Relation Extraction Task for Inorganic Material Synthesis Procedure

Shanshan Liu
RIKEN, AIP
shanshan.liu@riken.jp

Yuji Matsumoto
RIKEN, AIP
yuji.matsumoto@riken.jp

1 Introduction

To transform the knowledge existing in natural language text into a structured form, named entity recognition (NER) and relation extraction (RE) technologies are essential. Early methods are mostly pipelined, that is, first identify entities in the text then classify the relationships between entities. End-to-end RE methods that learn entity labels and relation labels together to take the interaction between entities into consideration during training become popular recently (like DyGIE framework [1]). Although the joint learning methods perform well on many standard benchmarks (ACE05, DocRED [2], CDR, SciREC and so on), the research of [3] shows that the pipeline methods can achieve state-of-the-art (SOTA) results over the existing joint learning methods. In addition, although there are many researches on RE, they still have not been verified on more practical issues. This is mainly due to the simple problem background of the current dataset, mostly are for sentence-level relation, with only one pair of entities in a sentence; or for document-level relation, but only one relation instance is annotated in one document (CDR [4]). In addition to the complexity of the relationships, there are also limitations in the knowledge domain – most of datasets are from news (ACE05), computer science (DocRED, SciERC) or biomedical fields (CDR, GDA), datasets from other fields are rarely reported.

In this paper, we give a more challenging task, the RE task in the inorganic material synthesis procedure (abbreviated as "procedure") extraction. The pipeline extraction of procedures consists of three sub-tasks namely, extracting text blocks containing procedure information from the original paper, performing named entity recognition on the text blocks, and classifying the identified entity pairs. [5] has made a good effort on procedure extraction. They made a definition of procedure and researched the models for the block extraction and NER task.

We evaluated the performance of the existing document-level relation extraction methods in two scenarios. In the first one, the text blocks and entities are all gold annotations. This is also the standard task of most RE researches. The second scenario is to use the result of the text block extractor, and the entities predicted by the named entity recognizer. It is the situation that the relation extractor faces in the pipelined knowledge extraction. We only applied the best method in standard RE task to the pipelined procedure extraction.

The models used in this work include the Bi-LSTM model, which is widely used in various knowledge domains, and the ATLOP [6], which is a SOTA model for extracting document-level relations. In view of that the knowledge of chemical domain is to be extracted, we choose the pre-trained language model SciBERT [7] to generate word representations. The dataset used for evaluation in our work is based on the dataset mentioned in [5], however, the number of papers is increased and each paper has procedure information.

Our contributions can be concluded in threefold:

- We evaluate two models' performance on a specific task that is close to the real need – to capture the relations in inorganic material synthesis procedure.
- We notice that the results of procedure extraction are similar to other tasks that also use named entity recognition and relationship extraction for knowledge extraction. The SOTA result on ACE05 dataset, ACE04 dataset and our procedure dataset are 67.8, 62.2 and 67.92 respectively.
- We introduce two effective techniques for the RE task of procedure extraction. The rule_top component helps to select the best types from several positive types predicted by multi-label binary classification models. The entity typemarker achieved the highest F1 score of 93.7 when the inputs of RE are gold.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

Table 1 Statistics of entities in procedure dataset

Entity type	#Train	#Dev	#Test	#All
Material	1092	153	145	1390
Condition	1631	211	201	2043
Method	231	32	31	294
Process	1249	187	171	1607
All	4210	583	548	5341

Dataset statistics about the number of named entities for training (#Train), development (#Dev), testing (#Test) and of all papers (#All).

Table 2 Statistics of relations in procedure dataset

Relation type	#Train	#Dev	#Test	#All
Input_of	761	103	94	958
Output_of	234	26	29	289
Condition_of	1757	235	222	2214
Method_of	203	31	28	262
Next_of	1032	163	147	1342
All	3987	558	520	5065

2 Methodology

2.1 Task definition

The definition of the procedure follows [5]. A procedure is represented by four types of entities and five types of relations. Four types of entities are "Material", "Condition", "Method", and "Process". Five types of relations are from "Process" to all four types of entities. If the tail entity is "Material", two relation types are possible: "Input_of", "Output_of"; if the tail entity is "Condition" or "Method", the relation type is "Condition_of" or "Method_of" respectively. Two "Process" entities are linked by "Next_of" relationship.

We evaluate the RE methods in two scenarios. One is the standard RE task, meaning that the inputs of RE methods are gold text blocks and only human-annotated entities. The other one is the RE task in a pipelined procedure extraction. In this situation, the inputs of RE methods contains 3 combinations: the gold named entities (NEs) in the predicted blocks, the predicted NEs in the gold blocks, and the predicted NEs in the predicted blocks. The outputs of the RE task are the outputs of the procedure extraction if the inputs of the RE task are predicted NEs in the predicted blocks.

2.2 Model

In this section, we give the details of the methods applied in experiments.

Standard Relation Extraction One of our baseline is the **rule-based relation extractor** in [5] with small changes. We select the longest material entity in each document as output, while other materials are inputs of procedures. We add a candidate condition list, to deal with the situation that a material entity may play a role as a condition of the process. For a material entity in this list, the relationship contains it is labeled as "Condition_of".

