Improving Chemical-induced Disease Relation Extraction with Learned Features Based on Convolutional Neural Network

Hoang-Quynh Le
Faculty of Information Technology,
University of Engineering and Technology,
VNUH
Hanoi, Vietnam
lhquynh@vnu.edu.vn

Duy-Cat Can
Faculty of Information Technology,
University of Engineering and Technology,
VNUH
Hanoi, Vietnam
catcd_58@vnu.edu.vn

Thanh Hai Dang
Faculty of Information Technology,
University of Engineering and Technology,
VNUH
Hanoi, Vietnam
hai.dang@vnu.edu.vn

Mai-Vu Tran
Faculty of Information Technology,
University of Engineering and Technology,
VNUH
Hanoi, Vietnam
vutm@vnu.edu.vn

Quang-Thuy Ha
Faculty of Information Technology,
University of Engineering and Technology,
VNUH
Hanoi, Vietnam
thuyhq@vnu.edu.vn

Nigel Collier
Department of Theoretical and Applied Linguistics
University of Cambridge
Cambridge, UK
nhc30@cam.ac.uk

***Abstract*— There have been an increasing number of various machine learning-based models successfully proposed and applied for automatic chemical-induced disease (CID) relation extraction. They, however, usually require carefully handcrafted rich feature sets, which rely on expert knowledge, thus require expensive human labor but normally still cannot generalize data well enough. In this paper, we propose a CID relation extraction model that learns features automatically through a Convolutional Neural Network (CNN) instead of traditional handcrafted features. We exploit the shortest dependency path between a disease and a chemical for identifying their CID relation. Dependency relations, with and without their direction information, are further investigated. Experimental results on benchmark datasets (namely the BioCreative V dataset) are very potential, demonstrating the effectiveness of our proposed model for CID relation extraction.**

Keywords—chemical-induced disease; relation extraction; deep learning; convolutional neural network

I. INTRODUCTION

Chemicals, diseases and their relations always attract much interest of pharmaceutics and biomedicine research community. Discovering adverse drug reactions (ADR) and chemical safety/toxicity among patient groups are essentially important for getting insights into drugs' profile. In recent years, there has been an increasing focus on developing computational approaches for automatic recognition of chemicals, diseases and especially their ADR (or CID) from large-scale biomedical literature. To accelerate progress of text mining on identification of unexpected side-effect relations between chemicals and diseases, the annual challenge BioCreative V[1] (2015) held a chemical-disease relation (CDR) track, with sub-task DNER on the diseases and chemicals/drugs named entity recognition, and sub-task CID on chemical-induced disease (CID) relation extraction.

As automatic relation extraction from unstructured text is a complex problem with several challenges, simple approaches such as co-occurrence statistics [1] or rule-based methods [2] often do not produce satisfactory results. Therefore, machine learning-based methods are currently one of the top choices for relation extraction. According to the overview of the BioCreative V CDR task [3], in total of 18 teams who participated in the CID sub-task, 11 used machine learning methods, and 5 combined machine learning with pattern-based methods. Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Naïve Bayes, Maximum Entropy, Conditional Random Field (CRF), etc. are among commonly used methods. These traditional machine learning methods usually require a rich feature set which is handcrafted based on experts' domain knowledge. Features used in CID relation extraction models in BioCreative V are very abundant [3], including lexical features, semantic features, statistical features, syntactic features, document features, distributional features, dictionary features, and even word embedding. In this regard, our previous work for participation in the BioCreative V [4, 5] used a very rich feature set for SVM classifier, which includes feature vectors of >300.000 dimensions.

In this paper, we propose a CID relation extraction model that learns features automatically through CNN using SDP between a disease and a chemical, further taking into account

[1] https://www.biocreative.org /

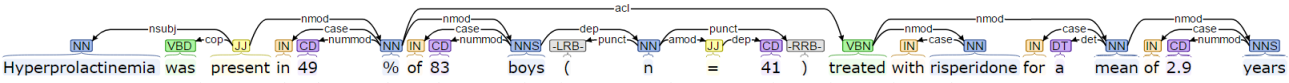

Figure 1: The shortest dependency parsing for an example sentence from BioCreative V data (PMID 20331935)

dependency relations, with and without their direction information. As a result, we integrate this model into our system named UET-CAM, which was ranked 4th in the BioCreative 2016 CDR track.

