Extracting adverse drug reactions and their context using sequence labelling ensembles in TAC2017

Maksim Belousov[1], Nikola Milosevic[1,4], William Dixon[2,3], and Goran Nenadic[1,2]

[1] School of Computer Science, University of Manchester, UK
[2] Health eResearch Centre, Farr Institute, Manchester Academic Health Science Centre, The University of Manchester, UK
[3] Arthritis Research UK Centre for Epidemiology, The University of Manchester, UK
[4] Manchester Institute of Innovation Research, Alliance Manchester Business School, The University of Manchester, UK
{maksim.belousov,nikola.milosevic,will.dixon,gnenandic}@manchester.ac.uk

Abstract. Adverse drug reactions (ADRs) are unwanted or harmful effects experienced after the administration of a certain drug or a combination of drugs, presenting a challenge for drug development and drug administration. In this paper, we present a set of taggers for extracting adverse drug reactions and related entities, including factors, severity, negations, drug class and animal. The systems used a mix of rule-based, machine learning (CRF) and deep learning (BLSTM with word2vec embeddings) methodologies in order to annotate the data. The systems were submitted to adverse drug reaction shared task, organised during Text Analytics Conference in 2017 by National Institute for Standards and Technology, archiving F1-scores of 76.00 and 75.61 respectively.

Keywords: health informatics, text mining, table mining, drug labels, adverse drug reactions

1. Introduction

Adverse drug reactions (ADRs) are unwanted or harmful effects experienced after the administration of a certain drug or a combination of drugs [8]. They present a challenge for drug development and drug administration. During 1994, it was estimated that 700,000 patients in the United States suffered from adverse drug reaction, while 100,000 died as a consequence of such reactions [7]. Roughly half of the people in the UK take prescribed medications. Adverse drug reactions are serious burden on health care systems. About 7% of all hospital admissions were accounted to ADRs. Moreover, quality of life and adherence to treatment is, as well, affected by adverse drug reactions [10]. Also, they are important source of human phenotypic data and can be used to predict drug targets [6].

In the United States, drug product labels are required by law to contain the information regarding clinically significant adverse drug reactions [16]. All drug

product labels in the United States are freely available through the National Library of Medicine's DailyMed website[5] in a standard format called Structured Product Label (SPL).

The task of recognising specific mentions (such as ADRs) in a text is a task of named entity recognition (NER) or tagging, which can be approached using sequence labelling techniques. Sequence labelling problems are usually solved by using sequence modelling machine learning techniques, such as hidden Markov models, conditional random fields or recurrent neural networks.

Within the drug informatics domain, the SPLICER system [3] was successfully applied to extract adverse drug events from text and tables in the Adverse Reactions section of SPLs. Other efforts focus on side effects and drug indications [4, 5, 1]. The SIDER (Side Effect Resource) database uses named entity recognition to extract side effects and indications from product labelling, including SPLs [6]. More recently, starting with full-text papers from the Journal of Oncology, drug side effect relationships were extracted and compared to the SIDER database [19].

Neural networks with word embeddings have recently showed successes in the biomedical named entity recognition. Word2vec embeddings with bidirectional recurrent neural networks combined with a CRF tagger and SVM classifier showed promising results for disease recognition [17]. Named entity recognition methodology based on recurrent neural networks and word embeddings (GloVe or Word2vec) was used for de-identification of electronic health records and gave the state-of-the-art results, with GloVe embeddings giving slightly better results [2].

In this paper, we present our approaches to the recognition of adverse drug reactions and related entities, developed for a shared task organised during the Text Analytics Conference 2017 (TAC 2017). The task was co-organised by the US National Institute of Standards and Technology (NIST) and the US Food and Drug Administration (FDA). The objective of the shared task was to extract adverse drug reactions from drug labelling text documents using natural language processing techniques. In the task 1, in which we participated, the participants were supposed to build a system to extract adverse drug reactions and related mentions such as severity, drug class, negation, factors, and whether it was reported on animals[6].

