Token- and Constituent-based Linear-chain CRF with SVM for Named Entity Recognition

Slavko Žitnik[1,2] and Marko Bajec[1]

[1] University of Ljubljana, Faculty of computer and information science,
Tržaška cesta 25, 1000 Ljubljana, Slovenia
{slavko.zitnik,marko.bajec}@fri.uni-lj.si
[2] Optilab d.o.o.,
Župančičeva 8, 5270 Ajdovščina, Slovenia

Abstract. The amount of chemical information is rapidly growing in the scientific literature and all other sorts of free text documents. We here propose a novel system that uses different types of linear-chain conditional random fields models and combines their results using a support vector machine classifier. We introduce the constituent-based models, which are in contrast to traditional token-based approaches, trained using noun phrase (i.e. possible mention) sequences. Furthermore, we employ two types of sequence labelings. For the first type the labels consist of a full set of target classes and for the second they consist of only binary labels, which denote whether the current node represents a chemical entity mention or not. We applied the proposed algorithm to the CHEMDNER 2013 task and for this report we evaluated it on the training and development datasets.

Key words: conditional random fields, named entity recognition, chemical compounds

1 Introduction

Named entity recognition is one of the main and the most thoroughly researched subtasks in the field of information extraction [12]. Due to the complexity of human language, syntatic errors and many different forms in which a specific entity can be represented as (i.e., chemical compound in our case), it is very difficult for a machine to automatically extract them from unstructured text.

In this paper we present our contribution to the Chemical compound and drug name recognition (CHEMDNER) task [7]. The task consists of two subtasks, of which chemical document indexing subtask (CDI) focuses on ranking of unique mentions from a specific document, and chemical entity mention recognition (CEM) subtask, which focuses on extraction of exact start and end indices that correspond to a specific chemical mention. We propose a system that combines four different types of linear-chain conditional random fields (CRF) models (using various sequences and labelings) with a support vector machine (SVM) classifier. First, we divide the models by type of tokens in a sequence, which can be represented as words or noun phrases (i.e. constituents). Then, each type of the model is trained against specific CEM classes and separately against binary labeling, which only identifies the presence of a chemical mention. Lastly, the SVM classifier decides whether the merged results of all these four CRF model combinations represent a valid mention or not.

The paper is structured as follows. In the next Section we briefly overview related work regarding named entity recognition using SVM and CRF classifiers. In Section 3 we present the proposed system and explain the the execution on an example document. After that we show and discuss the evaluation results on training and development dataset. Lastly, in Section 5 we conclude the paper and reveal the further work.

2 Related work

A vast majority of research in the text mining field focuses on the identification of named entities such as person names, locations, organizations, chemicals, drugs, etc. Traditional approaches are roughly classified as dictionary-based, morphology-based and context-based [6]. In our approach we indirectly use all of them within the CRF feature functions, which extract additional knowledge from the input data. We also use the DrugBank database as a third-party data source for a gazeteer-based feature function.

In the literature there are some approaches that combine CRF [1] and SVM [8] algorithms in order to extract named entities. CRF algorithm is intended to tag sequences, while SVM is better when we need to classify only an instance, described with feature values, into a specific class. It has been shown that for named entity recognition, CRF can achieve better results than SVM [9]. This holds especially for the cases when we use a lot of well-chosen features because in the field of natural language processing it is normal to have a few ten thousands or more features. There was also a two-stage SVM/CRF classifier proposed for identication of handwritten leters [10]. In contrast to our approach, the authors first employed an SVM classifier and then used its results as features within CRF. The most similar approach to the our method was done by Cai et al. [11], who first extracted entity boundaries using CRF and then used SVM along with other features to get final predictions. We also first use CRF classifiers, but we use more feature functions and different types of models and then according to the merged results from all of the models predict the final values using an SVM classifier.

Conditional random fields, which we mainly use, is a discriminative model that estimates the joint distribution $p(\overline{y}|\overline{x})$ over the target sequence $\overline{y}$ conditioned on the observed sequence $\overline{x}$. We tag the target sequence $\overline{y}$ using the label of specific CEM class or "O" for all other words. The algorithm was previously successfully employed for various sequence labeling tasks and can deal especially good with a large number of multiple, overlapping and non-independent features. Due to the speed of training and inference we use only simple linear-chain CRF (LCRF) model, which depends only on the current and previous label.

