information about substances in drug development (*e.g.* Roberts and Hayes (2008)), the identification of adverse drug effects (*e.g.* Leaman et al. (2010)) and the recognition of drug-drug interactions (*e.g.* Thomas et al. (2011)). Given that most of the information related to drug research is covered by medical reports and pharmacological publications, computational methods for information extraction should be used to support this task.

The SemEval-2013 Task 9.1 competition[1] (Segura-Bedmar et al., 2013) aims at a fair assessment on the state-of-the-art of tools that recognize and classify mentions of pharmacological substances in natural language texts – a task referred to as drug named entity recognition (NER). The goal of participating teams is to recreate the gold annotation on a held-out part of an annotated corpus. Four classes of entities have to be identified: `Drug`, `DrugN`, `Group` and `Brand`. Entities of class `Drug` denote any kind of drug that is approved for use in humans, whereas `DrugN` denotes substances that are not approved. `Group` are terms describing a group of drugs and `Brand` stands for drug names introduced by a pharmaceutical company.

The aim of this study is to examine whether it is worthwhile to implement domain-specific features for supporting drug NER. The question we attempt to answer is whether such features really help in identifying and classifying mentions of drugs or whether a mostly domain-independent feature set, which can be applied to many other tasks, achieves a comparable performance.

2 Related work

Various NER systems for identifying different classes of chemical entities, including mentions of drugs, trivial names and IUPAC terms, have been proposed.

[1] `http://www.cs.york.ac.uk/semeval-2013/task9/` (accessed 2013-04-29)

*Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 356–363, Atlanta, Georgia, June 14-15, 2013.

Klinger et al. (2008) trained a conditional random field (CRF) (Lafferty et al., 2001) for extracting mentions of IUPAC and IUPAC-like entities. They report an F_1 measure of 85.6% on a hand-annotated corpus consisting of MEDLINE abstracts.

Segura-Bedmar et al. (2008) introduced Drug-NER, which is based on UMLS MetaMap Transfer (MMTx) and nomenclature rules by the World Health Organization International Nonproprietary Names (INNs). Their system extracts and classifies mentions of drugs and achieves a precision of 99.1% and a recall of 99.8% on a silver-standard corpus.

OSCAR (Open-Source Chemistry Analysis Routines) (Corbett and Murray-Rust, 2006; Jessop et al., 2011) extracts mentions of a wide range of chemicals using a maximum entropy Markov model (McCallum et al., 2000). It achieves an F_1 of 83.2% on a corpus consisting of PubMed abstracts and 80.7% on a corpus consisting of chemistry papers (Corbett and Copestake, 2008).

Hettne et al. (2009) compiled Jochem (the joint chemical dictionary) from ChemIDplus, ChEBI, DrugBank, PubChem, HMDB, KEGG, MeSH and CAS Registry IDs. Jochem was used with Peregrine (Schuemie et al., 2007), a dictionary-based NER tool, achieving an F_1 of 50% on the SCAI corpus (Kolárik et al., 2008).

We developed ChemSpot (Rocktäschel et al., 2012), a system for extracting mentions of various kinds of chemicals from text. We applied a CRF for extracting mentions of IUPAC entities based on the work of Klinger et al. (2008) and used Jochem (Hettne et al., 2009) with an adapted matching-mechanism for identifying trivial names, drugs and brands. ChemSpot v1.0 achieved an overall F_1 of 68.1% on the SCAI corpus. In the meantime, we have worked on several enhancements (see Section 3.1).

The SemEval-2013 Task 9.1 poses new challenges on NER tools. Instead of targeting all kinds of chemicals, it focuses on drugs, *i.e.*, pharmacological substances that affect humans and are used for administration. Moreover, entities need to be classified into the four categories mentioned above.

3 Methods

Our approach is based on a linear-chain CRF with mostly domain-independent features commonly applied to NER tasks. In addition, we employ various domain-specific features derived from the output of ChemSpot's components, as well as Jochem, the PHARE ontology (Coulet et al., 2011) and the ChEBI ontology (De Matos et al., 2010). In the following, we first explain extensions to ChemSpot. Subsequently, we give a brief introduction to linear-chain CRFs before describing the general and domain-specific features used by our system. Finally, we explain the experimental setup and discuss our results.