1.1. Data

The shared task organisers published a training dataset containing 101 annotated drug labels (documents) and a dataset containing 2,208 unannotated drug labels. An unseen subset of unannotated documents was used as a testing data during the task evaluation. The drug label is a multi-section document that may contain headings, paragraphs, tables and lists. In the provided dataset each drug

[5] `https://dailymed.nlm.nih.gov/dailymed/index.cfm`
[6] `https://bionlp.nlm.nih.gov/tac2017adversereactions/`

label was converted to a text document disregarding the structure (i.e. representing all elements as an unformatted text, keeping only the main sections of the document). It is worth noting that the gold-standard dataset contained some *discontinuous annotations* (6.8% of all annotations). Annotation that involves more than one continuous span of characters is considered discontinuous annotation. For the simplicity of tagging schemes, we ignored discontinuous annotations during the document parsing.

The class distribution of annotated entities is imbalanced, where the majority of annotations were adverse drug reactions. On the other hand, some related entities had only a few annotations. The numbers of annotated mentions (groups of tokens), tokens and the average number of tokens per mention are presented in Table 1. Lack of data for certain related entities presented a challenge for developing named entity recognition systems based on machine learning.

Entity class	**#mentions**	**#tokens**	**Avg. tk/mention**
Adverse drug reaction	12,792	21,258	1.66
Severity	863	1,306	1.51
Factor	602	653	1.08
Drug class	248	518	2.09
Negation	95	109	1.47
Animal	44	44	1.00

Table 1. The number of annotated mentions (group of tokens), number of tokens, and the average number of tokens per mention in the provided training data

2. System description

The architecture of the proposed systems consists of three stages: (1) document parsing, (2) word vectorisation, (3) tagging ADRs and their related entities. During the document parsing stage we attempt to restore the original structure of the document and recognise elements such as headings, tables (with rows and cells), lists (with items) and text paragraphs. The word vectorisation stage depended on the type of tagging model in the following stage and aimed to generate word vectors from text sequences using either hand-crafted features or unsupervised learning. The main task of tagging stage is to extract mentions of specific type from text by sequence labelling of extracted word vectors. Since some related entities rely on ADR mentions, they are performed separately, after the ADR tagging is completed. The pipeline is presented in Figure 1 and the following subsections provide details about each processing stage.

2.1. Document parsing

The aim of this stage is to perform re-engineering the structure of the document so that later their content can be treated differently. For instance, it might be

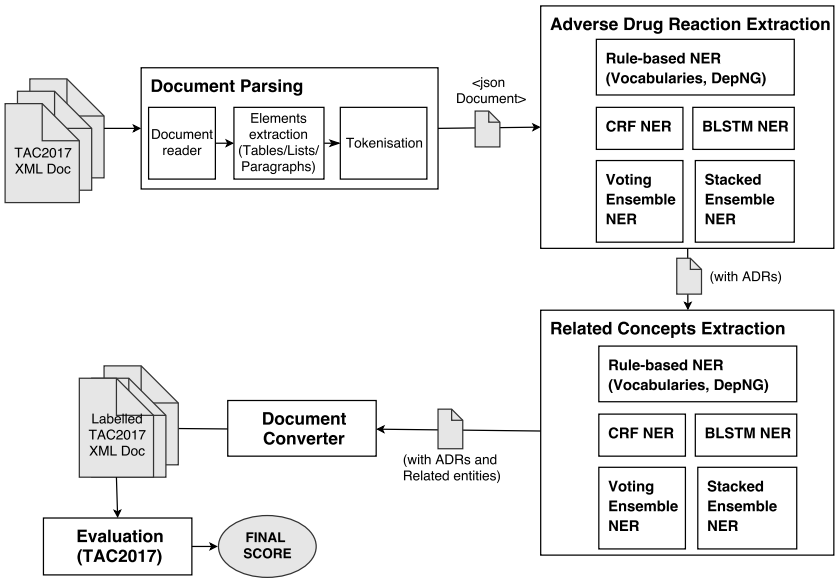

Fig. 1. Document processing and tagging pipeline.

