Joint Extract Method from Scholarly Papers

Jianfan Ge, Ting Jiang

Nanjing University Of Finance & Economics, NanJing, China

Abstract

Entities and relations concisely reflect important information related to the subject matter of the literature, which is essential for understanding and analyzing it. In scientific research, methods are indispensable tools and important research objects for solving scientific problems (methods include tasks, discipline-specific methods, models, algorithms, and metrics, etc.). Therefore, 'method' is indispensable for the understanding and analysis of academic literature. This paper aims to extract method-like entities and relations from scientific abstracts using a semantically enhanced deep learning model. We explored the impact of linguistic information on the entity and relation extraction task, and to this end, we added additional POS tag information to the word vectors obtained through the pre-trained model to highlight POS tag information, which proved to be superior to the pre-trained word vectors alone. Individually, in the entity recognition part, the token sequence length of entities is considered as the feature, and in the relationship extraction part, performing max pooling over the context between entity candidates has been proven better than full context, additionally, the distance between entity candidates is embedded by us as an additional feature. Entity type is also entered as an additional feature. The sequences of rich token representations constitute a span, over which entities and relations are learned jointly. The results on several datasets show that the embedding of rich semantic information outperforms the original span-based model.

Keywords

entity extraction, relation extraction, method,POS tag,entity distance,token length,entity type

1. Introduction

In the era when the data was dense, a large number of papers are published daily[1-3]. For most academic researchers, considering the diversity and explosive growth of research in the field, the speed of reading is much slower than the speed of publication, making it impossible to always access the latest methods from the most recent literature, which means that traditional manual searches to find scientific methods become challenging[4]. Therefore, in order to help scholars form a methodological system for their research directions and obtain the most cutting-edge methods related to the research content while greatly saving the labor and time of researchers, it is essential to study the method to extract methods from large-scale academic literature. As is well known, entity recognition (ER) and relationship extraction (RE) are essential and challenging tasks in natural language processing (NLP), which can be beneficial to information extraction from academic literature. Entity recognition and relationship extraction from academic texts refer to identifying academic entities, such as Task, Method, Metric, and so on, and extracting semantic relations among these entities, e.g Evaluate-For, Used-For, and so on. ER and RE from literature are used in a wide range of academic applications, including academic information retrieval, knowledge graph construction, question answering, article recommendation, etc. The joint model of ER and RE from scholarly texts aims to extract the entity-relationship-entity tuples. For example, the following sentence S1 shown in Figure 1 contains two entities; we delineate the mention with square brackets and the corresponding entity types with suffixes：

AHPCAI2022@2nd International Conference on Algorithms, High Performance Computing and Artificial Intelligence
EMAIL: gejianfan189@gmail.com (Jianfan Ge), * Corresponding author:1259850279@qq.com (Ting Jiang)

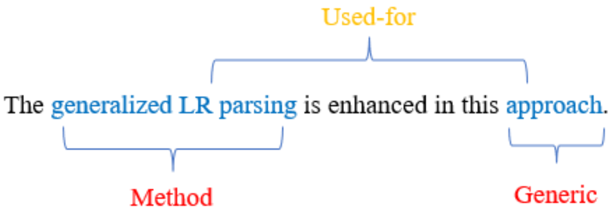

Fig.1 The example sentence from the SciERC dataset. Generalized LR parsing and approach are entities, which are methods and generic types, respectively. The relation (or relation type) that points from generalized LR parsing to approach is Used-for.

Previous research has shown that method entities identified by complex rules are more responsive to searchers' needs than those identified by matching terms in academic literature[5]. As a result, researchers have proposed more complex rules to accomplish the extraction task, including cue words, language patterns, lexicality, word position, etc[6-10]. duck et al.[11] created a named entity recognizer used in bioinformatics called bioNerDS to extract software entities and dataset entities from papers. Noun phrases from academic papers were extracted and scored based on different rules. In the first round, candidate entities are checked to see if they appear in the generated dictionary. In the second round, strong rules extracted from the article, such as version information, references and URLs, are classified into positive and negative rules and assigned different scores. In the last round, some clues, such as specific verbs, and indicative but ambiguous titles, are combined into weak clues and assigned scores. Candidate entities were scored according to their compliance with the rules and judged as method entities based on their final scores.