3.1 Improvements of ChemSpot

To improve ChemSpot's chemical NER performance, we extend it by two components and modify its match-expansion mechanism.

The first addition is a pattern-based tagger for chemical formulae. In its basic form it extracts mentions matching the regular expression `(S N?(\+|-)?)+` where `S` denotes a chemical symbol and `N` a natural number greater one.[2] This pattern is augmented by filters to comply with other naming conventions, such as correct grouping of compounds with parentheses.

The second extension targets ambiguous abbreviations. For example, the abbreviation "DAG" could denote "diacylglycerol" or "directed acyclic graph". We use ABBREV, an algorithm proposed by Schwartz and Hearst (2003), for extracting such abbreviations and their definitions (*e.g.* "diacylglycerol (DAG)"). Note that the position of the long form (LF) and short form (SF) is interchangeable. To disambiguate between chemical and non-chemical abbreviations, we apply the following two rules to the mentions extracted by ChemSpot: (1) For a given pair of LF and SF, we check whether the LF was found to be a chemical but the SF was not. In this case we add a new annotation for every occurrence of the SF in the document. (2) Contrary to that, if only the SF was tagged as a chemical but the LF was not, we assume that the abbreviation does not refer to a chemical and remove all annotations of the SF in the document.

ChemSpot's match-expansion often leads to the extraction of non-chemical suffixes corresponding to verbs, *e.g.*, "-induced", "-enriched" or "-mediated".

[2] By convention, 1 is ommited (*e.g.* CO_2 instead of C_1O_2).

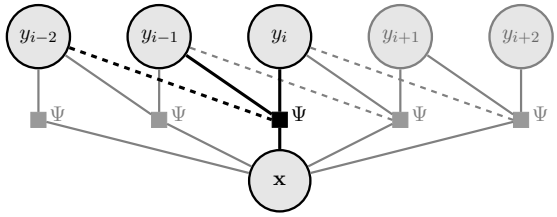

Figure 1: A factor graph for a 1st-order linear-chain CRF (2nd-order with dashed edges). Note that for each feature function f_j in the factor Ψ, the same weight θ_j is used at all other positions (gray) in the sequence (parameter tying).

To tackle this issue we stop the expansion at tokens whose part-of-speech tag refers to a verb form. Furthermore, we integrated OPSIN (Lowe et al., 2011) to normalize entity mentions to InChI strings.

The current v1.5 release of ChemSpot achieves an overall F_1 of 74.2% on the SCAI corpus, improving the performance by 6.1 percentage points (pp) F1 compared to ChemSpot v1.0.

3.2 Linear-chain conditional random fields

Contrary to the hybrid strategy used in ChemSpot, we follow a purely machine learning based approach for drug NER in this work. NER can be formulated as a sequence labeling task where the goal is to find a sequence of labels $\mathbf{y} = \{y_1, \ldots, y_n\}$ given a sequence $\mathbf{x} = \{x_1, \ldots, x_n\}$ of observed input tokens. Labels commonly follow the `IOB` format, where `B` denotes a token at the beginning of an entity mention, `I` denotes the continuation of a mention and `O` corresponds to tokens that are not part of a mention. Extracting entity mentions from a tokenized text $\mathbf{x}$ then amounts to finding $\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x})$.

Linear-chain CRFs are well-known discriminative undirected graphical models that encode the conditional probability $p(\mathbf{y} \mid \mathbf{x})$ of a *set* of input variables $\mathbf{x}$ and a *sequence* of output variables $\mathbf{y}$ (see Wallach (2004) or Klinger and Tomanek (2007) for an introduction). In the case of NER, $\mathbf{x}$ is a *sequence* of n tokens and $\mathbf{y}$ a sequence of n corresponding labels. Linear-chain CRFs of order k factorize $p(\mathbf{y} \mid \mathbf{x})$ into a product of factors Ψ, globally normalized by an input-dependent partition function $Z(\mathbf{x})$:

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(x)} \prod_{i=1}^{n} \Psi(y_{i-k}, \ldots, y_{i-1}, y_i, \mathbf{x}, i).$$

A factor Ψ is commonly defined as the exponential function of a sum of weighted feature functions $\{f_1, \ldots, f_m\}$:

$$\frac{1}{Z(\mathbf{x})} \prod_{i=1}^{n} \exp\left(\sum_{j=1}^{m} \theta_j f_j(y_{i-k}, \ldots, y_{i-1}, y_i, \mathbf{x}, i) \right).$$

The feature function weights $\{\theta_j\}$ can be learned from training data and are shared across all positions i (*parameter tying*).