beneficial to analyse the content of a table cell individually rather than the whole chunk of the text that contain multiple rows and cells. We identified four different document element types in the document:

- **Headings** are numbered titles for sections and sub-sections (e.g. *"5.1 Asthma-Related Death [See Boxed Warning]"*)
- **Tables** have *heading rows* and *content rows*, each of them is also having *cells*. Each row might have different number of cells. In addition, a table may have *caption* (which usually starts with "Table NUM.") and a *footer* that contain additional notes. We treated all text lines after the aforementioned caption trigger and before the paragraph separator (multiple empty lines) as potential table rows. Then, we categorised each row candidate as part of the caption, header, content and footer, based on the number of potential columns, numerical cells and words in each cell.
- **Lists** are groups of multiple bullet-points or *items*. Consequent text lines that starts with asterisk character (*) are considered as list items. List should have more than one item.
- **Paragraphs** are any other chunk of text separated with multiple new-line characters.

For some tagging models we applied two different document *splitting strategies*: (1) take the ***whole element*** (i.e. table, list, paragraph) and represent them as text or (2) take the textual content of ***sub-elements*** (such as table cells and list items) and treat them as individual items.

2.2. Tagging models

We utilised various types of tagging methods based on knowledge-driven rules, conditional random fields (CRF), bidirectional long short-term memory networks (BLSTM) and two different types of ensemble methods. We generated word vectors differently depending on the sequence labelling approach by using either hand-crafted features or obtaining word embeddings from unsupervised learning models trained on large text corpora.

Rule-based models

Rule-based methods are based on a knowledge-driven approach and manually curated dictionaries. In particular, we applied them for *negation* and *animal* classes, since there was not enough labelled data to be modelled by machine learning algorithms.

- To identify **negations**, we have developed a rule based tagger using the modification of DepND[7] that uses GENIA dependency parser [12] to recognise the scope of the negation and the dictionary of negation triggers. In particular, we added a list of phrases that need to be ignored if appeared in a negation phrase or scope (such as "not available" or "could not be assessed") and labelled negations only when an ADR mention is found inside the negation scope. We applied the negation tagger on the sub-element level (i.e. on table cells and list items).
- For the **animal** class, we made an assumption that animals are not mentioned in drug labels unless adverse events are reported on them during the trials. Also, there is a close set of animal spices that are usually used in medical experiments [9]. We have developed a dictionary-based tagger that labelled all mentions of animals from our list. The animal tagger was used on the sub-element level.

CRF models

Linear chain conditional random fields (CRF) is a linear statistical model that encodes conditional distributions $p(y|x)$ between observations (input features) and output variables (labels). Prior to passing a text input into the model, each sequence item (i.e. word or token) should be converted into a *feature vector*. In particular, we experimented with lexical features, part-of-speech tags, grammatical relations (dependencies), vocabulary and semantic features (such as corresponding semantic types and named entity tags from various medical systems). In order to capture the context for a given token, the mentioned features were extracted from a certain number of surrounding tokens (context window). All CRF models were used on the whole elements (i.e. tables, lists) represented as a text.

[7] `https://github.com/zachguo/DepND`

- For **ADR** mentions, we extracted word lemmas, part-of-speech tags (retrieved using the GENIA tagger [15]), UMLS semantic types (obtained using QuickUMLS [13]) and lexicon match (i.e. whether the current word is exist in the ADR lexicon[8]). We trained word2vec on lemmatised sentences extracted from 2,208 unannotated drug labels that were provided as a part of this task. In particular, we extracted 200-dimensional feature vectors from continuous bag-of-words model with a context window of size 5, trained with negative sampling using five noise words. Then we performed K-means clustering (n=50) of the word-vector space. For words that are found in the model, we used their corresponding cluster number, otherwise we used the lemma of the word as a feature. In order to capture the context we also extracted features from surrounding words (i.e. five preceding and five following words).
- For the **severity**, **factor** and **drug class**, we used a similar set of features with additional lexicon features. In particular, a lexicon for drug class was obtained from DrugBank[9] and Anatomical Therapeutic Chemical Classification System (ATC)[10], whereas for other aforementioned classes we experimented with lexicons obtained from the provided labelled data. We also added an additional binary feature that indicates whether the ADR is mentioned in the surrounding context.