In this paper, we propose a semantically enhanced term entity relation extraction model to jointly extract method entities and relationships from the abstracts of scientific papers. We use a recent model called SPERT as the baseline, which uses a pre-trained transformer. A shallow entity classifier and a shallow relationship classifier are applied to extract entities and relations, respectively. The transformer generates embeddings of tokens in the abstract, and merges the embeddings of a span of tokens into one. Many natural language processing tasks benefit from the use of linguistic information, such as part-of-speech tagging, but they are less explored in deep neural models of NER and RE. Petasis et al.[12] believed that named entities were proper nouns (PN), which served as the name of someone or something. From the perspective of ontology, Alfonseca and Manandhar[13] proposed that named entities were objects used to solve specific problems. Borrega et al.[14] defined named entities in detail from the perspective of linguistics, stipulating that only nouns and noun phrases can be used as named entities. Although these definitions are not uniform, it can be sure that named entities at least the vast majority are nouns or noun phrases. The relationships between entity pairs also seem to be related to entity types, for example, we often find "used-for" relations between "generic" entities and "method" entities. In relation extraction, contextual information between candidate entities has been proven to be superior to global contextual information. And in addition to the important information brought by the semantics itself, the distance information between candidate entities should also be an important feature. For this purpose, we counted the distance between relational entities, as shown in Table 1. Through the data, we can find that Conjunction and Feature-of method types basically appear in close entities while other relationship types are more average. By counting the percentage of different relationships appearing before and after the entity type, we found that the entity type characteristics also have an important influence on the relationship, here we take a table as an example, as shown in Table 2. Therefore, we propose a semantically enhanced model. The main contributions of our work are as follows:

- We propose an improved joint model of NER and RE for academic texts, in which the type of entity and the distance between candidate entities that potentially constitute the relationship are considered in RE, and the token sequence length of an entity is considered in NER.
- We enrich the initial embeddings, the initial embeddings are augmented by semantic information and syntactic information.
- Experiments on real data sets validate the effectiveness of our proposed method.

2. Related works

With the development of technology, deep learning methods are becoming the focus of research in the field of machine learning. Deep learning models focus more on the capability of machines and abandon complex feature engineering, which greatly reduces the labor as well as time cost required for extracting feature engineering compared to statistical models of machine learning. Therefore, deep learning has become a very important research direction.

The supervised deep learning method used in relation extraction can solve the main problems of manual feature extraction and error propagation that exist in classical methods. The low-level features are combined to form more abstract high-level features. At present, supervised relation extraction methods mainly include pipeline approaches and joint approaches.

2.1 Pipeline approaches

Pipeline approaches refer to the extraction of relations between entities directly based on the entity recognition already done. The early pipeline approaches mainly used two types of structures, convolutional neural networks (CNNs) and recurrent neural networks (RNNs). Among them, CNNs with diverse convolutional kernels are good for recognizing structural features of the target, while RNNs can fully consider the dependency between long-range words and their memory function is good for recognizing sequences. Zeng et al.[15] used CNN to extract word-level and sentence-level features for the first time, and improved the accuracy of the relationship extraction model by using the hidden layer and softmax layer for relationship classification. Socher et al.[16] first used an RNN approach for entity relation extraction, which uses recurrent neural networks to syntactically parse the sentences in the annotated text and then obtains a vector representation considering the syntactic structure of the sentences after continuous iterations.

As the research progresses, CNN and RNN methods are continuously improved and refined, and many variants are generated, such as long short-term memory (LSTM), and bidirectional long short-term memory (Bi- LSTM), which are improved to solve the gradient disappearance. Xu et al.[18] proposed an LSTM-based relation extraction method based on the shortest path of syntactic dependency analysis tree, incorporating features such as word vector, part of speech tags, WordNet, and syntax, using maximum pooling layer, softmax layer, etc. In addition, with the application of graph convolutional network (GCN) in the field of natural language processing, GCN has been increasingly used for mining and exploiting potential information between entities, providing new ideas for solving relation overlap and entity overlap, and thus further promoting the development of relation extraction. Schlichtkrull et al.[19] proposed the use of relational graph convolutional neural networks (R-GCNs) on two standard knowledge bases to accomplish link prediction and entity classification, respectively, where link prediction extracts missing relations and entity classification completes the missing attributes of entities; Zhang et al.[20] proposed an extended graph convolutional neural network that can effectively handle arbitrary dependency structures in parallel and facilitate the extraction of entity relationships to effectively utilize negative class data.