The factorization of a CRF can be illustrated by a factor graph (Kschischang et al., 2001). A factor graph is a bipartite graph, where one set of nodes corresponds to random variables and the other to factors. Each factor is connected to the variables of its domain, making the factorization of the model apparent. Figure 1 shows a factor graph for a segment of a linear-chain CRF of order one and two respectively. In a linear-chain CRF of order k, the label of a token at position i is connected via feature functions of a factor to the input sequence $\mathbf{x}$ as well as the previous k labels. For example, in a first-order linear-chain CRF, one of the feature functions could be $f_{[\texttt{O}\rightarrow\texttt{B},\text{capital}]}(y_{i-1}, y_i, \mathbf{x}, i)$, which evaluates to 1 if $y_{i-1} = \texttt{O}$, $y_i = \texttt{B}$ and $\mathbf{x}_i$ starts with a capital letter (otherwise it yields 0). Multiplication with the weight $\theta_{[\texttt{O}\rightarrow\texttt{B},\text{capital}]}$ yields an unnormalized local score that indicates how favorable a transition from `O` to `B` is provided that the token at position i starts with a capital letter. Note that the terms *feature function* and *feature* are often used synonymously and the case differentiation for different labels is implicit. In this work a feature f_{capital} denotes its corresponding set of first-order feature functions $\{f_{[\text{s}\rightarrow\text{t},\text{capital}]}(y_{i-1}, y_i, \mathbf{x}, i) \mid (s,t) \in \{(\texttt{B},\texttt{B}), (\texttt{B},\texttt{I}), (\texttt{B},\texttt{O}), (\texttt{I},\texttt{B}), (\texttt{I},\texttt{I}), (\texttt{I},\texttt{O}), (\texttt{O},\texttt{B}), (\texttt{O},\texttt{O})\}\}$[3] (similarly for second-order).

We employ MALLET (McCallum, 2002) as underlying CRF implementation and use a second-order linear-chain CRF with offset conjunctions of order two. Offset conjunctions of order k adds features around a window of k to the features at a particular position, providing more contextual information. Akin to Klinger et al. (2008), we perform fine-grained tokenization, splitting at special characters and transitions from alphanumeric characters to digits. An exemplary tagging sequence is shown in Table 1.

[3] Transitions from `O` to `I` are invalid.

i	0	1	2	3	4	5	6	7	8	9	10
y	O	B-DrugN	O	B-DrugN	I-DrugN	I-DrugN	O	B-Group	O	O	O
x	Both	ibogaine	and	18	-	MC	ameliorate	opioid	withdrawal	signs	.

Table 1: Example label sequence for the tokenized sentence MedLine.d110.s4 of the training corpus.

	Feature Class	Description
F_C	f_{CHEMSPOT}	part of a prediction by ChemSpot
	f_{IUPAC}	part of an IUPAC entity
	f_{FORMULA}	part of a chemical formula
	$f_{\text{DICTIONARY}}$	part of a dictionary match
	f_{ABBREV}	part of a chemical abbreviation
F_J	f_{JOCHEM}	dictionaries in Jochem
	f_{PREFIX}	frequent chemical prefix
	f_{SUFFIX}	frequent chemical suffix
F_O	f_{PHARE}	PHARE ontology
	$f_{\text{CHEBDESCS}}$	#descendants in ChEBI ontology
	$f_{\text{CHEBDEPTH}}$	average depth in ChEBI ontology
F_G	f_{KLINGER}	see Klinger et al. (2008)
	f_{BANNER}	see Leaman and Gonzalez (2008)
	f_{ABNER}	see Settles (2005)
F_F	$f_{\text{UPPERCASESENT.}}$	part of an upper-case sentence
	$f_{\text{PREVWINDOW}}$	text of preceding four tokens
	$f_{\text{NEXTWINDOW}}$	text of succeeding four tokens

Table 2: Overview of features used for identifying and classifying mentions of pharmacological substances.

