UC3M-NII Team at SemEval-2018 Task 7: Semantic Relation Classification in Scientific Papers via Convolutional Neural Network

Víctor Suárez-Paniagua, Isabel Segura-Bedmar
Computer Science Department
Universidad Carlos III de Madrid
Leganés 28911, Madrid, Spain
vspaniag,isegura@inf.uc3m.es

Akiko Aizawa
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo, Japan
aizawa@nii.ac.jp

Abstract

This paper reports our participation for SemEval-2018 Task 7 on extraction and classification of relationships between entities in scientific papers. Our approach is based on the use of a Convolutional Neural Network (CNN) trained on 350 abstract with manually annotated entities and relations. Our hypothesis is that this deep learning model can be applied to extract and classify relations between entities for scientific papers at the same time. We use the Part-of-Speech and the distances to the target entities as part of the embedding for each word and we blind all the entities by marker names. In addition, we use sampling techniques to overcome the imbalance issues of this dataset. Our architecture obtained an F1-score of 35.4% for the relation extraction task and 18.5% for the relation classification task with a basic configuration of the one step CNN.

1 Introduction

Nowadays, there is a high increase in the publication of scientific articles every year, which demonstrates that we are living in an emerging knowledge era. Experts cannot deal with this explosion of information and it is very hard to be up to date about the state-of-the-art techniques in a given field. This arduous task could be reduced if we automatically identify concepts from scientific articles and recognize the semantic relations between them with Natural Language Processing (NLP) techniques.

The Semantic Relation Extraction and Classification in Scientific Papers task at SemEval-2018 task 7 (Gábor et al., 2018) provides a framework for measuring the automatic annotation performance by models which are trained on scientific publications abstracts. The task defines six categories of relations between concepts and two tasks are proposed: (1) the classification of the relations between two entities in the predefined categories, which is divided in two scenarios according to the data used: clean or noisy; and (2) the extraction of the relations given the entities from the clean data, which also could involve their subsequent classification.

In this paper, we describe our participation for SemEval-2018 Task 7 on the extraction of relationships between entities in scientific papers and also the subsequent classification in the predefined classes of this relations with one step classifier. The model is based on the Convolutional Neural Network (CNN) proposed in (Kim, 2014), which was the first work to exploit this architecture for the task of sentence classification. CNN is a robust deep-learning architecture which has exhibited good performance in others NLP tasks such as semantic clustering (Wang et al., 2016), sentiment analysis (Dos Santos and Gatti, 2014) and event detection (Nguyen and Grishman, 2015). The model uses as the input of each instance the transformation into real value vectors of the words of the sentence, the distances to the target entities of each word and the Part-of-Speech types. Furthermore, we carry out a sampling technique to alleviate the imbalance issues of the dataset equalizing the number of the instances for all the classes.

2 Dataset

An annotated corpus for training and testing the participating systems was provided in the SemEval-2018 Task 7. The dataset contains 350 and 150 abstract from scientific articles for training and testing set, respectively.

The relation instances are divided into the following classes: *USAGE*, *RESULT*, *MODEL*, *PART WHOLE*, *TOPIC* and *COMPARISON*. All of them are asymmetrical except *COMPARISON*, where

Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), pages 793–797
New Orleans, Louisiana, June 5–6, 2018. ©2018 Association for Computational Linguistics

both entities are involved in the same bidirectional relation. A detailed description and analysis of the corpus and its methodology used to collect and process the scientific abstracts can be found in (Gábor et al., 2018).

2.1 Pre-processing phase

The relations between scientific concepts are annotated pair by pair in the abstracts. All annotated relations span within one sentence, thus, we split the paragraphs of the abstracts into sentences with NLTK tool[1] to generate all the possible instances in the corpus.

After that, each instance was tokenized, all words were converted to lower-case and special character were removed in order to clean the sentences as the approach described in (Kim, 2014). In addition, we used entity blinding for each relation to generalize the model, in which the two target entities of the relations were replaced by entity markers as "*entity1*" and "*entity2*", and "*entity0*" for the remaining entities. Since relations can be asymmetrical, we considered both directions. In other words, for each pair of candidates entities, we generated two different instances. For the *COMPARISON* class, which is a bidirectional relationship, we annotated both instances with the same class label. For example, the sentence: *'We suggest a method that mimics the behaviour of the oracle using a neural network or a decision tree.'* should be transformed to the relation instances showed in Table 1.