Although the Pipeline approach is easy to implement, the entity model and the relation model can use independent datasets and do not need to label both entity and relation datasets, there are several disadvantages:

a) error accumulation: errors in entity extraction will affect the performance of the next step of relation extraction.

Table 1: Relation entity distance of SciERC.

Type	Relation entity distance (world)				
	[0-3]	[4-7]	[8-11]	[>11]	Ave[>11]*4
Conjunction	84.5%	10.5%	3.3%	1.8%	0.6%
Feature-of	69.4%	24.3%	5.8%	0.6%	0.3%
Hyponym-of	46.3%	29.2%	11.7%	12.8%	3.0%
Used-for	45.2%	31.0%	12.8%	11.0%	1.1%
Part-of	48.0%	28.5%	11.7%	11.7%	2.1%
Compare	36.1%	28.3%	19.9%	15.7%	4.5%
Evaluate-for	34.2%	32.6%	15.7%	17.6%	3.1%
All 50.1%	27.8%	11.7%	10.3%	1.0%	All

Table 2: Relation type in SciERC. It shows the percentage of relations between the subject as Task type and six types of object. It is obvious that when the subject is the type of Task and the object is the type of Material, the probability that the relation is used-for is 100%. We can also find that when the subject is Task type, the relation is mainly found in Used-for and Part-of.

Type (Task)	Relation type						
	Conjunction	Feature -of	Hyponym-of	Uses-for	Part-of	Compare	Evaluate-for
Task	42.8%	0.5%	18.8%	20.7%	11.4%	2.5%	3.3%
Method	8.0%	12.0%	\	46.0%	\	\	34.0%
Metric	\	\	\	14.3%	\	\	85.7%
Material	\	\	\	100.0%	\	\	\
OtherScientificTerm	5.9%	\	\	85.3%	\	\	8.8%
Generic	\	1.8%	34.9%	26.0%	7.1%	\	30.2%

b) entity redundancy: the redundant information brought by the candidate entity pairs without relationships will enhance the error rate and increase the computational complexity, since the extracted entities are firstly paired, and then extract relations between entity pairs.

c) lacking interaction: the intrinsic connection and dependency between the two tasks are ignored.

2.2 Joint approaches

Joint approaches can further use the potential information between the two tasks to mitigate the disadvantage of error propagation. The difficulty of joint approaches is how to enhance the interaction between the entity model and the relation model. In early works, the connection method relied heavily on fine-grained feature engineering to establish the interaction between NER and RE[21-23]. Recently, end-to-end neural networks have proven successful in extracting relational triples[15,24-26], becoming the mainstream for joint entity and relation extraction.

Based on their differences in encoding task-specific features, most existing approaches can be divided into two categories: sequential encoding and parallel encoding. Sequential encoding generally encodes task features in the sequential order of NER and then RE, and this encoding approach can keep the later encoded features from affecting the first encoded features directly away, resulting in unbalanced inter-task interactions. Zeng et al.[27] and Wei et al.[28] are typical examples of this category. They extracted features for different tasks in a predefined order. Parallel encoding uses two independent encoders to generate task features, which have no interaction other than shared input, leading to insufficient inter-task interaction, and in contrast to sequential encoding, models built based on this scheme do not need to worry about the effect of encoding order. , and encoded the entity and relationship information separately, and finally completed the extraction of task-specific features in two separate submodels, respectively. Both encoding methods have their own drawbacks, the inter-task

