YNU-junyi in BioNLP-OST 2019: Using CNN-LSTM Model with Embeddings for SeeDev Binary Event Extraction

Junyi Li, Xiaobing Zhou*, Yuhang Wu, Bin Wang
School of Information Science and Engineering
Yunnan University, Yunnan, P.R. China
*Corresponding author, zhouxb@ynu.edu.cn

Abstract

We participated in the BioNLP 2019 Open Shared Tasks: binary relation extraction of SeeDev task. The model was constructed using convolutional neural networks (CNN) and long short term memory networks (LSTM). The full text information and context information were collected using the advantages of CNN and LSTM. The model consisted of two main modules: distributed semantic representation construction, such as word embedding, distance embedding and entity type embedding; and CNN-LSTM model. The F1 value of our participated task on the test data set of all types was 0.342. We achieved the second highest in the task. The results showed that our proposed method performed effectively in the binary relation extraction.

1 Introduction

The goal of Information Extraction (IE) (Finkel et al., 2005) is to transform textual information into structured information, and to focus on quickly locating and finding useful information in large amounts of data. Information Extraction (IE) (Fader et al., 2011) is also capable of mining useful data and hiding knowledge from a large number of corpus texts, which has led to some new research methods in many disciplines. For example, with the growing demand for key issues related to life and biology, many biological problems have fallen into the bottleneck due to inadequate methods. Biological information extraction (Bio-IE) emerges in time and attracts more and more researchers to solve problems. For instance, in the identification of named entities, the classification of relationships between proteins and the extraction of links between drugs. In addition, information extraction in the field of biology, especially event extraction, has entered people's views. This will be a far-reaching task and a major biological challenge for information extraction tasks.

The BioNLP Shared Task Series is a representative of biomolecular event extraction and has been held four times. This year is the fifth time that BioNLP has shared tasks. The topics in this series include fine-grained extraction, generalization to knowledge base construction. In addition, the scope of this task has become more extensive in each time. For instance, the BioNLP 2016 Shared Task(Nédellec et al., 2016) contained three separate parts, the Bacteria Biotope subtask (BB3), the Seed Development subtask (SeeDev) and the Genia Event subtask (GE4). However, the BioNLP 2019 Open Shared Task contains seven separate parts, the Integrated structure, semantics and coreference subtask (CRAFT), the PharmaCoNER task, the Active Gene Annotation Corpus subtask (AGAC), the BB3, the SeeDev and the Research Domain Criteria subtask(RDoc).

We mainly participated in the binary relation extraction task, which is part of the SeeDev task. The SeeDev task (Nédellec et al., 2013)(Chaix et al., 2016) aims to promote complex event extraction on regulations in plants from scientific articles. It focuses on events describing genetic and molecular mechanisms involved in seed development of the model plant, Arabidopsis thaliana. It involves n-ary and binary relation extraction. Meanwhile, the SeeDev task was proposed for the first time at BioNLP Shared Task 2016(Nédellec et al., 2016) (Mehryary et al., 2016). This 2019 edition is a rerun of the task, with an evaluation methodology more focused on the biological contribution.

Many teams participated in the BioNLP 2016 Shared Task(He et al., 2016). For example, VERSE uses a support vector machine (SVM) and k-fold cross-validation to identify the best parameters.(Lever and Jones, 2016) DUTIR uses a deep learning method that utilizes a convolutional neu-

Proceedings of the 5th Workshop on BioNLP Open Shared Tasks, pages 110–114
Hong Kong, China, November 4, 2019. ©2019 Association for Computational Linguistics

ral network(Li et al., 2016). Motivated by the previous study, based on CNN, we have integrated LSTM(Hochreiter and Schmidhuber, 1997) to solve the defect that convolutional neural networks can not obtain context information. After improving the method, we got good results.

The rest of our paper is structured as follows. Section 2 introduces models. Section 3 describes results and discussion. Conclusions are described in Section 4.