Instances after entity blinding (*entity1*, *entity2*)
(oracle, neural network) *'We suggest a method that mimics the behaviour of the entity1 using a entity2 or a entity0.'*
(neural network, oracle) *'We suggest a method that mimics the behaviour of the entity2 using a entity1 or a entity0.'*
(oracle, decision tree) *'We suggest a method that mimics the behaviour of the entity1 using a entity0 or a entity2.'*
(decision tree, oracle) *'We suggest a method that mimics the behaviour of the entity2 using a entity0 or a entity1.'*
(neural network, decision tree) *'We suggest a method that mimics the behaviour of the entity0 using a entity1 or a entity2.'*
(decision tree, neural network) *'We suggest a method that mimics the behaviour of the entity0 using a entity2 or a entity1.'*

Table 1: Instances of a sentence in the corpus after applying the pre-processing phase with entity blinding.

[1]http://www.nltk.org

Table 2 shows the number of the instances extracted in the training set per each class. The *None* class represents the number of pairs of entities that are not related (negative instances). The number of positive instances is very low compared to the negative ones, 1323 over 19210 (around 7%), mainly because most classes are unidirectional and we annotated the reverse instance as *None*.

We followed a similar sampling technique described in (Wang et al., 2017) to adjust the same numbers of instances per each class. Therefore, we randomly discard 60% of the negative instances and we duplicate the instances in each class until having the same number as the more representative class, 483 corresponding to *USAGE*. Thus, we try to solve possible issues associated with the imbalanced dataset.

Classes	Instances
COMPARE	190
MODEL-FEATURE	326
PART WHOLE	234
RESULT	72
TOPIC	18
USAGE	483
None	17887
Total	19210

Table 2: Number of instances in the dataset.

3 Method

In this section, we present a CNN model to detect and classify relationships between scientific concepts. Figure 1 shows the whole process from its input, which is a sentence with blinded entities, until the output, which is the classification of the instance into one of the relation types defined by the task.

3.1 Word table layer

Firstly, we determined n as the maximum sentence length in the training dataset. Those sentences with lengths shorter than n are padded with an auxiliary token "*0*". After that, we assigned a randomly initialized vector for each different word, creating thus a word embedding matrix: $\mathbf{W}_e \in \mathbb{R}^{|V| \times m_e}$ where V is the vocabulary size and m_e is the word embedding dimension. Finally, we obtained a matrix $\mathbf{x} = [x_1; x_2; ...; x_n]$ for each instance where the words are represented by their corresponding word embedding vectors.

In addition, we used the word position embedding described in (Zeng et al., 2014), which