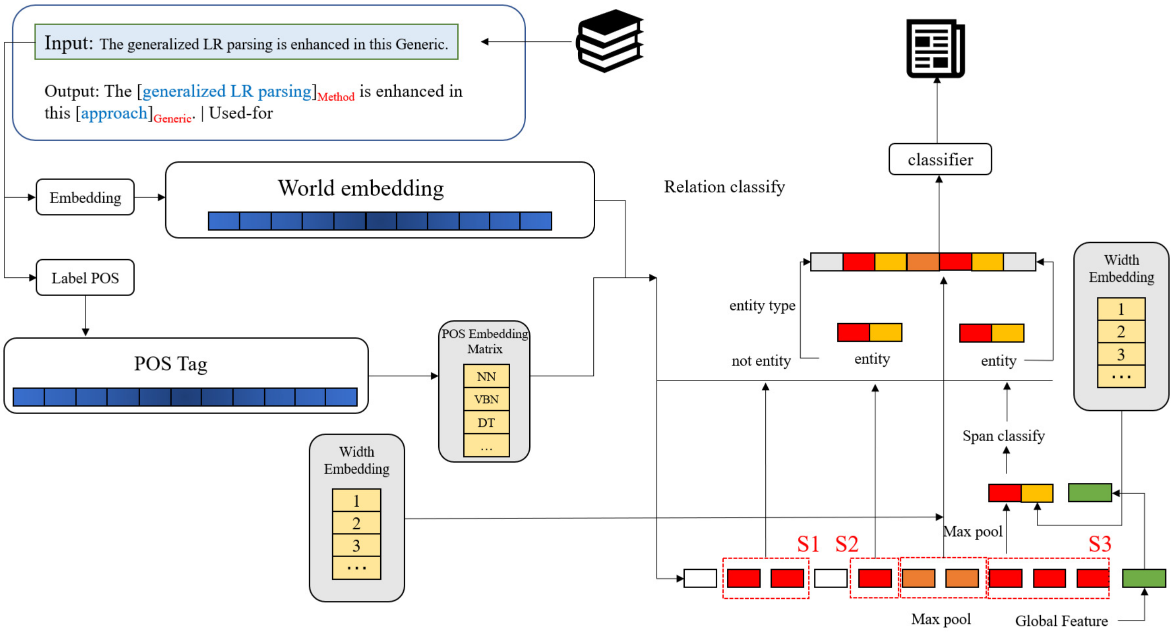

Fig.2 Framework of the model. Given a corpus of academic texts, the goal of academic entity recognition and relation extraction is to obtain entity-relationship triples. The input to the model is a list of academic texts, which are then converted into a sequence of tokens by pretrained model. The output of the tasks is shown in Figure2.

interaction, and in contrast to sequential encoding, models built based on this scheme do not need to worry about the effect of encoding order, and encoded the entity and relation information separately, and finally completed the extraction of task-specific features in two separate submodels, respectively. Both encoding methods have their own drawbacks, the inter-task interaction in sequential encoding is one-way with a specific order, while the problem with parallel encoding is that they only retain the shared features and actively ignore the features that are task-beneficial separately for each task.

3. Model

3.1 Model Architecture

In this section, we provide a detailed explanation of our model, the framework is shown in Figure 2. We develop a joint model including five components: 1) an embedding layer, which converts tokens into embedding vectors, 2) a POS encoder, which converts tokens into part-of-speech, 3)a fusion module, which fuses word embedding and part-of-speech embedding into one vector,4)a shallow entity classifier, which classifies any possible sequence of consecutive tokens and 5)a shallow relationship classifier, which classifies relation for any given set of entity pairs.

3.2 Embedding Layer

Given a sequence of sentences $S = \{s_1, s_2, \ldots, s_n\}$, the embedding layer transforms the sentences into the vector matrix, in which each token in the sentence is represented by a pre-trained embedding. The embedding of each token consists of two parts: pre-trained transformer and POS embeddings.

a) Pretrained transformer: The recognition and relationship extraction of academic terminology entities are different from the conventional entity relationship extraction, which is more specific and the relation between terms is more abstract. Traditional methods are based on manually generated features, while pre-training techniques are now widely used in deep learning, which have achieved good performance in computer vision, natural language processing, and other fields. Usually, a model that has been trained on large-scale data can achieve satisfactory performance with simple training, i.e., fine-tuning. Obtaining a high-quality initial value of parameters with the help of pre-training techniques not only reduces the training burden but also helps to improve the model generalization ability.

However, as Bert received pre-training on general texts from Wikipedia and book corpora, its performance on domain-specific tasks proved to be suboptimal in several previous works [29-30]. These empirical findings have driven the development of domain-specific pre-trained language models. For example, SciBERT in the scientific domain and BioBERT and ALBERT in the biomedical domain. so the domain-specific pre-trained model SciBERT and BioBERT was used for domain academic entities, through which academic entity features were obtained to obtain better feature representations.