The novel contributions of this paper are as follows: (i) we propose a CNN-based model for CID relation extraction, (ii) we demonstrate the efficacy of using SDPs for CID relation extraction.

II. RELATED WORKS

Recently, deep Neural Networks have increasingly been used for various NLP related tasks. They have been shown to be extremely good at automatically feature engineering from noisy data, thus, not requiring a handcrafted feature set but still yielding good performances. Zeng et al. (2014) [6] used all words in sentences and a special feature related to positions of two involving entities for relation classification using a Convolutional Neural Networks (CNN). In 2015, Nguyen and Grishman [7] re-constructed this model and tested on various datasets for evaluating. Based on this model, Shen and Huang (2016) [8] proposed a word level attention mechanism that is able to better determine which parts of a sentence are most influential with respect to the two entities of interest.

We note that more attentions have been paid to modeling the shortest dependency path (SDP) of sentences. Xu et al. (2015a) [9] compared CNN based relation extraction models with and without using the directed and/or undirected SDP, as well as the use of negative sampling based on the reversed SDP. Liu et al. (2015) [10] developed an Augmented Dependency Path-based CNN model for relation extraction, in which the CNN was used to capture informative features on the SDP and the recursive neural network to model sub-trees of each node on the SDP. Xu et al. (2015b) [11] also used on SDP for relation classification, but they use a Long short term memory (a special case of Recurrent Neural Network). Mehryary et al. (2016) [12] proposed a voting system of 15 RNNs model with LSTM unit, they embedded tokens, their POS tags and dependency relations in SDP separately and then concatenate them to pass through a classifier. Cai et al. (2016) [13] used a bidirectional architecture to learn relation representations with direction information along both the forward and backward SDP at the same time. Their model combined CNN and two channel RNN with LSTM units.

III. SHORTEST DEPENDENCY PATH

Given a sentence, dependency tree is a special structure in which every two neighboring words are connected by a dependency relation. It may have more than one path between two words, and the path going through the fewest number of words is called shortest dependency path (SDP). To extract the relationship between two entities, the most direct approach is using the SDPs because they usually contain necessary information to identify their relationship [15].

For example, Fig. 1 shows the dependency parsing result of sentence "*Hyperprolactinemia was present in 49% of 83 boys (n = 41) treated with risperidone for a mean of 2.9 years*" which is generated by Stanford CoreNLP[2] tool. Tags such as "*nsubj*", *"acl"* and "*nmod*" are called the dependency relations[3] between two tokens. In this sentence, there is a CID relation between disease "*hyperprolactinemia*" and chemical "*risperidone*". There are three different ways to represent the SDP between *"hyperprolactinemia"* and *"risperidone"*.

(1) SDP without dependency relations: Only use sentence tokens that appear on the SDP

Hyperprolactinemia *present % treated* ***risperidone***

(2) Undirected SDP: Use both words and the relations between each word pair in the SDP, but does not consider directions of the relations.

Hyperprolactinemia *(nsubj) present (nmod) % (acl) treated (nmod)* ***risperidone***

(3) Directed SDP: Use both words and the relations between each word pair in the SDP with their directions.

$$\textbf{\textit{Hyperprolactinemia}}\ \overleftarrow{nsubj}\ \textit{present}\ \overrightarrow{nmod}\ \%\ \overrightarrow{acl}\ \textit{treated}\ \overrightarrow{nmod}\ \textbf{\textit{risperidone}}$$

In this paper, we perform all three representations and make the comparison between them.

IV. THE PROPOSED MODEL

In our previous UET-CAM system, inter-sentence relations (cross-sentence relations) were converted into intra-sentence relations (i.e., the relations between a chemical and a disease which are mentioned in the same sentence) before passed through the machine learning phase, and this paper only considers the intra-sentence CID relation.