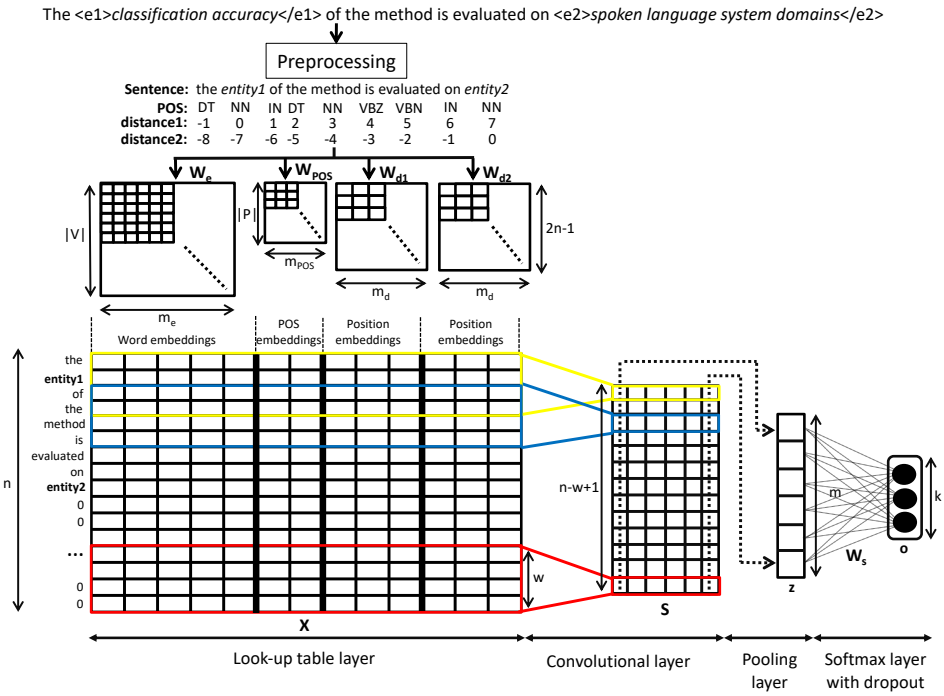

Figure 1: CNN model for the semantic relation classification in scientific papers of SemEval-2018 Task 7.

maps the distances of each word with respect to the two candidate entities into a real value vector using two position embedding matrices $\mathbf{W}_{d1} \in \mathbb{R}^{(2n-1)\times m_d}$ and $\mathbf{W}_{d2} \in \mathbb{R}^{(2n-1)\times m_d}$ where m_d is the position embedding dimension. Moreover, we extracted the Part-of-Speech (POS) feature of each word (entities are marked as common nouns) and create a POS embedding matrix as (Zhao et al., 2016) $\mathbf{W}_{POS} \in \mathbb{R}^{|P|\times m_{POS}}$ where P is the POS types vocabulary size and m_{POS} is the POS embedding dimension.

Finally, we created an input matrix $\mathbf{X} \in \mathbb{R}^{n\times(m_e+m_{POS}+2m_d)}$ which is represented by the concatenation of the word embedding, the POS embedding and the two position embeddings for each word in the instance.

3.2 Convolutional layer

Once we obtained the input matrix, we applied the convolutional operation with a context window of size w to create higher level features. For each filter in $\mathbf{f} = [f_1; f_2; ...; f_w]$, we created a score matrix for the whole sentence as

$$s_i = g(\sum_{j=1}^{w} f_j x_{i+j-1}^T + b)$$

where b is a bias term and g is a non-linear function (such as tangent or sigmoid) of m number of filters.

3.3 Pooling layer

We extracted the most relevant features of each filter using the max function, which produces a single value in each filter as $z_f = max\{\mathbf{s}\} = max\{s_1; s_2; ...; s_{n-w+1}\}$. Thus, we created a vector $\mathbf{z} = [z_1, z_2, ..., z_m]$, whose dimension is the total number of filters m representing the relation instance. In the end, we concatenated the output values of the different filters in this layer.

3.4 Softmax layer

In this layer, we performed a dropout to prevent over-fitting obtaining a reduced vector $\mathbf{z}_d$ randomly dropping elements in $\mathbf{z}$. After that, we fed this vector into a fully connected softmax layer with weights $\mathbf{W}_s \in \mathbb{R}^{m\times k}$ to compute the output prediction values for the classification as

$$\mathbf{o} = \mathbf{z}_d\mathbf{W}_s + d$$

where d is a bias term. At test time, the vector $\mathbf{z}$ of a new instance is directly classified by the softmax layer without a dropout.

3.5 Learning

We defined the CNN parameter set to be learned in the training phase as θ = ($\mathbf{W}_e$, $\mathbf{W}_{POS}$, $\mathbf{W}_{d1}$, $\mathbf{W}_{d2}$, $\mathbf{W}_s$, $\mathbf{F}_m$), where $\mathbf{F}_m$ are all of the m filters $\mathbf{f}$. For this purpose, we used the conditional probability

of a relation r obtained by the softmax operation as

$$p(r|\mathbf{x},\theta) = \frac{\exp(\mathbf{o}_r)}{\sum_{l=1}^{k}\exp(\mathbf{o}_l)}$$

to minimize the cross-entropy function for all instances ($\mathbf{x}_i$,y_i) in the training set T as follows

$$J(\theta) = \sum_{i=1}^{T} \log p(y_i|\mathbf{x}_i,\theta)$$

In addition, we minimized the objective function by using stochastic gradient descent over shuffled mini-batches and the Adam update rule (Kingma and Ba, 2014) to learn the parameters.

4 Results and Discussion

We define the CNN parameters for the experiments using the values described in Table 3. The number of epochs was fine-tuned in the validation set using the stopping criteria.