We split each sentence s_i into a sequence of tokens $T_i = \{[CLS], t_1, t_2, \dots, t_m, [SEP]\}$, where [CLS] and [SEP] are special symbols. The [CLS] captures the contextual information of the text, while the [SEP] acts as a separator to separate adjacent sentences between them. We use transformer to generate pre-trained embeddings as in (1):

$$Transformer(\mathrm{T}) = \left(\mathrm{b}_{[\mathrm{CLS}]},\mathrm{b}_1,\mathrm{b}_2,\cdots,\mathrm{b}_\mathrm{n},\mathrm{b}_{[\mathrm{SEP}]}\right) \quad (1)$$

Whereb$i\epsilon R^{d_1}$, and $d1$ is the embedding dimension.

b) POS Tag: Part of speech is the important information a word carries, which reflects the components that the word plays. Dependency syntactic parse in natural language processing is the conversion of sentences into trees based on POS tags. Xu et al.[18] proposed an LSTM-based relation extraction method based on the shortest path of the syntactic dependency analysis tree, incorporating features such as word vector, part-of-speech, WordNet, and syntax, and using maximum pooling layer, softmax layer, etc. for relation classification. The addition of POS ultimately achieves the goal of effect enhancement. So we take POS tags into consideration. We generate part of speech tags for the input sentences and assign the POS tag of the parent word to each child word tag it generates. We use a directed embedding matrix to generate a embedding sequence $P_i = \{[CLS], p_1, p_2, \dots, p_m, [SEP]\}$for each positional tag of dimension d2. The BERT embedding of the token and the lexical POS embedding are then stitched together to obtain a new vector representation of dimension. We use a directed embedding matrix to generate embeddings for each positional tag of dimension d2.

$$C = \left(c_{[\mathrm{CLS}]},c_1,c_2,\cdots,c_\mathrm{n},c_{[\mathrm{SEP}]}\right) \quad (2)$$

where the output embedding is a combination of the above two vectors. The dimension of the output embedding is $d_1 + d_2$.

3.3 Span Classification

To detect entities, a vector is obtained by doing max-pooling of the embedding representation$(c_i,\cdots,c_{i+k-1})$ of a sequence of tokens of length k of each successive possible constituent entity.

$$V(s) = \mathrm{max}pool(c_i,\cdots,c_{i+k-1})\epsilon R^{d_1+d_2} \quad (3)$$

where$ci\epsilon R^{d_1+d_2}$, and $d1$+ $d2$ is the embedding dimension.

In order to study the influence of the length of entity span, we counted the lengths of all entity spans, as shown in Table 3. Overall, the percentage of entity span lengths on the interval (1-3) is more than half, reaching 58.7%, and the data on the intervals (4-6) (7-10) and (>10) show that the possibility of entity span becoming an entity is inversely proportional to the length. In terms of entity type breakdown, there is also a large difference in the proportion of different entity types in each interval, for example, Generic type has 96.9% on the (1-3) interval, which can be almost considered as Generic type only in the (1-3) interval, while Method and Task types are relatively evenly distributed. Thus entity span length is an important feature in entity classification. We train a specific width embedding matrix $W_k\epsilon R^{d_3}$ to obtain an embedding for a span of length k.

$$V'(s) = V(s) \circ W_k\epsilon R^{d_1+d_2+d_3} \quad (4)$$

where$W_k\epsilon R^{d_3}$, and $d1$+ $d2$+ $d3$ is the embedding dimension.

Finally,[CLS] , which represents the sentence context, is concatenated with $V'(s)$ to obtain the vector $V''(s)$, which is passed through a softmax classifier to predict the entity type.

$$V''(s) = V'(s) \circ b_{[CLS]}\epsilon R^{2d_1+d_2+d_3} \quad (5)$$

$$e(s) = W \cdot V''(s) + b\epsilon R^{d_4} \quad (6)$$

Where $b_{[CLS]}\epsilon R^{d_1}$, $2d1$+ $d2$+ $d3$ is the embedding dimension. And $d_1 + d_2$, "+1" is due to the 'null' entity ϕ that denotes the absence of an entity.

3.4 Relation classification

Those spans that are classified as ϕ by the entity classifier are filtered out. For the remaining spans, the task is to identify the relation between every pair of them. Consider a pair of spans (s_1,s_2) where $s1$ occurs before $s2$ in the input sentence. We assume relations to be asymmetric, so the relation directed from $s1$ to $s2$ may be different from that directed from $s2$ to $s1$, and each of them must be separately classified. We take the representations, $(c_i,\cdots,c_j)$, where ci is the embedding of the first token following $s1$ and cj is that of the last token preceding $s2$ in the sentence, and max-pooling:

$$c(s_1,s_2) = \text{max}pool(c_i,\cdots,c_j)\epsilon R^{d_1+d_2} \qquad (7)$$

Where $ci\epsilon R^{d_1+d_2}$, and $d1$+ $d2$ is the embedding dimension.The candidate relations from span $s1$ to $s2$ and s2 to s1 are separately encoded as in (8),(9):

$$R_{12} = V'(s_1) \circ c(s_1,s_2) \circ V'(s_2)\epsilon R^{3d_1+3d_2+2d_3} \qquad (8)$$

$$R_{21} = V'(s_2) \circ c(s_1,s_2) \circ V'(s_1)\epsilon R^{3d_1+3d_2+2d_3} \qquad (9)$$