Fig. 2 illustrates our model for CID relation extraction. The pre-processing and mention generation phase were done in our previous research [4, 5]. After these two phases, we have the set of sentences that include at least one chemical mention and one disease mention, and then we build the SDPs between these chemical-disease pairs. For classifying whether a sentence contains CID relations or not, we classify the SDP between its disease mention and chemical mention using a CNN model.

[2]Stanford CoreNLP – Core natural language software: http://corenlp.run

[3]More detailed description for dependency relations: http://universaldependencies.org/u/dep/all.html#al-u-dep/

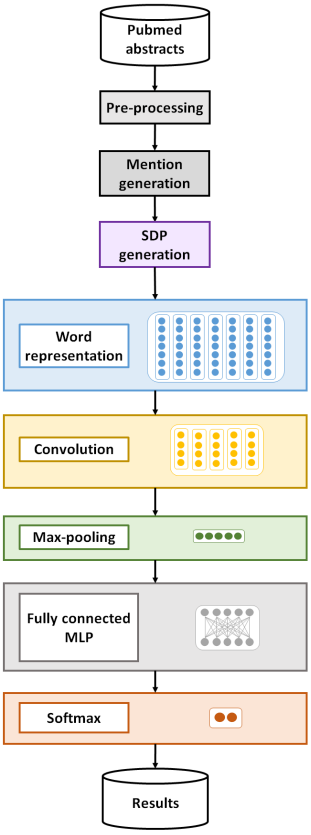

Figure 2. An Overview of proposed model

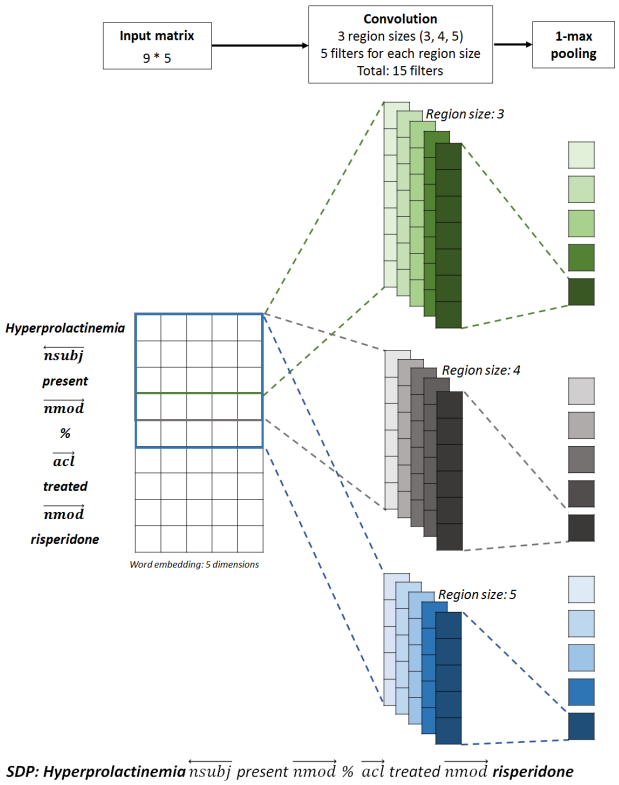

Figure 3. An example of word representation, convolution and max-pooling phase

Our CNN is built based on the original CNN model which was proposed for text classification by Kalchbrenner et al. (2014) [16] and Kim (2014) [17].

1. Word representation

Since the input of CNN model is the matrix with fixed size, we heuristically choose a fixed length of the input and then insure it by trimming longer input or padding shorter input if necessary. The key point of using CNN as well as other deep learning methods for text processing is transforming all tokens into numeric vector space but still keep its semantic and syntactic properties. This can be done by looking up a word embedding table that can be initialized by using a pre-trained word embedding model. In an example in Fig. 3, applying a 5-dimension word embedding for the input of 9 tokens, we have a 9 * 5 input matrix.