Note that the sequence-labeling approach in the described form cannot cope with discontinuous entity mentions. Since only a tiny fraction ($\approx 0.3\%$) of entities in the training corpus are discontinuous, we simply neglect these for training and tagging.

3.3 Feature sets

An overview of the features used by our system is shown in Table 2. Our first two submissions for the SemEval-2013 Task 9.1 differ only in that they use different subsets of these features. Run 1 employs a feature set assembled from common general features used for biomedical NER ($F_G \cup F_F$), whereas Run 2 additionally uses features tailored for extracting mentions of chemicals ($F_C \cup F_J \cup F_O$).

3.3.1 Run 1: general features F_G and F_F

We employ a union of common, rather domain-independent features published by Klinger et al. (2008), Settles (2005) and Leaman and Gonzalez (2008). Note that these feature sets have been successfully applied to a wide range of different biomedical NER tasks, *e.g.*, identifying mentions of DNA sequences, genes, diseases, mutations, IUPAC terms, cell lines and cell types. They encompass morphological, syntactic and orthographic features, such as the text of the token itself, token character n-grams of length 2 and 3, prefixes and suffixes of length 2, 3, and 4, characters left and right to a token and part-of-speech tags. Furthermore, they contain various regular expressions that capture, for instance, whether a token starts with a capital letter or contains digits.

In addition, we employ features based on NER examples in FACTORIE (McCallum et al., 2009). Specifically, we use the text of the four preceding and succeeding tokens and whether a token is part of a sentence that contains only upper-case characters. The latter is commonly the case for headlines, which likely contain an entity mention.

3.3.2 Run 2: domain-specific features

In addition to the features of Run 1, we use predictions of our improved version of ChemSpot, as well as features derived from Jochem, PHARE and ChEBI.

ChemSpot-based features F_C: When a token is part of a mention extracted by one of ChemSpot's components (*i.e.* IUPAC entity, chemical formula, dictionary match or chemical abbreviation), we use the name of the respective component as feature. In addition, we determine whether a token is part of an entity predicted by ChemSpot after match-expansion, boundary-correction and resolution of overlapping entities. Using the output of ChemSpot as features for our system could be framed as *stacking* (see Wolpert (1992)).

Jochem-based features F_J: For every dictionary contained in Jochem, we check whether a token is part of an entity in that dictionary and use the name of the dictionary as feature. Furthermore, we compile

a list of frequent chemical suffixes and prefixes of length three from Jochem.

Ontology-based features F_O: It is often hard to determine whether a mention refers to a specific chemical entity or rather an abstract term denoting a group of chemicals. To distinguish between these two cases, we calculate the average depth and the number of descendants of a term in the ChEBI ontology and use the binned count as feature. The idea behind these features is that the specificity of an entity correlates positively with its depth in the ontology (*e.g.* leaf nodes are likely specific chemicals) and negatively with the number of descendants (*i.e.* having few descendants indicates a specific entity).

Further ontology-based features are derived from PHARE, which consists of 200 curated relations. If possible, we map a token to a term in that ontology and use its label as feature.

3.4 Experiments

We perform document-level 10-fold cross-validation (CV) on the training corpus to measure the impact of domain-specific features. To ensure comparability between Run 1 and Run 2, we use the same splits for evaluation. Furthermore, we train models on the complete training corpus and evaluate on the test corpus of DDI Task 9.1 for each run respectively. In addition, we train a third model based on the best feature set determined with CV and use the entity mentions of the Task 9.2 test corpus, which also contains annotations of drug-drug interactions, as additional training data (Run 3). Following the SemEval-2013 Task 9.1 metrics, we evaluate *exact* matching performance (correct entity boundaries) and *strict* matching performance (correct boundaries and correct type).

4 Results

Table 3 shows micro-average CV results for identifying and classifying mentions of pharmacological substances in the training corpus. The performance varies drastically between different entity classes regardless of the feature set, *e.g.*, Run 1 achieves an F_1 of 91.0% for `Drug`, but only 15.9% F_1 for `DrugN`.