Two neural network (NN) methods are applied in standard RE task. Benefit from the capability to process the long input max to 1024 tokens, the combination of the **Glove embedding, Bi-LSTM encoder and Bilinear layer decoder** (abbreviated as **"Bi-LSTM"**) is used as a baseline in both sentence-level and document-level RE tasks [2]. We apply this method to see the performance of simple NN-based method on complex issues. To research whether the novel techniques perform well in standard benchmarks are effective in other domains, we apply the model **ALTOP**. ALTOP utilizes the BERT-based pre-trained language model which is better than GloVe in recent works. Two techniques in ALTOP make it achieved SOTA on DocRED datasets: adaptive thresholding loss to reduce decision errors during inference, and localized context pooling to better use of the local context of the entities.

Pipelined procedure extraction As with [5], we use a pipelined approach to extract the procedures which includes three sub-tasks: extracting text blocks containing procedure information from the original chemical paper, performing named entity recognition on the text blocks, and classifying the identified entity-pairs. Our research only focuses on the RE task, so we use the best performing strategy in block extraction and named entity recognition published by [5]. The output of the block extractor is generated based on the classification of a sentence classifier that predicts whether a sentence is in a block that describes a procedure. The classifier is in the architecture of SciBERT, multi-layer perceptrons, and a SoftMax layer. The NER model utilizes SciBERT to generate token vectors, 2-layer Bi-LSTM to encode sentence-level information, and a SoftMax layer to predict the token label.

Technique for the procedure extraction To reduce the false positive instances predicted by multi-label binary classification, we provide a **rule_top** component. In our definition of relations, the head entity of each relation must be "Process", and a tail entity only links to one head entity most of the time. When several head entities are predicted

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

Table 3 Result of block extraction

	Pre.	Rec
Sentence-level	31.80	98.98
Block-level	70.27	96.30

positive for a tail entity by the RE model, the rule_top component only outputs the head with the highest score predicted by the model and treats others as negative instances.

Inspiring by the work by [3], we try to import **entity typemarker** in our method. Two relation models with typemarker are published in [3]. One considers every pair of entities independently by inserting typed entity markers, which means a sentence will be encoded several times if it contains multiple entity pairs. This is time-consuming in our task because we need to process long texts while multiple relations may appear in each input text. They also provide a relation model with batch computations, that typemarker and corresponding entity token (the start marker to the first token of an entity, the end marker to the last token) share the positional embedding. The ALTOP model is designed to utilize the positional information encoded by BERT instead of additional positional feature. There is still a way to insert typemarker, which is simple and practical. After the input text is tokenized by the BERT-based tokenizer, "*" is inserted into the sentence before and after each entity in ALTOP. We change the "*" mark into typemarker. Given an entity, we concatenate "E:" or "/E:" to the first three letters of the entity type as its typemarkers. For example, the typemarker of "Material" will be "<E:Mat>" and "</E:Mat>". The typemarkers are different for the head and tail entities in [3], while we do not design different markers for head and tail entities.

3 Experiment

3.1 Dataset

The **procedure dataset**[1)] we used contains 241 thermoelectric material papers annotated by chemistry experts, including entity and relation labels. Each paper contains at least one procedure. 193 papers are used for training, 24 for development, and the remaining 24 for evaluation, all selected randomly. The dataset statistics are shown in Table 1 and Table 2.

1) The annotation results will be made public.

Table 4 Result of named entity recognition

Model	Pre.	Rec.	F1
pred_g	78.30	81.70	79.96
pred_p	70.35	80.18	74.94

"Pred_g" is the NER model trained on the gold blocks, while "pred_p" is trained on the predicted blocks.

Table 5 Result of standard RE task

Model	Pre.	Rec.	F1
Rule-based	90.26	86.24	88.21
Bi-LSTM	86.13	68.08	76.05
+rule_top	93.36	67.69	78.48
ATLOP	89.94	89.42	89.68
+rule_top	95.00	88.37	**91.37**

3.2 Experiment setting

For the Bi-LSTM RE model, we implement a model same as the one implemented in [2]. For the ALTOP RE model, we select scibert_scivocab_cased as the pre-trained model. The batch sizes for training and testing are 4 and 8 respectively. We set a learning rate of 1e-5 for weights in the pre-trained model, 5e-5 for others, with a linear warm-up for the first 6% steps followed by a linear decay to 0. For the block extractor, we follow [5] to train a sentence classifier and construct predicted blocks. The performance of the block extractor is shown in Table 3. 98.98% of the sentences with procedure information are successfully extracted. Two named entity recognizers are trained with different inputs but with the same architecture and setting followed as [5]. The "pred_g" is the NER model trained on the annotated entities in gold blocks, while "pred_p" is trained on the predicted blocks with annotated entity labels. Table 4 presents the results of our NER models. The max sequence length of both NER and RE tasks is 1024.