BLSTM models

Bidirectional Long Short-Term Memory networks (BLSTM) are specific type of recurrent neural networks designed to learn long-term dependencies. In order to increase the amount of input information, the given sequence is read in both directions (forward and backward). For this tagging model we obtained word vectors from multiple word2vec models trained on large text corpora from generic and target domains. The generic 200-dimensional word embeddings were trained on a combination of PubMed and PMC texts with texts extracted from a recent English Wikipedia dump [11], whereas the target 200-dimensional word embeddings were trained on 2,208 unannotated drug labels. The *BLSTM* model was trained using RMSprop [14] algorithm with the learning rate of 1×10^{-5}. For regularisation, dropout with the rate of 0.1 was applied on each LSTM layer with 170 units. We trained BLSTM model for 50 epochs and used early stopping with patience of 10 epochs. Since this model does not rely on hand-crafted features, we used the same model configuration for both adverse reactions and related entities. For all entity types, we have trained a single BLSTM model on the whole elements (i.e. tables, lists) represented as a text.

Ensemble models We have created two different ensemble models:

[8] `http://diego.asu.edu/downloads/publications/ADRMine/ADR_lexicon.tsv`
[9] `https://www.drugbank.ca/`
[10] `http://www.atccode.com/`

- A *voting BLSTM and CRF ensemble* was training both CRF and BLSTM classifiers in parallel and selected the best candidate based on the highest average predicted probability of each class obtained from each classifier.
- A *stacked CRF-BLSTM ensemble* is our proposed modification of Wolpert's stacked generalisation [18] that firstly trains the CRF classifier, using the previously described features, and then utilises its predicted probabilities for each class to build an additional token-level embeddings for the BLSTM classifier. In this way, the obtained word vector has the dimension of the number of target classes used in CRF and its values will correspond to predicted probabilities.

For the voting and stacked ensembles we have utilised an ADR-specific feature extractor and trained a single ensemble model on all classes.

3. Evaluation of the tagging models on the training data

The provided labelled data contained 101 documents. We evaluated the supervised machine learning models using holdout cross-validation; therefore the dataset was split into training (56 documents), validation (24 documents) and testing (21 documents) sets. The rule-based models were evaluated on the whole dataset. The evaluation results for all developed taggers are presented in Table 2.

As it can be seen from Table 2, we calculated precision, recall and F1-score for labelling tokens in the document. Later, sequential labels are post-processed and merged into mentions.

Both ensemble models usually outperformed individual models especially in cases where there was enough training and testing samples. The stacked and voting ensembles performed relatively similar, although the stacked ensemble was slightly better in general. The F1-score for labelling adverse drug reactions ranges between 85%-87%, with the maximum score for the ensemble and BLSTM tagger. The BLSTM tagger performed better on the severity and factor classes. Drug class gave the best results on the test set with the CRF tagger, however, these results were quite unstable. While CRF performed on the test set with F1-score of 38%, on the validation set the F1-score was only 22%. The rule based approach gave the best results for the rare classes, such as negation and animal.

4. Runs and system evaluation

Using the evaluation results presented in the previous section, we have combined the best-performing taggers and created two systems which correspond to the two runs submitted for the final shared task evaluation (on test data).