The results are passed through a simple classifier with a confidence interval α (results beyond α are considered to have this relationship) and a sigmoid activation function to predict the type of relation. We have tried to include logical information and distance between candidate entities to predict entity type. We train a specific width embedding matrix $W_r\epsilon R^{d_5}$ to obtain an embedding for the distance between candidate entities.

$$R_{12}{}' = R_{12} \circ e(s_1) \circ e(s_2)\epsilon R^{3d_1+3d_2+2d_3+2d_4} \qquad (10)$$

$$R_{21}{}' = R_{21} \circ e(s_2) \circ e(s_1)\epsilon R^{3d_1+3d_2+2d_3+2d_4} \qquad (11)$$

$$R_{12}{}'' = R_{12}{}' \circ W_r\epsilon R^{3d_1+3d_2+3d_3+2d_4+d_5} \qquad (12)$$

$$R_{21}{}'' = R_{21}{}' \circ W_r\epsilon R^{3d_1+3d_2+3d_3+2d_4+d_5} \qquad (13)$$

$$y = \sigma(W\cdot R_{12}{}'+b) \qquad (14)$$

The loss function of the joint model is the sum of the cross-entropy losses of the entity classifier and the relational classifier. End-to-end training is performed by back-propagation, and the transformer is fine-tuned during the training process. To train the entity classifier, we used real entity spans as positive samples and added some non-entity spans as negative samples.

4. Experiments And Results

4.1 Datasets

Our goal is to extract method entities and relations from the scientific literature, so we evaluated our model on SciERC, a dataset from the scientific literature that contains both entities

Table 3. Length of tokenized entities. Ave[>10]*4 means the average percentage of every four lengths longer than 10.

Type	Length of tokenized entities				
	[1-3]	[4-6]	[7-10]	[>10]	Ave[>10]*4
Task	47.0%	31.9%	12.9%	8.3%	1.8%
Method	44.1%	36.0%	12.7%	7.3%	1.4%
Metric	72.3%	20.3%	4.3%	3.0%	1.7%
Material	52.6%	33.8%	9.4%	4.2%	1.9%
OtherScientificTerm	55.7%	32.8%	8.7%	2.8%	0.7%
Generic	96.9%	2.0%	1.1%	0.0%	0.0%
All	58.7%	27.8%	9.0%	4.5%	0.6%

Table 4. Performance on SciERC.

Model	NER			Boundaries RE			Strict RE		
	P(%)	R	F1	P	R	F1	P	R	F1
SCIIE[17]	67.2	61.5	64.2	47.6	33.5	39.2	-	-	-
PURE[31]	-	-	66.6	-	-	48.2	-	-	35.6
DYGIE[32]	68.6	67.8	68.2	46.2	38.5	42.0	-	-	-
DYGIE+[33]	-	-	67.5	-	-	48.4	-	-	-
PFN[34]	64.8	69.0	66.8	-	-	-	40.6	36.5	38.4
SPERT[35]	70.9	69.8	70.3	53.4	48.5	50.8	40.5	36.8	38.6
Ours	70.0	71.4	70.7	52.6	51.5	52.1	41.7	39.9	40.8

Table 5. Ablation study on SciERC. SpanL means the length of token span, Dist means the distance between candidate entities.

Model	NER			Boundaries RE			Strict RE		
（Sci）	P(%)	R	F1	P	R	F1	P	R	F1
Ours	70.0	71.4	70.7	52.6	51.5	52.1	41.7	39.9	40.8
-SpanL	69.5	71.0	70.3	51.3	51.5	51.4	40.0	40.3	40.1
-Type	69.4	70.5	70.0	51.3	50.0	50.7	39.4	38.6	39.0
-Dist	69.5	71.5	70.5	51.6	50.8	50.9	40.2	39.3	39.8

and relations.