Run 2 outperforms Run 1 for entities of class `Drug` (+1.2 pp F_1) and `DrugN` (+4.9 pp F_1), but yields a lower performance for `Brand` (−0.9 pp F_1) and no change for `Group` entities. Overall, the micro-average F_1 measure increases by 0.8 pp for strict matching and 1.4 pp F_1 for exact matching.

	Run 1			Run 2			
	P	R	F_1	P	R	F_1	ΔF_1
`Drug`	92.1	89.9	91.0	92.0	92.3	92.2	+1.2
`DrugN`	54.7	9.3	15.9	62.4	12.5	20.8	+4.9
`Group`	87.2	82.5	84.8	87.3	82.3	84.8	0.0
`Brand`	87.8	70.8	78.4	87.1	69.8	77.5	−0.9
Exact	93.9	86.9	90.3	94.5	89.0	91.7	+1.4
Strict	90.3	83.6	86.8	90.3	85.1	87.6	+0.8

Table 3: Document-level 10-fold cross-validation micro-average results on the training corpus.

The performance on the test corpus (see Table 4) is drastically lower compared to CV results (*e.g.* 17.6 pp F_1 for strict evaluation of Run 1). Except for entities of class `Group`, using domain-specific features leads to a superior performance for identifying and classifying mentions of pharmacological substances. Run 2 outperforms Run 1 by 1.6 pp F_1 for strict evaluation and 5.9 pp F_1 for exact evaluation. Using entity mentions of the Task 9.2 test corpus as additional training data (Run 3) further boosts the performance by 0.7 pp F_1 for strict evaluation.

5 Discussion

Our results show a clear performance advantage when using domain-specific features tailored for identifying mentions of chemicals. CV results and results on the test corpus show an increase in precision and recall for exact matching and an increase in recall for strict matching. The considerably higher recall for exact matching can be attributed to a higher coverage of chemical entities by features that exploit domain-knowledge.

It is striking that the performance for `DrugN` entities is extremely low compared to the other classes. We believe that this might be due to two reasons. First, entities of this class are underrepresented in the training corpus ($\approx 3\%$). Since machine learning based methods tend to favor the majority class, it is likely that many `DrugN` entities were classified as mentions of one of the much larger classes `Drug` ($\approx 64\%$) or `Group` ($\approx 23\%$). This can be confirmed by the large differences between strict and exact matching results shown in Table 3 and Table 4.

		Run 1			Run 2				Run 3			
	#	P	R	F_1	P	R	F_1	ΔF_1	P	R	F_1	ΔF_1
`Drug`	351	74.2	79.5	76.8	72.9	85.2	78.6	+1.8	73.6	85.2	79.0	+0.4
`DrugN`	121	25.0	4.1	7.1	35.7	8.3	13.4	+6.3	31.4	9.1	14.1	+0.7
`Group`	155	77.3	74.8	76.1	78.1	73.5	75.7	−0.4	79.2	76.1	77.6	+1.9
`Brand`	59	76.2	81.4	78.7	77.8	83.1	80.3	+1.6	81.0	86.4	83.6	+3.3
Exact	686	82.1	72.9	77.2	85.6	80.8	83.1	+5.9	85.5	81.3	83.3	+0.2
Strict	686	73.6	65.3	69.2	73.0	68.8	70.8	+1.6	73.4	69.8	71.5	+0.7
DrugBank (Strict)	304	86.9	85.2	86.0	87.3	86.2	86.8	+0.8	88.1	87.5	87.8	+1.0
MEDLINE (Strict)	382	60.8	49.5	54.5	60.5	55.0	57.6	+3.1	60.7	55.8	58.1	+0.5

Table 4: Results on the test corpus. ΔF_1 denotes the F_1 pp difference to the preceding Run and # the number of annotated mentions

Second, `DrugN` denotes substances that have an effect on humans, but are not approved for medical use – a property that is rarely stated along with the entity mention and can thus often only be determined with domain-knowledge.