2 Model

The SeeDev-binary task can be thought of as a binary relationship extraction, which specifies whether there is interaction between the two entities. In relation extraction, the semantic and syntactic information of a sentence plays an important role. Traditional methods often require the design and extraction of complex features based on domain-specific knowledge (such as tree kernels and graphics kernels) to construct the model. As a result, this results in a much lower corpus-dependent generation capability. Therefore, we use CNN to replace complex manual design feature engineering, and learn the advanced function automation by modeling the word embedding and fully connected neural networks from the original input through convolution and pooling operations. Besides, we capture relative distance information and entity types as complementary features of the sentence. After that, we input the data processed by the CNN into the LSTM. Because CNN do not get good context information, and sometimes the connection between text contexts can help us do relation extraction more accurately. So, LSTM can get text context information, which allows us to get a better result in the end.

As shown in Figure 1, the model consists of two modules: distributed semantic representation construction, such as embedded characters, distance embedding and entity type embedding, and CNN-LSTM module. In the next section, we will introduce more details.

2.1 Data preprocessing

When doing data preprocessing, first we use the Stanford CoreNLP(Manning et al., 2014) tool to process the task's data. The text is divided into sentences and tokenized. Parts-of-speech and lemmas are identified and a dependency parse is generated for each sentence. Then, we further process the preprocessed data.

2.2 Embedding

We use the context of two entities to predict the type of relationship. In our task, the context is represented by words between two entities in a sentence. Then, by analyzing the data, we observe that different entities with different types have different mutual interaction probabilities if the entity types satisfy the relationship constraints. Therefore, the entity type of the two entities is the important factor of the predicted relationship type. In our model, entity types are seen as a complement to word embedding. In addition, we find that distance information usually plays an important role. The distance can capture the relative position between two entities. So, we concatenate the word embedding(Levy and Goldberg, 2014), type embedding(Su and Wang, 2011), and distance embedding(Cormode, 2003). We use the pre-trained word embedding.[1]

Then, we would introduce some formulas about word embedding, entity type embedding and distance embedding.

$$LT_W(S) =$$

$$[< W >_{E_1}, < W >_{W_1}, ..., < W >_{W_n}, < W >_{E_2}]$$

$$LT_{W,W^T}(S) =$$

$$[< W >_{E_1}, ..., < W >_{E_2}, < W^T >_{type(E_1)}]$$

$$LT_{W^d}(S) =$$

$$[< W^d >_{d(E_1,E_1)}, ..., < W^d >_{d(E_2,E_1)}, 0, 0]$$

where S stands for the sentences. E_1 and E_2 are the type 1 and type 2 respectively. W_1 stands for the first word. W is the word embedding table. W^T is type embedding table and W^d stands for the distance embedding table. $LT_W(S)$ is the representation of word embedding. $LT_{W,W^T}(S)$ is the representation type embedding. $LT_{W^d}(S)$ is the distance embedding. In the distance embedding, zero vector(0) is used to pad the sentence.

[1] https://github.com/cambridgeltl/BioNLP-2016

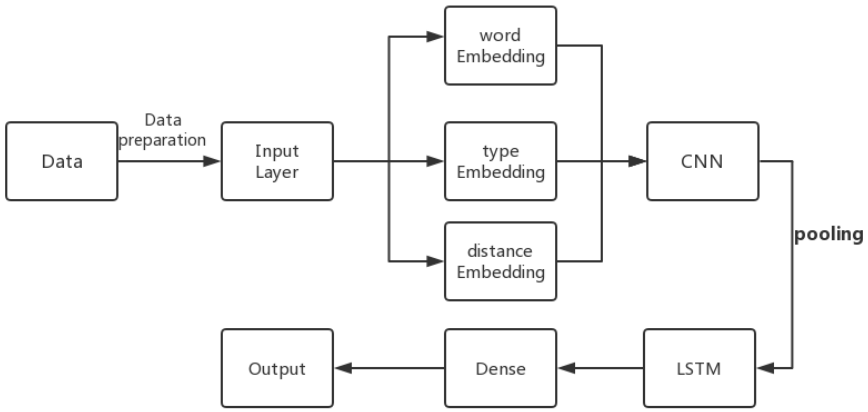

Figure 1: Our proposed CNN-LSTM based model

model	dropout	batch	epoch	F1
CNN	0.5	64	120	0.52
CNN-LSTM	0.5	64	120	0.60

Table 1: The F1 score of CNN and CNN-LSTM on the dev data set for SeeDev-binary task