The SciERC dataset is constructed from 500 abstracts of papers in the field of artificial intelligence, with a total of 2687 sentences. It contains six scientific entities as well as seven relations. The six scientific entities are **Task**, **Method**, **Metric**, **Material**, **Other-Scientific-Term**, and **Generic**, while the seven methods are **Compare**, **Conjunction**, **Evaluate-For**, **Used-For**, **Feature-of**, **Part-of**, and **Hyponym-of**. We follow the official cutoff methods: train (1861), dev (275), and test (551).

4.2 Evaluation Metrics

We use the standard Precision (P)(15), Recall (R)(16), and F1-score(17) to evaluate the model performance.

$$P = \frac{TP}{TP+FP} \tag{15}$$

$$R = \frac{TP}{TP+FN} \tag{16}$$

$$F1 = \frac{2*P*R}{P+R} \tag{17}$$

where TP, FP and FN stand for true positive, false positive, and false negative, respectively.

4.3 Results

Performance on SciERC. We report the performance on the SciERC dataset in Table 4. We compare the experimental results for six different models, and the F1 values for all three tasks improve compared to the baseline SPERT. Compared to the 0.36% F1 value improvement for the NER task, the performance improvement for the RE task is relatively significant, with Boundaries RE and Strict RE improving by 1.26% and 2.23%, respectively.

4.4 Ablation study

The ablation study in Table 5 shows the effect of removing entity span length, entity type, and relation distance on the final classification score. In the ablation experiments, we take the average of the three best results out of 20 experiments. Intending to compare the upper limits of the effect of feature. We observe that removing the entity span length decreases the F1 value of NER by 0.4% and has an impact on the subsequent RE tasks as well. Removing the entity type feature reduced the F1 scores of boundaries RE and strict RE by 1.3% and 1.8%, respectively, and removing the relation distance reduced the F1 scores of boundaries RE and strict RE by 1.16% and 1.0%, respectively, so we can conclude that entity type has a significant effect on relation extraction, especially on strict RE, while as the similar distance feature, the improvement of relationship distance for RE is significantly higher than that of entity span for NER.

5. Conclusion

We propose a semantically enhanced deep-learning model for extracting entities and relations from the scientific literature. We explored the impact of linguistic information on entity and relation extraction tasks, for which we added additional lexical information to the word vectors obtained through the pre-trained model to highlight lexical information, which proved to be superior to the pre-trained word vectors alone. Individually, in the entity recognition part, the length of the entity's token sequence is considered as a feature, while in the relation extraction part, entity type and relation distance are added, which also improves the accuracy of the task. In the future, extending the in-sentence feature information to inter-sentence contextual information is a promising challenge.

6. Funding Statement:

This research was supported by the Young Scientists Fund of the National Natural Science Foundation of China under Grant[71904078], Natural Science Foundation of Jiangsu Province of China under Grant[BK20190793], the Postgraduate Research and Practice Innovation Program of Jiangsu Province of China under Grant[GJFXW21001].

7. References

[1] Jinha A E. Article 50 million: an estimate of the number of scholarly articles in existence[J]. Learned publishing, 2010, 23(3): 258-263.

[2] Hassan S U, Safder I, Akram A, et al. A novel machine-learning approach to measuring scientific knowledge flows using citation context analysis[J]. Scientometrics, 2018, 116(2): 973-996.

[3] Xie I, Babu R, Lee T H, et al. Enhancing usability of digital libraries: Designing help features to support blind and visually impaired users[J]. Information Processing & Management, 2020, 57(3): 102110.

[4] Ding Y, Stirling K. Data-driven discovery: A new era of exploiting the literature and data[J]. Journal of Data and Information Science, 2016, 1(4): 1-9.

[5] Bhatia S , Mitra P , Giles C L . Finding algorithms in scientific articles[C]// Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010. ACM, 2010.

[6] Katsurai M, Joo S. Adoption of Data Mining Methods in the Discipline of Library and Information Science[J]. Journal of Library & Information Studies, 2021, 19(1).

[7] Lam C, Lai F C, Wang C H, et al. Text mining of journal articles for sleep disorder terminologies[J]. PloS one, 2016, 11(5): e0156031.

[8] Li K, Yan E. Co-mention network of R packages: Scientific impact and clustering structure[J]. Journal of Informetrics, 2018, 12(1): 87-100.

[9] Wang Y, Zhang C. Finding more methodological entities from academic articles via iterative strategy: A preliminary study[J]. training, 2019, 2787: 2.73.