2. Convolution

The matrix representing the input, which were created in the previous step, is fed into the convolutional layer to extract higher abstract level features. We perform convolution on the input matrix via linear filters, i.e., we use filters for choosing the important features from the input matrix and the values of filters are generated through learning process. The feature extraction is thus automatic. Because each rows in the input matrix represent one token, the filters must have widths equal to the dimension number of word embedding. In this phase, we use various region sizes and various filters for each size, which allows CNN model to capture wider ranges of n-grams that are helpful for relation extraction. For moving each filter along the sentence length, we obtain one feature map. The dimensionality of the feature map generated for different region size is not constant, it varies as a function of the sentence length and the filter region size. For example, in Fig. 3, filters of region sizes 3-4-5 generate 7-6-5 dimensions feature maps, respectively.

3. Pooling

Pooling phase is used to abstract the features generated from the previous convolution phase. The common strategy is max-pooling, because of two reasons (1) it can choose the most important or relevant features from feature maps and (2) it can induce a fixed-length vector. In this paper, we use 1-max pooling [18], which extracts a scalar node with highest value from each feature map.

4. Regularization and classification

In this phase, values from the 1-max pooling phase for all filters are concatenated into a single feature vector to represent the input. In which, the dimensionality of feature vector is equal to the number of filters we used. Then we execute a "dropout" [19] as a mean of regularization by randomly setting a proportion of the feature vector's elements to 0. After dropout

step, new feature vector is then fed into a fully connected multilayer perceptron network (MLP). This MLP may have 0, 1 or more hidden layers. Output of this MLP is a vector, and softmax function is then applied to this output vector to perform classification.

V. EXPERIMENTS AND EVALUATION

1. *Data set*

TABLE I. SUMMARY OF THE CDR TRACK DATA SET

Data set	Articles	Chemical		Disease		CID+
		*Men**	*ID*	*Men**	*ID*	
Training	500	5,203	1,467	4,182	1,965	1,038
Development	500	5,347	1,507	4,244	1,865	1,012
Test	500	5,385	1,435	4,424	1,988	1,066

*Men: Mention, +CID: CID relations

Our experiments are conducted on the BioCreative V data. It is an annotated text corpus that consists of human annotations for all chemicals, diseases and their chemical-induced disease relations at the abstract-level. This corpus contains a total of 1,500 PubMed articles that are separated into three subsets, each of 500 for the training, development and test set (the details are shown on Table I). According to the data survey of BioCreative [20], most abstracts were selected from an existing CTD-Pfizer collaboration related dataset [21], there are 100 new curated abstracts and they are incorporated into the test set.

2. *Word embedding*

The SDP consists tokens and relations between them.

- For embedding tokens in the input, we use an open source word2vec model provided by NLPLab[4] [22], which were pre-trained using all PubMed abstracts and PMC full texts (4.08 million distinct words) with 200 dimensions.
- We note that SDP relations such as *nmod:in*, *acl*, etc. do not appear in the public available pre-trained model such as NLPlab word2vec. We use the one-hot vector for them, the dimensionality is set to the same with that of token word embedding (200 dimensions).

3. *Experimental system configuration*

We implement the proposed model based on Keras[5], using the Python programming language. We trained the model on the training data and tested them on the development data for tuning. As a result, the configurations as follows are chosen:

- Dropout: 0.25
- Epoch: 900
- Region sizes: 2-3-4-5
- Filters for each region size: 24 (corresponds to 4 region sizes, we use total 96 filters).
- Hidden layer of fully connected MLP: 1
- Optimizer: RMSProp

4. *Results and evaluation*

The test set was used for evaluation, the relations (represented by chemical- disease mention pairs) were compared to the golden standard annotation using three standard metrics: precision (P), recall (R) and F1. In which, P indicates the percentage of system positives that are true instances, R indicates the percentage of true instances that the system has retrieved, and F1 is the harmonic mean of P and R.