We think it is also important to point to the large difference between results obtained by 10-fold CV on the training corpus and test results (*e.g.* up to 17.6 pp F_1 for Run 1). One reason might be the large fraction ($\approx 83\%$) of entity mentions that appear more than once in the training corpus compared to presumably many unseen entities in the test corpus. For 10-fold CV this means that an entity in the evaluation fold has already been seen with a high probability in one of the nine training folds, yielding results that overestimate the generalization performance. Moreover, our results indicate that identifying and classifying pharmacological substances is much harder for MEDLINE documents than for DrugBank documents with a difference of up to 31.5 pp F_1 (*cf.* the last two rows of Table 4). Hence, another apparent reason for the performance differences is the substantial skew in the ratio of DrugBank to MEDLINE documents in the training corpus (roughly 4:1) compared to the test corpus (roughly 1:1). Since both sets of documents stem from different resources, this can be referred to as domain-adaptation problem.

In additional experiments we found that the general-purpose chemical NER tool ChemSpot achieves an F_1 of 65.5% for exact matching on the test corpus. This is 17.8 pp F_1 below our best results obtained with a machine learning based system (*cf.* Run 3) that is able to exploit properties of the task-specific annotations of the corpora.

6 Conclusion

We described our contribution to the SemEval-2013 Task 9.1: Recognition and classification of pharmacological substances. We found that a system based on rather general features commonly used for a wide range of biomedical NER tasks yields competetive results. Implementing this system needed no domain-adaptation and its performance could be sufficient for applications building upon drug NER. Nevertheless, adding domain-specific features boosts the performance considerably. Further improvements can be achieved by using entity annotations of the Task 9.2 test corpus as additional training data.

We identified two limitations of our approach. First, we found that entities of the minority class (`DrugN`) are very hard to classify correctly. Second, differences between DrugBank and MEDLINE documents probably cause a domain-adaptation problem. For future work, one could investigate whether the latter can be addressed by domain-adaptation techniques (*e.g.* Satpal and Sarawagi (2007)). To cope with `DrugN` entities, one could implement features derived from those resources that were used by the annotators for deciding whether a substance is approved for use in humans, *e.g.*, Drugs@FDA[4] and the WHO ATC[5] classification system.

[4] `http://www.accessdata.fda.gov/scripts/cder/drugsatfda/` (accessed 2013-04-29)

[5] `http://www.whocc.no/atc_ddd_index/` (accessed 2013-04-29)

Acknowledgements

We thank Philippe Thomas for preparing a simplified format of the corpora. We thank him and Roman Klinger for fruitful discussions.

Funding: Tim Rocktäschel is funded by the German Federal Ministry of Economics and Technology (BMWi) [KF2205209MS2], Torsten Huber and Michael Weidlich are funded by the German Federal Ministry of Education and Research (BMBF) [0315746].

References

Peter Corbett and Ann Copestake. 2008. Cascaded classifiers for confidence-based chemical named entity recognition. *BMC Bioinf.*, 9(Suppl 11):S4.

Peter Corbett and Peter Murray-Rust. 2006. High-throughput identification of chemistry in life science texts. In *Proc. of CompLife 2006*, pages 107–118.

Adrien Coulet, Yael Garten, Michel Dumontier, Russ B. Altman, Mark A. Musen, and Nigam H. Shah. 2011. Integration and publication of heterogeneous text-mined relationships on the Semantic Web. *J. Biomed. Semantics*, 2(Suppl 2):S10.

Paula De Matos, Rafael Alcántara, Adriano Dekker, Marcus Ennis, Janna Hastings, Kenneth Haug, Inmaculada Spiteri, Steve Turner, and Christoph Steinbeck. 2010. Chemical Entities of Biological Interest: an update. *Nucleic Acids Res.*, 38:D249–D254.

Kristina M. Hettne, Rob H. Stierum, Martijn J. Schuemie, Peter J.M. Hendriksen, Bob J.A. Schijvenaars, Erik M. Van Mulligen, Jos Kleinjans, and Jan A. Kors. 2009. A dictionary to identify small molecules and drugs in free text. *Bioinformatics*, 25(22):2983–2991.

David M. Jessop, Sam E. Adams, Egon L. Willighagen, Lezan Hawizy, and Peter Murray-Rust. 2011. OSCAR4: a flexible architecture for chemical text-mining. *J. Cheminf.*, 3(1):41.