3 System description

In this section we introduce the proposed system used for the CHEMDNER shared task. The system takes a raw text document as input, employs preprocessing techniques on it, then it extracts chemical or drug mentions using four different types of CRF models, merges their results and returns final results using an SVM classifier. A high-level architecture of the system is shown in Figure 1 and the source code of the system is publicly available[1].

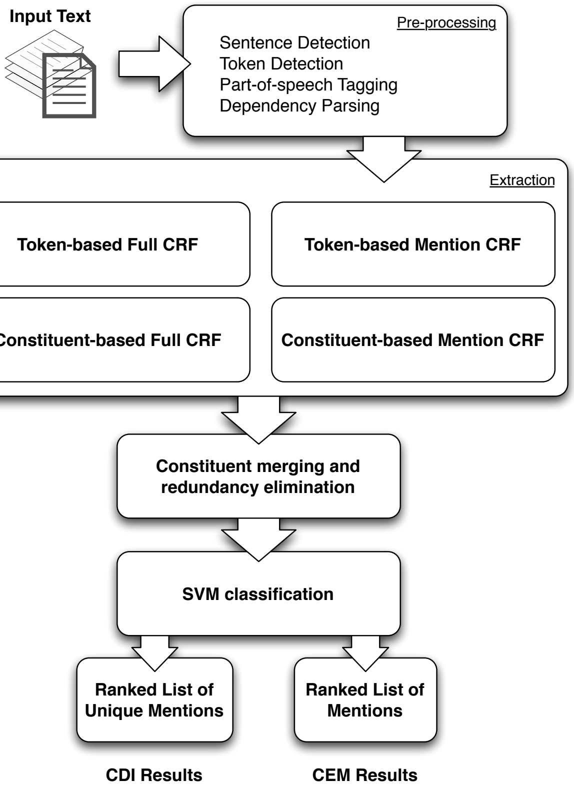

Fig. 1. High-level architecture of the proposed system for CHEMDNER task.

(I) The input text consists of unstructured text from which the system needs to recognize different chemical entity mention classes (CEM subtask) and to rank the extractions according to the probability, which defines how likely the extracted mention is correct (CDI subtask). For the CHEMDNER task the text

[1] `https://bitbucket.org/szitnik/iobie`

consists of PubMed titles and abstracts.

(II) Preprocessing part is responsible to transform the input data into the system's internal representation and to enrich the data with additional text labelings. We first detect sentences, perform tokenization [4] along with some specific manual rules (e.g., split tokens by greek characters or periodic system elements) and then enrich the data using part-of-speech tags [5].

(III) Extraction of mentions consists of four different types of LCRF models:

- *Token-based CRF* models are trained on all tokens from training data. One sequence of tokens represents all words along with additional labelings from one sentence. From these sequences we form two types of labelings and therefore train two different CRF models. The first is labeled according to specific CEM classes (i.e. *Full* type). The second is relabeled, so that each token is labeled as "`M`" if it represents a CEM class, otherwise it is labeled as "`O`" (i.e. *Mention* type).
- *Constituent-based CRF* models are trained against constituents (i.e. noun phrases), which consist of one or more consecutive words. A constituent is represented as a three-tuple, such as (start word index, end word index, target CEM class) for a specific document. We extract the noun phrase constituents from parse trees [4] of sentences and build one sequence of constituents per document. This sequence is then processed by a CRF algorithm. Also, for this type of model we employ Full type and Mention type of CRF models using the same approach as for Token-based models.

Thus, the four CRF models are of the following types: (1) Token-based Full, (2) Token-based Mention, (3) Constituent-based Full and (4) Constituent-based Mention model.

The engineering of informative feature functions is the main source of increase of precision and recall when training CRF classifiers. Usually these are given as templates and final features are generated by scanning the entire training data. Due to space limitations we do not overview all the used feature functions we used, but they can be retrieved from the source code repository[2].

(IV) Merging and redundancy elimination step combines the results of the extraction step to prepare them for the classification using an SVM. First we transform the extractions of Token-based CRFs into a constituent form (i.e., to be compliant with constituent-based CRF results), so therefore we can merge the identified mentions according to the position of their occurence within a document. The two constituents from a document are merged if they intersect by at least one token. In the final result we output the most probable constituent from the merged ones.

[2] See function *FeatureFunctionPackages.standardChemdner2013FFunctions()*

(V) SVM classifier [3] can be thought of as a meta-classifier because it is trained on top of the merged constituents from the previous step. Each SVM training instance could consist of four merged constituents, each detected by a separate CRF model. From each constituent we select its marginal probability and corresponding sequence probability that was assigned by a specific CRF model. Thus, from four merged constituents we use eight features, against which an SVM model is trained. The class label is binary value and shows whether the merged mentions represent a real CEM mention or not, which is also the final result of the SVM model.