The experiments are conducted following 2 different experimental strategies:

- 10-fold cross validation on the train and the development data set. The results are evaluated in sentence-level and abstract-level, with both micro-average and macro-average P, R, F1.
- The model was trained on training and development data set and then tested on the test set. The results are evaluated in sentence-level and abstract-level. Then the abstract-level results are used to compare with those from other systems.

a) *Results of the proposed system*

For comparison, we also apply SVM instead of MLP and softmax function on the last phase. i.e., after 1-max pooling, the convolutional vector is fed into a SVM classifier. We use the Liblinear tool[6] to train a supervised binary SVM classifier (L2 regularized and L1 loss), which is similar with [5].

Table II gives the full system's results on training and development set after 10-fold cross validation. For these experiments, we use the NER tags provided in the golden corpus and new mentions obtained from the co-reference resolution module of our previous UET-CAM system. Using the configuration that we mentioned above, our system achieves the micro-average F1 of 71.75% at the sentence-level and 68.90% at the abstract level. The results for micro-average P are 76.31% at the sentence-level and 60.48% at the abstract level. Those for micro-average R are 67.70% and 80.04%, respectively. These results are very potential, since CID relations are very complex and even include cross-sentence relations. Moreover, the model that uses MLP and softmax for classification has better results than the model that uses SVM has, but the difference is not significant (less than 1% of F1.

[4]Pre-trained word embedding model is provided by Biomedical natural language processing lab: http://bio.nlplab.org/

[5]Deep Learning library for Python: https://github.com/fchollet/keras

[6]LIBLINEAR -- A Library for Large Linear Classification: http://www.csie.ntu.edu.tw/~cjlin/liblinear/

TABLE II. THE PROPOSED SYSTEM'S RESULTS ON TRAINING AND DEVELOPMENT SET (10-FOLD CROSS VALIDATION)

			CNN+MLP-softmax	CNN + SVM
Sentence-level	***Micro****	***P***	76.31	74.15
		R	67.70	67.85
		F1	71.75	70.86
	Macro*	***P***	76.20	74.23
		R	67.65	68.01
		F1	71.65	70.98
Abstract-level	***Micro****	***P***	60.48	59.83
		R	80.04	80.16
		F1	68.90	68.52
	Macro*	***P***	60.57	59.22
		R	80.13	80.36
		F1	68.91	68.19

Both models use directed SDP with dependency relations
*Micro-average, *Macro-average

TABLE III. SYSTEM RESULTS ON TEST SET

	Sentence-level			Abstract-level		
	P	***R***	***F1***	***P***	***R***	***F1***
CNN + MLP-Softmax*	74.49	63.86	68.77	58.02	76.20	65.88
CNN + SVM*	73.69	65.65	69.44	56.80	77.70	65.63

* Both models use directed SDP with dependency relations

Table III shows the results evaluated on the test set when the system is trained on the training and development set. As mentioned above, the test set includes some new data that may be useful to evaluate the performance and the adaptability of the system when applied in reality. Comparing with the micro-average results in table II, at sentence-level, our system result decreases about 1.82% of P, 3.84% of R and 2.98% of F1. At the abstract-level, these are 2.46%, 3.84% and 3.02% of P, R and F1, respectively. These results are partially met our expectations when applying the proposed system to new data. The model that uses SVM for classification yields the similar results in 10-fold cross validation. The difference between results of two models in the test set is very small.

b) Comparing the use of different input representations

In these experiments, we compare the results when using different input representations to choose the most accordant representation for the CNN model. Table IV shows the comparative results on the training and development data set (10-fold cross validation) and Table V shows the results of the test set. Firstly, we use all words between two interested entities (called word-based) for classifying their relations and comparing the results. The results have demonstrated the effectiveness of using SDP (called SDP-based), even the weakest SDP-based model still brings better result than word-based model. At the sentence-level, SDP-based models outperform the word-based (better more than 10% of micro-average F1). At the abstract level, may be the word-based model is good at common relations, which have high frequencies. The difference between the results of the SDP-based and word-based model is less than that at the sentence-level. It can be explained by the fact that using all words may include much irrelevant information, especially when two entities are at a long distance. We also make a comparison between three ways of representing the SDP: SDP without dependency relations, undirected SDP and directed SDP. Our results are not similar to the one published by Xu et al. (2015a) [9], which demonstrates the benefit of using direction of SDP. For both 10-fold cross validation on the training and development set (Table IV) and the test set (Table V), the results do not change much for different SDP representations. A possible reason may be that our representation of relations (directed or undirected) with one-hot vector is not good enough and cannot represent the semantic information of them.