2.3 Model training

We run our model 5 times and use the maximum as the final result of the model. In all model runs, the dropout(Srivastava et al., 2014) is set to 0.5. We found that our loss function tends to stabilize when the epoch reaches around 120. So, we think that our model can converge at this time, so set epoch = 120. The batch size is set to 64. And, we use a pooling approach that combines average pooling and max pooling.

In this task, we choose the CNN-LSTM model to compare with a single CNN model. We find that the CNN-LSTM model works better than a single CNN model on development data set. So, we choose the CNN-LSTM model in the final submission.

3 Results and discussion

The SeeDev-binary task data sets consist of three parts which are the training set, the development set, and the test set. There are a total of 87 sections from 20 complete articles on Arabidopsis seed de-

Cluster	F1	Recall	Precision
Comparison	0.5	0.6	0.43
Function	0.25	0.19	0.35
Regulation	0.34	0.47	0.27
Genic Regulation	0.23	0.24	0.22
Composition	0.35	0.57	0.25
Interaction	0.22	0.16	0.33

Table 2: The F1, recall and precision of cluster on the test data set for SeeDev-binary task

Team	F1	Recall	Precision
MIC-CIS-1	0.373	0.511	0.295
YNU-junyi	**0.342**	**0.458**	**0.273**
Yunnan...	0.067	0.133	0.045
YNUBY	0.019	0.070	0.011

Table 3: The result of all types on the test data set for SeeDev-binary task

Team	F1	Recall	Precision
MIC-CIS-1	0.443	0.606	0.349
YNU-junyi	**0.394**	**0.528**	**0.314**
Yunnan...	0.135	0.267	0.090
YNUBY	0.074	0.274	0.043

Table 4: The result of ignoring types on the test data set for SeeDev-binary task

Binary relation type	F1	Recall	Precision
Binds_To	0.31	0.28	0.35
Composes_Primary_Structure	0.34	0.44	0.28
Composes_Protein_Complex	0	0	0
Exists_At_Stage	0.14	0.1	0.25
Exists_In_Genotype	0.42	0.64	0.31
Interacts_With	0.09	0.06	0.19
Is_Involved_In_Process	0	0	0
Is_Localized_In	0.27	0.52	0.18
Is_Member_Of_Family	0.35	0.62	0.25
Is_Protein_Domain_Of	0.25	0.39	0.18
Occurs_In_Genotype	0.17	0.14	0.2
Occurs_During	0	0	0
Regulates_Accumulation	0.17	0.19	0.15
Regulates_Development_Phase	0.23	0.34	0.17
Regulates_Expression	0.22	0.25	0.19
Regulates_Molecule_Activity	0	0	0
Regulates_Process	0.43	0.66	0.32
Regulates_Tissue_Development	0	0	0
Transcribes_Or_Translates_To	0.34	0.38	0.32
Is_Linked_To	0.15	0.1	0.33
Is_Functionally_Equivalent_To	0.64	0.57	0.74
Has_Sequence_Identical_To	0.56	0.77	0.44

Table 5: Detailed results of our method on the test data set for SeeDev-binary task

velopment. This task defines 16 different types of entities and 22 different types of binary relationships.

Our method obtained F1 scores of 0.342 for all types and 0.394 for ignoring relation types and direction on the test set. In this task, the organizer gives the results of the evaluation obtained from three different evaluation conditions. Compared with 2016 BioNLP Shared Task, the organizer has added two more evaluations in order to have better biological contributions. These evaluation conditions are global results, relations by type cluster, and ignoring relation types and direction, respectively. We obtained a good score compared to the official results from different systems, and we ranked the second among all teams. It proves that our proposed method has good performance in binary relation extraction.