[10] Zhu G, Yu Z, Li J. Discovering Relationships between Data Structures and Algorithms[J]. J. Softw., 2013, 8(7): 1726-1735.

[11] Duck G, Nenadic G, Brass A, et al. bioNerDS: exploring bioinformatics' database and software use through literature mining[J]. BMC bioinformatics, 2013, 14(1): 1-13.

[12] Petasis G, Cucchiarelli A, Velardi P, et al. Automatic adaptation of Proper Noun Dictionaries through cooperation of machine learning and probabilistic methods[C]//Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval. 2000: 128-135.

[13] Alfonseca E, Manandhar S. An unsupervised method for general named entity recognition and automated concept discovery[C]//Proceedings of the 1st international conference on general WordNet, Mysore, India. 2002: 34-43.

[14] Borrega O, Taulé M, Martı M A. What do we mean when we speak about Named Entities[C]//Proceedings of Corpus Linguistics. 2007.

[15] Zeng Daojian, Liu Kang, Lai Siwei, et al. Relation classification via convolutional deep neural network [C]∥Proc of the 25th Int Conf on Computational Linguistics. Stroudsburg: ACL,2014

[16] Socher R , Huval B , Manning C D , et al. Semantic Compositionality through Recursive Matrix-Vector Spaces[C]// Joint Conference on Empirical Methods in Natural Language Processing & Computational Natural Language Learning. 2012.

[17] Luan Y, He L, Ostendorf M, et al. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction[J]. arXiv preprint arXiv:1808.09602, 2018.

[18] Xu K , Feng Y , Huang S , et al. Semantic Relation Classification via Convolutional Neural Networks with Simple Negative Sampling[J]. Computer Science, 2015, 71(7):941-9.

[19] Schlichtkrull M , Kipf T N , Bloem P , et al. Modeling Relational Data with Graph Convolutional Networks[J]. 2017.

[20] Zhang Y , Guo Z , Lu W . Attention Guided Graph Convolutional Networks for Relation Extraction[J]. 2019.

[21] Yu X , Lam W . Jointly identifying entities and extracting relations in encyclopedia text via a graphical model approach. 2010.

[22] Qi Li , Ji H . Incremental Joint Extraction of Entity Mentions and Relations[C]// Meeting of the Association for Computational Linguistics. 2014.

[23] Miwa M, Sasaki Y. Modeling joint entity and relation extraction with table representation[C]//Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014: 1858-1869.

[24] Gupta A, Eral H B, Hatton T A, et al. Nanoemulsions: formation, properties and applications[J]. Soft matter, 2016, 12(11): 2826-2841.

[25] Katiyar A, Cardie C. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees[C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017: 917-928.

[26] Shen X, Tang H, McDanal C, et al. SARS-CoV-2 variant B. 1.1. 7 is susceptible to neutralizing antibodies elicited by ancestral spike vaccines[J]. Cell host & microbe, 2021, 29(4): 529-539. e3.

[27] Wu Q, Zeng Y, Zhang R. Joint trajectory and communication design for multi-UAV enabled wireless networks[J]. IEEE Transactions on Wireless Communications, 2018, 17(3): 2109-2121.

[28] Wei Z , Su J , WaNg Y , et al. A Novel Cascade Binary Tagging Framework for Relational Triple Extraction[C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020.

[29] Beltagy I, Lo K, Cohan A. SciBERT: A pretrained language model for scientific text[J]. arXiv preprint arXiv:1903.10676, 2019.

[30] Lee J, Yoon W, Kim S, et al. BioBERT: a pre-trained biomedical language representation model for biomedical text mining[J]. Bioinformatics, 2020, 36(4): 1234-1240.

[31] Zhong Z, Chen D. A frustratingly easy approach for entity and relation extraction[J]. arXiv preprint arXiv:2010.12812, 2020.

[32] Luan Y, Wadden D, He L, et al. A general framework for information extraction using dynamic span graphs[J]. arXiv preprint arXiv:1904.03296, 2019.

[33] Wadden D, Wennberg U, Luan Y, et al. Entity, relation, and event extraction with contextualized span representations[J]. arXiv preprint arXiv:1909.03546, 2019.

[34] Yan Z, Zhang C, Fu J, et al. A partition filter network for joint entity and relation extraction[J]. arXiv preprint arXiv:2108.12202, 2021.

[35] Eberts M, Ulges A. Span-based joint entity and relation extraction with transformer pre-training[J]. arXiv preprint arXiv:1909.07755, 2019.