TABLE IV. COMPARING RESULTS OF USING DIFFERENT INPUT REPRESENTATIONS ON THE TRAINING AND DEVELOPMENT SET (10-FOLD CROSS VALIDATION)

			Word-based	SDP-based		
				Without depRE*	***Undirected SDP***	***Directed SDP***
Sentence-level	***Micro****	***P***	67.00	73.01	75.75	76.31
		R	56.57	71.82	71.10	67.70
		F1	61.34	72.41	73.35	71.75
	Macro **	***P***	67.23	73.46	75.87	76.20
		R	56.22	71.51	71.31	67.65
		F1	60.74	72.10	73.32	71.65
Abstract-level	***Micro****	***P***	60.30	55.39	59.26	60.48
		R	72.54	82.04	82.38	80.04
		F1	65.85	66.13	68.91	68.90
	Macro **	***P***	60.73	56.46	59.73	60.57
		R	72.48	82.17	82.46	80.13
		F1	65.54	66.45	69.09	68.91

*SDP without dependency relations
*Micro-average, **Macro-average

TABLE V. COMPARING RESULTS OF USING DIFFERENT INPUT REPRESENTATIONS ON THE TEST SET

		Sentence – level			Abstract-level		
		P	***R***	***F1***	***P***	***R***	***F1***
Word-based		68.01	44.82	54.03	61.07	60.74	60.91
SDP-based	***Without depRE****	72.95	66.30	69.47	55.54	77.43	64.69
	Undirected SDP	72.92	67.66	70.19	55.22	78.11	64.70
	Directed SDP	74.49	63.86	68.77	58.02	76.20	65.88

*SDP without dependency relations

VI. CONCLUSIONS

In this article, we have presented our systematic study for chemical-induced disease relations extraction. We propose to use CNN for automatic feature extraction (instead of hand-crafted feature set) in a CID prediction model that makes use of SDP. The proposed method is then integrated into our previous UET-CAM system [4, 5] to be evaluated on the BioCreative V benchmark data. The experimental result on the test set archives 58.02% of P, 76.20% of R, 65.88% of F1 (abstract-level). This result shows the potential effectiveness of the CNN-based feature extraction and the use of SDP in CID prediction.

For evaluating the effectiveness of the whole model, specifically the combination of CNN based feature extraction

and MLP-softmax based classification, we build a comparative model that uses SVM instead. The results demonstrate the rationality of our proposed model: the MLP and softmax classifier brought better result than the SVM classifier on both 10-fold cross validation and the test set evaluation.

Several experiments are made for comparing different input representations, including: words between two involving entities, SDP without dependency relations, undirected SPD and directed SDP. The results demonstrate the advantage and robustness of using SDP compare to the raw words. It, however, also shows that we need to improve the word embedding strategy to represent the dependency relations and their directionalities better.

Some more experiments will be done in the near future, such as comparing our proposed model with traditional machine learning models, tuning the SVM classifier, etc. Our system is extensible in several ways. For the first one, the word embedding method should be able to exploit directed dependency relations in SDP. The second approach may come from several useful biomedical resources that we do not utilize in this work. The last one may be the application of several state-of-the-art deep learning methods on a hybrid scheme for improving the prediction performance.

ACKNOWLEDGMENT

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.05-2016.14.

REFERENCES

[1] Chen, E. S., Hripcsak, G., Xu, H., Markatou, M., & Frie88dman, C. (2008). Automated acquisition of disease–drug knowledge from biomedical and clinical documents: an initial study. Journal of the American Medical Informatics Association, 15(1), 87-98.