Table 2 shows the F1, recall and precision of cluster on the test data sets, and Table 3 shows the result of all types on the test data sets. Table 4 shows the result of ignoring types on the test data sets and Table 5 shows detailed results of our method on the test data set.

4 Conclusions

We use distributed semantic representation and CNN-LSTM model to extract the binary relationship between entities, then build a word embedding with rich semantic knowledge, distance embedding and entity type embedding to feed it into the CNN and learn the intrinsic relationship between the candidate entities. In the task, our F1-score of all types is 0.342, which indicates that our proposed method works efficiently in extraction of binary relations.

However, using only the original words embedded in CNN-LSTM may not be sufficient to understand the hidden information between words. Using our model to get this score does not mean that the model works well in other tasks.

In the future, we will continue to focus more on building rich distributed semantic embedding and we will improve our model by changing our model structure and adjusting paraments. In addition, we will explore various neural networks with multilayer architectures, such as the attention mechanism and capsule networks, to solve binary relationships or event extraction problems.

References

Estelle Chaix, Bertrand Dubreucq, Abdelhak Fatihi, Dialekti Valsamou, Robert Bossy, Mouhamadou Ba, Louise Deléger, Pierre Zweigenbaum, Philippe Bessières, Loïc Lepiniec, and Claire Nedellec. 2016. Overview of the regulatory network of plant seed development (seedev) task at the bionlp shared task 2016. In *Proceedings of the 4th BioNLP Shared Task Workshop, BioNLP 2016, Berlin, Germany, August 13, 2016*, pages 1–11.

Graham Cormode. 2003. *Sequence distance embeddings*. Ph.D. thesis, University of Warwick.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1535–1545. Association for Computational Linguistics.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics.

Xinyu He, Lishuang Li, Jieqiong Zheng, and Meiyue Qin. 2016. Extracting biomedical event using feature selection and word representation. In *Proceedings of the 4th BioNLP Shared Task Workshop*, page 101, Berlin, Germany. Association for Computational Linguistics.

Sepp Hochreiter and Jrgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Jake Lever and Steven JM Jones. 2016. VERSE: Event and relation extraction in the BioNLP 2016 shared task. In *Proceedings of the 4th BioNLP Shared Task Workshop*, pages 42–49, Berlin, Germany. Association for Computational Linguistics.

Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*, pages 2177–2185.

Honglei Li, Jianhai Zhang, Jian Wang, Hongfei Lin, and Zhihao Yang. 2016. DUTIR in BioNLP-ST 2016: Utilizing convolutional network and distributed representation to extract complicate relations. In *Proceedings of the 4th BioNLP Shared Task Workshop*, pages 93–100, Berlin, Germany. Association for Computational Linguistics.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Farrokh Mehryary, Jari Björne, Sampo Pyysalo, Tapio Salakoski, and Filip Ginter. 2016. Deep learning with minimal training data: TurkuNLP entry in the BioNLP shared task 2016. In *Proceedings of the 4th BioNLP Shared Task Workshop*, pages 73–81, Berlin, Germany. Association for Computational Linguistics.

Claire Nédellec, Robert Bossy, and Jin-Dong Kim. 2016. Proceedings of the 4th bionlp shared task workshop. In *Proceedings of the 4th BioNLP Shared Task Workshop*.

Claire Nédellec, Robert Bossy, Jin-Dong Kim, Jung-Jae Kim, Tomoko Ohta, Sampo Pyysalo, and Pierre Zweigenbaum. 2013. Overview of bionlp shared task 2013. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 1–7.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Jiabao Su and Zhi-Qiang Wang. 2011. Sobolev type embedding and quasilinear elliptic equations with radial potentials. *Journal of Differential Equations*, 250(1):223–242.