- **Run #1:** We applied the `rule-based` approaches for the *Negation* and *Animal* classes. For *Adverse Drug Reactions* we utilised the `CRF` model with the hand-crafted features. For all other entity types (i.e. *Severity*, *Factor*

Entity class	Method	Precision	Recall	F1-score
ADR	CRF	90	82	86
	BLSTM	86	84	85
	Voting BLSTM+CRF	91	84	**87**
	Stacked CRF+BLSTM	90	85	**87**
Severity	CRF	67	51	58
	BLSTM	55	75	64
	Voting BLSTM+CRF	70	65	**67**
	Stacked CRF+BLSTM	58	71	64
Factor	CRF	52	20	29
	BLSTM	73	46	56
	Voting BLSTM+CRF	87	36	51
	Stacked CRF+BLSTM	82	41	55
Drug class	CRF	41	35	**38**
	BLSTM	57	21	31
	Voting BLSTM+CRF	62	12	20
	Stacked CRF+BLSTM	57	24	34
Negation	CRF	25	18	21
	BLSTM	22	12	15
	Voting BLSTM+CRF	50	06	11
	Stacked CRF+BLSTM	57	24	33
	Rule-based	66	66	**66**
Animal	CRF	76	100	87
	BLSTM	100	46	63
	Voting BLSTM+CRF	100	38	56
	Stacked CRF+BLSTM	40	31	35
	Rule-based	86	100	**93**

Table 2. Token-level evaluation of the taggers by the entity class and method used on the provided 101 labelled documents using holdout cross-validation.

and *Drug class*) we used the BLSTM tagger. The three related entities used one BLSTM model.
- **Run #2:** The `rule-based` tagger was applied only for the *Negation* class, whereas all other classes were handled with the `Stacked CRF+BLSTM` ensemble model.

4.1. Results

The systems were trained on the whole annotated dataset provided (101 documents) and applied on unannotated dataset for automatic tagging (2,208 documents). Then, sample of the automatically tagged documents were used for the evaluation. The primary metric for this evaluation was the micro-averaged F1-score. We have presented the system evaluation results in Table 3.

	Considering entity type			**Not considering entity type**		
Submission	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
Run #1	80.19	72.23	**76.00**	80.19	72.23	76.00
Run #2	76.84	74.36	**75.58**	76.87	74.39	75.61

Table 3. Performance of the submitted systems on the test data considering and not considering types of annotated entities. The primary metric used for the evaluation is marked in bold.

4.2. Discussions

The submitted systems had similar performance, with Run 1 having slightly better performance on the test data (by less than 0.5%). The achieved results are similar to the results obtained on the training data using 3-fold cross-validation (F1-score of 77.26 for the Run 1, and 76.61 for the Run 2).

The classes were not balanced. Some classes, such adverse drug reactions had a fair number of labelled entities in the training set, and therefore the machine learning models could be efficiently trained on this class. However, other classes were relatively small compared to the ADR class. Also, other classes were related to the ADR class and were only triggered if the ADR class is labelled in its vicinity. Context of the labels had significant importance in this task, as the same phrase is labelled depending on whether it is in vicinity of an ADR and whether it closer describes an ADR. For example word “serious” will be labelled as severity in context of “serious headache”, however, it will not be labelled in other contexts, such as for example in “serious consideration”.

On the other hand, some classes, such as animal and negation had a few of annotations in the training dataset. Therefore, it was impossible to train a machine learning model and it was necessary to develop a rule based approach. The rules for the negation class were considering context and whether in the scope of the negation is present an ADR. On the other hand, mentions of animals

unrelated to an ADR were rare. Therefore, it was safe to make an assumption that all animal mentions are related to adverse drug reactions.

4.3. Conclusion

In this paper we presented a number of different methodologies for labelling adverse drug reactions and related factors, severity, drug class, negation and whether they were reported on animals. We presented two systems made out of the best performing taggers that were submitted to the ADR track shared task of the Text Analytics Conference (TAC2017). The systems performed with F1-scores of 76% and 75.58% on the testing data, which we considering encouraging.

There is still space for improvement of the system and performing additional experiments. More informative features of the text could help improve the CRF machine learning taggers, while more representative word embeddings could be helpful for the BLSTM based taggers. This can be achieved using additional vocabularies, semantic resources and knowledge bases.