[2] Liu,J., Li,A. and Seneff, S. (2011) Automatic drug side effect discovery from online patient-submitted reviews: Focus on statin drugs. *Proceedings of First International Conference on Advances in Information Mining and Management (IMMM), Barcelona, Spain (pp. 23-29)*.

[3] Wei,C.H., Peng,Y., Leaman,R. et al. (2016). Assessing the state of the art in biomedical relation extraction: overview of the BioCreative V chemical-disease relation (CDR) task. Database (Oxford) (2016) 2016: baw032.

[4] Le, H. Q., Tran, M. V., Dang, T. H., & Collier, N. (2015). The UET-CAM system in the BioCreAtIvE V CDR task. In Fifth BioCreative challenge evaluation workshop (pp. 208-213).

[5] Le, H. Q., Tran, M. V., Dang, T. H., Ha, Q. T., & Collier, N. (2016). Sieve-based coreference resolution enhances semi-supervised learning model for chemical-induced disease relation extraction. Database, 2016, baw102.

[6] Zeng, D., Liu, K., Lai, S., Zhou, G., & Zhao, J. (2014, August). Relation classification via convolutional deep neural network. COLING 2014.

[7] TH Nguyen, R Grishman (2015). Relation extraction: Perspective from convolutional neural networks. NAACL-HLT 2015, 39-48

[8] Shen, Y., & Huang, X (2016). Attention-Based Convolutional Neural Network for Semantic Relation Extraction. COLING 2016.

[9] Xu, K., Feng, Y., Huang, S., & Zhao, D. (2015). Semantic relation classification via convolutional neural networks with simple negative sampling. EMNLP 2015.

[10] Liu, Y., Wei, F., Li, S., Ji, H., Zhou, M., & Wang, H. (2015). A dependency-based neural network for relation classification. ACL 2015.

[11] Xu, Y., Mou, L., Li, G., Chen, Y., Peng, H., & Jin, Z. (2015). Classifying Relations via Long Short Term Memory Networks along Shortest Dependency Paths. In EMNLP (pp. 1785-1794).

[12] Mehryary, F., Björne, J., Pyysalo, S., Salakoski, T., & Ginter, F. (2016). Deep Learning with Minimal Training Data: TurkuNLP Entry in the BioNLP Shared Task 2016. ACL 2016, 73.

[13] Cai, R., Zhang, X., & Wang, H (2016). Bidirectional Recurrent Convolutional Neural Network for Relation Classification. ACL 2016.

[14] Davis,A.P., Murphy,C.G., Saraceni-Richards,C.A. et al. (2009) Comparative Toxicogenomics Database: a knowledgebase and discovery tool for chemical–gene–disease networks. Nucleic acids research, 37(suppl 1), D786-D792.

[15] Bunescu R. C., Mooney R. J. A shortest path dependency kernel for relation extraction. Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing; October 2005; Vancouver, Canada. Association for Computational Linguistics; pp. 724–731.

[16] Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. In Proceedings of ACL 2014.

[17] Kim, Y. (2014). Convolutional neural networks for sentence classification. In Proceedings of EMNLP 2014.

[18] Boureau, Y. L., Ponce, J., & LeCun, Y. (2010). A theoretical analysis of feature pooling in visual recognition. In Proceedings of the 27th international conference on machine learning (ICML-10) (pp. 111-118).

[19] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580.

[20] Li,J., Sun,Y., Johnson,R. et al. (2015) Annotating chemicals, diseases, and their interactions in biomedical literature. Proceedings of the Fifth BioCrea-tive Challenge Evaluation Workshop, 154-166.

[21] Davis,A.P., Wiegers,T.C., Roberts,P.M. et al. (2013) A CTD–Pfizer collaboration: manual curation of 88 000 scientific articles text mined for drug–disease and drug–phenotype interactions. Database, 2013, bat080.

[22] Pyysalo, S., Ginter, F., Moen, H., Salakoski, T. and Ananiadou, S. (2013). Distributional Semantics Resources for Biomedical Text Processing. LBM 2013