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 competition1 (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.

1http://www.cs.york.ac.uk/semeval-2013/task9/ (accessed 2013-04-29)
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 F₁ measure of 85.6% on a hand-annotated corpus consisting of MEDLINE abstracts.

Segura-Bedmar et al. (2008) introduced DrugNER, which is based on UMLS MetaMap Transfer (MIMT) 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 F₁ 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 F₁ of 50% on the SCAI corpus (Kolárík 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 F₁ 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 \( \text{(S N? (\+|-)\+)?} \) 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”.

²By convention, 1 is ommitted (e.g. CO₂ instead of C₁O₂).
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 F1 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 \( y = \{y_1, \ldots, y_n\} \) given a sequence \( x = \{x_1, \ldots, x_n\} \) 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 \( x \) then amounts to finding \( y^* = \arg \max_y p(y \mid x) \).

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

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

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

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

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 \( x \) as well as the previous \( k \) labels. For example, in a first-order linear-chain CRF, one of the feature functions could be \( f_{[0 \rightarrow B, \text{capital}]}(y_{i-1}, y_i, x, i) \), which evaluates to 1 if \( y_{i-1} = O, y_i = B \) and \( x_i \) starts with a capital letter (otherwise it yields 0). Multiplication with the weight \( \theta_{[0 \rightarrow B, \text{capital}]} \) 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 \( f_{\text{capital}} \) denotes its corresponding set of first-order feature functions \( \{f_{[s 
rightarrow t, \text{capital}]}(y_{i-1}, y_i, x, i) \mid (s, t) \in \{(B, B), (B, I), (B, O), (I, B), (I, I), (I, O), (O, B), (O, O)\}\}^3 \) (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.

\[\text{Table 1.} \]

\[
\text{Transitions from } O \text{ to } I \text{ are invalid.}
\]
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.

<table>
<thead>
<tr>
<th>i</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>O</td>
<td>B-DrugN</td>
<td>O</td>
<td>B-DrugN</td>
<td>I-DrugN</td>
<td>I-DrugN</td>
<td>O</td>
<td>B-Group</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
</tbody>
</table>

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

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

Note that the sequence-labeling approach in the described form cannot cope with discontinuous entity mentions. Since only a tiny fraction (≈ 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 on-
tology and use the binned count as feature. The idea 
behind these features is that the specificity of an en-
tity correlates positively with its depth in the ontol-
ogy (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 com-
plete training corpus and evaluate on the test corpus 
of DDI Task 9.1 for each run respectively. In addi-
tion, 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 as additional training data (Run 3) further boosts the performance 
by 0.7 pp $F_1$ for strict evaluation.

### 4 Results

Table 3 shows micro-average CV results for identi-
fying and classifying mentions of pharmacological 
substances in the training corpus. The performance 
varies drastically between different entity classes re-
gardless 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 
macro-average $F_1$ measure increases by 0.8 pp for 
strict matching and 1.4 pp $F_1$ for exact matching.

The performance on the test corpus (see Table 4) is 
dramatically lower compared to CV results (e.g. 17.6 
pp $F_1$ for strict evaluation of Run 1). Except for enti-
ties of class Group, using domain-specific features 
leads to a superior performance for identifying and 
classifying mentions of pharmacological substances.

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 preci-
sion 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 enti-
ties 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 ($≈ 3\%$). 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 ($≈ 64\%$) or Group ($≈ 23\%$). This can be confirmed by 
the large differences between strict and exact match-
ing results shown in Table 3 and Table 4.

<table>
<thead>
<tr>
<th></th>
<th>Run 1</th>
<th>Run 2</th>
<th>Δ$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$P$</td>
<td>$R$</td>
<td>$F_1$</td>
</tr>
<tr>
<td>Drug</td>
<td>92.1</td>
<td>89.9</td>
<td>91.0</td>
</tr>
<tr>
<td>DrugN</td>
<td>54.7</td>
<td>9.3</td>
<td>15.9</td>
</tr>
<tr>
<td>Group</td>
<td>87.2</td>
<td>82.5</td>
<td>84.8</td>
</tr>
<tr>
<td>Brand</td>
<td>87.8</td>
<td>70.8</td>
<td>78.4</td>
</tr>
<tr>
<td>Exact</td>
<td>93.9</td>
<td>86.9</td>
<td>90.3</td>
</tr>
<tr>
<td>Strict</td>
<td>90.3</td>
<td>83.6</td>
<td>86.8</td>
</tr>
</tbody>
</table>

Table 3: Document-level 10-fold cross-validation 
micro-average results on the training corpus.
Table 4: Results on the test corpus. $\Delta F_1$ 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 competitive 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.


\(^5\)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