Roman Klinger, Corinna Kolárik, Juliane Fluck, Martin Hofmann-Apitius, and Christoph M. Friedrich. 2008. Detection of IUPAC and IUPAC-like chemical names. In *Proc. of ISMB. Bioinformatics*, volume 24, pages i268–i276.

Roman Klinger and Katrin Tomanek. 2007. Classical Probabilistic Models and Conditional Random Fields. *Algorithm Engineering Report TR07-2-013*. Department of Computer Science, Dortmund University of Technology. ISSN 1864-4503.

Corinna Kolárik, Roman Klinger, Christoph M. Friedrich, Martin Hofmann-Apitius, and Juliane Fluck. 2008. Chemical names: terminological resources and corpora annotation. In *Proc. of the Workshop on Building and evaluating resources for biomedical text mining*, pages 51–58.

Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger. 2001. Factor graphs and the sum-product algorithm. *IEEE Trans. Inf. Theory*, 47(2):498–519.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proc. of ICML-2001*, pages 282–289.

Robert Leaman and Graciela Gonzalez. 2008. BANNER: an executable survey of advances in biomedical named entity recognition. In *Proc. of Pac Symp Biocomput*, pages 652–663.

Robert Leaman, Laura Wojtulewicz, Ryan Sullivan, Annie Skariah, Jian Yang, and Graciela Gonzalez. 2010. Towards internet-age pharmacovigilance: extracting adverse drug reactions from user posts to health-related social networks. In *Proc. of Workshop BioNLP*, pages 117–125. ACL.

Daniel M. Lowe, Peter T. Corbett, Peter Murray-Rust, and Robert C. Glen. 2011. Chemical name to structure: Opsin, an open source solution. *J. Chem. Inf. Model.*, 51(3):739–753.

Andrew K. McCallum. 2002. MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu.

Andrew K. McCallum, Dayne Freitag, and Fernando. Pereira. 2000. Maximum entropy Markov models for information extraction and segmentation. In *Proc. of ICML-2000*, pages 591–598.

Andrew K. McCallum, Karl Schultz, and Sameer Singh. 2009. FACTORIE: Probabilistic programming via imperatively defined factor graphs. In *Proc. of Neural Information Processing Systems (NIPS)*.

Phoebe M. Roberts and William S. Hayes. 2008. Information needs and the role of text mining in drug development. In *Proc. of Pac Symp Biocomput*, pages 592–603.

Tim Rocktäschel, Michael Weidlich, and Ulf Leser. 2012. ChemSpot: A Hybrid System for Chemical Named Entity Recognition. *Bioinformatics*, 28(12):1633–1640.

Sandeepkumar Satpal and Sunita Sarawagi. 2007. Domain adaptation of conditional probability models via feature subsetting. In *Knowledge Discovery in Databases: PKDD 2007*, pages 224–235. Springer.

Martijn J. Schuemie, Rob Jelier, and Jan A. Kors. 2007. Peregrine: Lightweight gene name normalization by dictionary lookup. In *Proc. of the Second BioCreative Challenge*, volume 2, pages 131–133.

Ariel S. Schwartz and Marti A. Hearst. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Proc. of Pac Symp Biocomput*, volume 8, pages 451–462.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts. In

Proc. of the 7th International Workshop on Semantic Evaluation (SemEval 2013).

Isabel Segura-Bedmar, Paloma Martínez, and María Segura-Bedmar. 2008. Drug name recognition and classification in biomedical texts. A case study outlining approaches underpinning automated systems. *Drug Discovery Today*, 13(17-18):816–823.

Burr Settles. 2005. ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text. *Bioinformatics*, 21(14):3191–3192.

Philippe Thomas, Mariana Neves, Illés Solt, Domonkos Tikk, and Ulf Leser. 2011. Relation extraction for drug-drug interactions using ensemble learning. In *Proc. of the 1st Challenge Task on Drug-Drug Interaction Extraction 2011*, pages 11–18.

Hanna M. Wallach. 2004. Conditional Random Fields: An Introduction. *Technical Report MS-CIS-04-21*. Department of Computer and Information Science, University of Pennsylvania.

David H. Wolpert. 1992. Stacked generalization. *Neural Networks*, 5(2):241–259.