(VI) The final results are exported in two different formats. For the CDI subtask we return a list of unique mentions for each document, which are ranked according to the maximum marginal probability. For the CEM subtask we return start and end indices of identified mentions and rank them according to the maximum tagged sequence probability.

3.1 A worked example

In this section we show an example of execution of the proposed approach using a sentence from the PubMed article `23444833` within the CHEMDNER training dataset.

(I) The input text (abstract) is as follows: "The lystabactins are composed of serine (Ser), asparagine (Asn), two formylated/hydroxylated ornithines (FOHOrn), dihydroxy benzoic acid (Dhb), and a very unusual nonproteinogenic amino acid, 4,8-diamino-3-hydroxyoctanoic acid (LySta)."

(II) Preprocessing transforms the input text and adds additional labelings such as POS tags to the input text: "`DT` The `NNS` lystabactins `VBP` are `VBN` composed `IN` of `NN` serine `-LRB-` (`NNP` Ser `-RRB-`) ..."

(III) Extraction part transforms the input data into appropriate sequences for each of the proposed CRF models (shown below) and trains them:

- *Token-based CRF Full*

The	lystabactins	are	composed	of	serine	(	Ser	)	,	asparagine	. . .
`O`	`FAMILY`	`O`	`O`	`O`	`TRIVIAL`	`O`	`FORMULA`	`O`	`O`	`TRIVIAL`	. . .

- *Token-based CRF Mention*

The	lystabactins	are	composed	of	serine	(	Ser	)	,	asparagine	. . .
`O`	`M`	`O`	`O`	`O`	`M`	`O`	`M`	`O`	`O`	`M`	. . .

- *Constituent-based CRF Full*

The lystabactins	serine	Ser	asparagine	Asn	two formylated/hydroxylated ornithines	. . .
`FAMILY`	`TRIVIAL`	`FORMULA`	`TRIVIAL`	`FORMULA`	`MULTIPLE`	. . .

– *Constituent-based CRF Mention*

```
The lystabactins serine Ser asparagine Asn two formylated/hydroxylated ornithines ...
        M          M    M      M       M                M                         ...
```

(IV) Merging and redundancy elimination step merges the extracted mentions from the previous step to prepare data for the SVM classifier. SVM class value is used for training and defines whether the merged constituent represents a CEM or not:

Constituent observable	Marginal and sequence probability pairs from CRF classifiers in the same order as in (III)				SVM class
"lystabactins"	(0.4, 0.2)	(0.38, 0.55)	(0.5, 0.35)	(0.6, 0.51)	yes
"serine"	(0.45, 0.24)	(0.37, 0.48)	(0.53, 0.38)	(0.63, 0.58)	yes
...	...				...
"nonproteinogenic"	(0.2, 0.3)	(0.22, 0.37)	(0.0, 0.0)	(0.0, 0.0)	no
...	...				...

(V) SVM classifier predicts whether the instances from previous step represent valid CEM mentions.

(VI) The final results are then exported for the CEM and CDI subtasks respectively (for CEM we output all probabilities as 0.5 because they are not important for the evaluation):

CEM results	CDI results
23444833 A:318:324 1 0.5	23444833 serine 1 0.63
23444833 A:289:301 2 0.5	23444833 lystabactins 2 0.6
23444833 A:448:464 3 0.5	23444833 nonproteinogenic 3 0.22
...	...

4 Results and discussion

We show the results of the evaluation on development and training data in Table 1. The submitted results were inferred by the models learnt using the same settings as these ones[3]. The results for *run1* and *run2* are retrieved only from both combinations of token-based LCRF models and for *run3*, the results are further classified using an SVM classifier. The results that we and other teams achieved on the CHEMDNER test dataset, are available in a separate paper [7].

[3] Values run1, run2, run3 correspond to the official CHEMDNER submissions.

From the results in Table 1 we can observe that token-based full and token-based mention CRF models produce similar results and that their combination using an SVM classifier results in a slightly better Micro-averaged F-score, but produces more false positive errors.