Other potential way to improve the performance of the tagging is to investigate alternative ensemble methods, e.g. utilise an additional meta-classifier to combine the CRF and BLSTM results. In addition, performance of the BLSTM model is directly depends on the word embeddings used, therefore alternative word representation models in addition to word2vec might be utilised, e.g. multi-level word representation or knowledge-infused word embeddings.

However, there is still challenge of labelling classes that have a low number of examples. In these cases, it is challenging to create a good performing machine learning models, because of the lack of examples. However, our rule based approaches can be further improved with additional samples and looking at additional data. Also, machine learning performance can be probably improved by using additional annotated data and external data sets.

References

1. Boyce, R., Gardner, G., Harkema, H.: Using natural language processing to extract drug-drug interaction information from package inserts. In: BioNLP: Proceedings of the 2012 Workshop on Biomedical Natural Language Processing. pp. 206–213
2. Dernoncourt, F., Lee, J.Y., Uzuner, O., Szolovits, P.: De-identification of patient notes with recurrent neural networks. Journal of the American Medical Informatics Association 24(3), 596–606 (2017)
3. Duke, J., Friedlin, J., Li, X.: Consistency in the safety labeling of bioequivalent medications. Pharmacoepidemiology and drug safety 22(3), 294–301 (2013)
4. Fung, K.W., Jao, C.S., Demner-Fushman, D.: Extracting drug indication information from structured product labels using natural language processing. Journal of the American Medical Informatics Association 20(3), 482–488 (2013)
5. Khare, R., Li, J., Lu, Z.: Labeledin: cataloging labeled indications for human drugs. Journal of biomedical informatics 52, 448–456 (2014)
6. Kuhn, M., Letunic, I., Jensen, L.J., Bork, P.: The sider database of drugs and side effects. Nucleic acids research p. gkv1075 (2015)

7. Lazarou, J., Pomeranz, B.H., Corey, P.N.: Incidence of adverse drug reactions in hospitalized patients: a meta-analysis of prospective studies. Jama 279(15), 1200–1205 (1998)
8. Lee, A.: Adverse drug reactions. Pharmaceutical press (2006)
9. Mukerjee, M.: Trends in animal research. Scientific American 276(2), 86–93 (1997)
10. Pirmohamed, M., James, S., Meakin, S., Green, C., Scott, A.K., Walley, T.J., Farrar, K., Park, B.K., Breckenridge, A.M.: Adverse drug reactions as cause of admission to hospital: prospective analysis of 18 820 patients. Bmj 329(7456), 15–19 (2004)
11. Pyysalo, S., Ginter, F., Moen, H., Salakoski, T., Ananiadou, S.: Distributional semantics resources for biomedical text processing (2013), `http://bio.nlplab.org/`
12. Sagae, K., Tsujii, J.: Dependency parsing and domain adaptation with lr models and parser ensembles. In: Emnlp-conll. vol. 2007, pp. 1044–1050. Prague, Czech Republic (2007)
13. Soldaini, L., Goharian, N.: Quickumls: a fast, unsupervised approach for medical concept extraction. In: MedIR workshop, sigir (2016)
14. Tieleman, T., Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning 4(2), 26–31 (2012)
15. Tsuruoka, Y., Tateishi, Y., Kim, J.D., Ohta, T., McNaught, J., Ananiadou, S., Tsujii, J.: Developing a robust part-of-speech tagger for biomedical text. In: Panhellenic Conference on Informatics. pp. 382–392. Springer (2005)
16. US Food and Drug Administration: Cfr-code of federal regulations title 21. Current good manufacturing practice for finished pharmaceuticals Part 211 (2014)
17. Wei, Q., Chen, T., Xu, R., He, Y., Gui, L.: Disease named entity recognition by combining conditional random fields and bidirectional recurrent neural networks. Database 2016 (2016)
18. Wolpert, D.H.: Stacked generalization. Neural networks 5(2), 241–259 (1992)
19. Xu, R., Wang, Q.: Large-scale automatic extraction of side effects associated with targeted anticancer drugs from full-text oncological articles. Journal of biomedical informatics 55, 64–72 (2015)