Parameter	Value
Maximal length in the dataset , n	152
Word embeddings dimension, M_e	300
POS embeddings dimension, M_{POS}	10
Position embeddings dimension, M_d	5
Filters for each window size, m	200
Filter sizes, w	(3, 4, 5)
Dropout rate, p	50%
Mini-batch size	50
Non-linear function, g	ReLU

Table 3: The CNN model parameters and their values used for the results.

Our CNN system obtained an F1-score of 35.4% for the relation extraction task in which only the detection of relation is taken into consideration. The official results obtained for the relation classification task are showed in Table 4. Our model reaches an F1-score in Macro-average of 18.5% with one step classifier, which means that the extraction and classification are considered at the same time. This performance was expected because we reached the similar results with a validation set created from the training set. Furthermore, we correctly predicted 147 instances with correct directionality over 367 (i.e. 40.05% in coverage).

The main problem is the high number of FP in the majority of classes, which are the *None* instances classified as a class. In some classes such as *PART WHOLE* and *USAGE* we have also a high number of FN compared to the total number of instances. We consider that the main reason is that the representation of the two directions of each relation is very similar, only the position distances and the target entity names are inverted, and the CNN cannot distinguish between them.

Classes	TP	FP	FN	P	R	F1
COMPARE	8	116	11	6.45%	42.11%	11.19%
MODEL-FEATURE	36	185	37	16.29%	49.32%	24.49%
PART WHOLE	22	66	60	25%	26.83%	25.88%
RESULT	2	21	14	8.7%	12.5%	10.26%
TOPIC	0	0	3	0%	0%	0%
USAGE	41	96	133	29.93%	23.56%	26.37%
Micro-averaged	-	-	-	18.38%	29.7%	22.71%
Macro-averaged	-	-	-	14.39%	25.72%	18.46%

Table 4: Results over the dataset using a CNN model measured by True Positives, False Positives, False Negatives, Precision, Recall and F1-measure, respectively.

5 Conclusions and Future work

A CNN model is used for the Relation Classification task of SemEval 2018 by UC3M-NII Team. Moreover, we balanced the dataset using sampling techniques, blinded the entities in the sentence and aggregated position embedding and POS embedding to the word embedding of each word to have more representation of each instance. This architecture obtained an F1-score of 35.4% and 18.5% for the relation extraction and classification task, respectively.

As future work, we proposed to use a two steps model to overcome the extraction of the relationships between two concepts and subsequently classify them in the different semantic classes. In addition, we also plan to rule out the reverse instances of each class as *None* in order to avoid having very similar representation with different labels. We plan to tackle the directionality problem with post-processing rules after the classification. Furthermore, we will train a CNN with different pre-trained word embedding models instead of using a random initialization.

Funding

This work was supported by the Research Program of the Ministry of Economy and Competitiveness - Government of Spain, (DeepEMR project TIN2017-87548-C2-1-R) and the TEAM project (Erasmus Mundus Action 2-Strand 2 Programme) funded by the European Commission.

Acknowledgments

We would like to thank the members of the Aizawa Laboratory and the HULAT research group for their fruitful discussions which were held.

References

C.N. Dos Santos and M. Gatti. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of the 25th International Conference on Computational Linguistics, (COLING 2014), Technical Papers*, pages 69–78.

Kata Gábor, Davide Buscaldi, Anne-Kathrin Schumann, Behrang QasemiZadeh, Haïfa Zargayouna, and Thierry Charnois. 2018. Semeval-2018 Task 7: Semantic relation extraction and classification in scientific papers. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA.

Y. Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In *ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference*, volume 2, pages 365–371. Association for Computational Linguistics (ACL).

Peng Wang, Bo Xu, Jiaming Xu, Guanhua Tian, Cheng-Lin Liu, and Hongwei Hao. 2016. Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. *Neurocomputing*, 174, Part B:806 – 814.

Wei Wang, Xi Yang, Canqun Yang, Xiaowei Guo, Xiang Zhang, and Chengkun Wu. 2017. Dependency-based long short term memory network for drug-drug interaction extraction. *BMC Bioinformatics*, 18(16):578.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014), Technical Papers*, pages 2335–2344, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Zhehuan Zhao, Zhihao Yang, Ling Luo, Hongfei Lin, and Jian Wang. 2016. Drug drug interaction extraction from biomedical literature using syntax convolutional neural network. *Bioinformatics*.