				Macro-average			**Micro-average**		
CEM	**TP**	**FP**	**FN**	**P**	**R**	**F**	**P**	**R**	**F**
Token Full CRF (run1)	18858	**3848**	10668	**79.6**	65.8	69.8	**83.1**	63.9	72.2
Token Mention CRF (run2)	19477	4619	10049	77.9	67.9	**70.4**	80.8	66.0	72.6
CRF + SVM (run3)	**19790**	4839	**9736**	77.2	**68.1**	70.2	80.4	**67.0**	**73.1**
CDI									
Token Full CRF (run1)	11324	**2884**	4771	**78.4**	72.1	**73.2**	**79.7**	70.4	74.7
Token Mention CRF (run2)	11582	3515	4513	76.2	73.6	73.1	76.7	72.0	74.3
CRFs + SVM (run3)	**11835**	3765	**4260**	75.2	**74.4**	73.0	75.9	**73.5**	**75.0**

Table 1. Overview of CEM and CDI results when the system is trained on the training dataset and tested on the development dataset.

The achieved results are fairly low, especially according to other general named entity recognition tasks. This is also due to the fact that we used feature functions based only on the observable syntactic values, part-of-speech tags and gazeteer-based feature functions (i.e., greek symbols and periodic table symbols). From the third-party sources we used only DrugBank database, which includes chemical names, brand names, IUPAC names and synonyms.

The approximate time that is needed to train and infer the results presented in Table 1 is about 1 hour. For the test dataset, which contains 20.000 documents, we needed about four hours to get the final results[4].

5 Conclusion

The paper proposes a novel multiple CRF and SVM-based approach for named entity recognition. The linear-chain CRF models are trained against standard token-based data and constituent-based sequences. Both of them also use different settings, once identifying the exact named entities or otherwise just identifying the presence of a chemical mention. We applied the algorithm to the CHEMDNER task (the results are accessible in [7]), evaluated it on training and development dataset and discussed the results using different settings.

In the future work we plan to include richer feature functions (e.g., from dependency parsing) and incorporate more data from publicly available databases to achieve better results. It would also be useful to weight specific predictions from different models for the final results.

[4] The machine we were working on is a standard Intel i7 processor with a 16GB of RAM.

Acknowledgments. The work has been supported by the Slovene Research Agency ARRS within the research program P2-0359 and part financed by the European Union, European Social Fund.

References

1. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: Proceedings of the Eighteenth International Conference on Machine Learning, pp. 282–289. Morgan Kaufmann, San Francisco (2001)
2. Okazaki N.: CRFSuite: a fast implementation of Conditional Random Fields, 2007, http://www.chokkan.org/software/crfsuite/
3. Hall M., Frank E., Holmes G., Pfahringer B., Reutemann P., Witten I.H.: The WEKA Data Mining Software: An Update. In: SIGKDD Explorations, pp. 10–18. ACM, New York (2009)
4. Socher, R., Bauer, J., Manning, C.D.,Ng, A.Y.: Parsing With Compositional Vector Grammars. In: Proceedings of ACL 2013. Association for Computational Linguistics, Stroudsburg (2013)
5. Apache OpenNLP, http://opennlp.apache.org/
6. Vazquez, M., Krallinger, M., Leitner, F., Valencia, A.: Text Mining for Drugs and Chemical Compounds: Methods, Tools and Applications. Mol. Inf. 30, 506–519 (2011)
7. Krallinger, M., Rabal, O., Leitner, F., Vazquez, M., Oyarzabal, J., Valencia, A.: Overview of the chemical compound and drug name recognition (CHEMDNER) task. Proceedings of the fourth BioCreative challenge evaluation workshop, vol. 2, (2013)
8. Corinna, C., Vapnik, V.N.: Support-Vector Networks. Machine Learning. 20, 1-31 (1995)
9. Li, D., Kipper-Schuler, K., Savova, G.: Conditional Random Fields and Support Vector Machines for Disorder Named Entity Recognition in Clinical Texts. In: Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing, pp. 94–95. Association for Computational Linguistics, Stroudsburg (2008)
10. Hoefel, G., Elkan, C.: Learning a Two-Stage SVM/CRF Sequence Classifier. In: Proceedings of the 17th ACM conference on Information and knowledge management, pp. 271–278. ACM, New York (2008)
11. Cai, P., Luo, H., Zhou, A.: Semantic Entity Detection by Integrating CRF and SVM. In: Proceedings of the 11th international conference on Web-age information management, pp. 483–494. Springer, Berlin (2010)
12. Sarawagi, S.: Information Extraction. Foundations and Trends in Databases. 1, 261–377 (2008)