4 Result and Discussion

RE models are compared in the standard RE task that the gold blocks and gold entity labels are given. The best RE model is applied in the procedure extraction, while the RE model has to face more noise because the blocks are predicted, or the named entities are results of NER models.

Standard RE task The experimental results of standard RE task are shown in Table 5. Following previous works, we use Precision, Recall and F1 scores in evaluation.

The first we noticed is the good performance of the rule-based approach. Although the rules we used are intuitive, the rule-based have an accuracy of 90.26. In most of other

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

Table 6 ALTOP + rule_top ignoring the missed relations caused by entity missing

Block	NE	Covered Rel (%)	Pre.	Rec.	F1
gold	gold	100.00	95.00	88.37	91.37
gold	pred_g	82.36	70.79	86.12	77.71
pred	gold	97.67	94.18	86.71	90.29
pred	pred_g	80.62	65.66	83.65	73.57
pred	pred_p	77.33	69.22	86.22	76.79

ALTOP + rule_top is trained on the gold NEs in gold blocks. "Covered Rel" is the relationships that both its head entity and tail entity are identified by the NER models.

Table 7 ALTOP + rule_top in procedure extraction

Block	NE	Covered Rel (%)	Pre.	Rec.	F1
gold	pred_g	82.36	70.79	70.93	70.86
pred	gold	97.67	94.18	84.69	89.18
pred	pred_g	80.62	67.44	67.44	66.54
pred	pred_p	77.33	69.22	66.67	67.92

tasks, NN-based methods can capture positive examples outside the rules and have a higher recall. In our task, there is a difference between the simple Bi-LSTM model and the rule-based method, except that Bi-LSTM performs worse than the rules, especially in the recall. This situation may be caused by the simplicity of the procedure dataset - the text order of a series of processes is the same as the actual processing order, and the description of procedures is in a certain form that matches with our rules.

Even though the simple NN-based model fails to outperform the rule-based method, the pre-trained language model and novel techniques help ATLOP win the game in this round with an F1 score of 89.68. And the rule_top component does reduce the count of false-positive instances, bringing us a high precision of 95.00. Comparing to the performance of the ATLOP, the Bi-LSTM model is not good enough in distinguishing between two head entities given a tail entity. The human-check of error predictions proves that the Bi-LSTM model may link a tail entity to an incorrect head entity close to the correct answer, even if two head entities are not semantically similar.

Pipelined procedure extraction For the RE task in procedure extraction, performances of the models are shown in Tables 6, 7, 8. Table 6 shows the results not taken the missed relations caused by missed entities into account. As is shown: 1) The impact brought by the NER models is greater than the block extractor. Performance of gold NEs in the predicted blocks gets 4 points lower than gold NEs in the gold blocks; on the predicted block, passing the pred_g model's output to the RE task decreases 18.5% of

Table 8 ALTOP + rule_top in procedure extraction with/without typemarker

Block	NE	Typemarker	Pre.	Rec.	F1
gold	gold	yes	96.33	90.96	93.57
		no	94.69	89.23	91.88
pred	pred_g	yes	64.63	67.64	66.01
		no	65.66	67.44	66.54
pred	pred_p	yes	66.01	64.73	66.34
		no	69.22	66.67	67.92

ALTOP + rule_top model with/without typemarker is trained on the gold NEs in gold blocks.

performance than gold NEs. 2) An NER model trained on the predicted blocks can reduce the false-positive entities appeared in the false sentences in the predicted block. The pred_p model achieves 3.22 points higher than pred_g. 3) Even though the NER model achieved over 80.0 recall, the missed relations caused by entity missing have big effects. Only 77.33% of real relations can be observed after block extraction and NER.

Taking the missed relations by the errors of NER into consideration, the recall of each combination of inputs reduces (See Table 7). The results on the pred_p NEs in predicted blocks are the final performance of procedure extraction. The pipelined methods taken ALTOP + rule_of model as RE model can capture 66.67% of relations in procedures, while 77.33% of the predictions are correct.

Results by the ALTOP model with or without typemarker are shown in Table 8. In the standard RE task, typemarker contributes 1.69 points in the F1 score, while it brings no improvement in the other situations. We think typemarkers help the model better fit into data, losing a degree of generalization ability to unobserved instances during training.

5 Conclusion

We experiment with two models that reported good performance in other works on the relation extraction of inorganic material synthesis procedures. We provide simple techniques, the rule_of component and entity typemarker to help existed methods get better performance on a specific task. We find the missed named entities during named entity recognition cause a great effect on the pipelined knowledge extraction. For that, we summarized two potential directions as our future works: improving the accuracy of named entity recognition, and improving the ability of the relation extraction model to distinguish between false-positive entities and real entities.

This work is licensed by the author(s) under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/).

References

[1] David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. Entity, relation, and event extraction with contextualized span representations. *arXiv preprint arXiv:1909.03546*, 2019.

[2] Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. Docred: A large-scale document-level relation extraction dataset. *arXiv preprint arXiv:1906.06127*, 2019.

[3] Zexuan Zhong and Danqi Chen. A frustratingly easy approach for joint entity and relation extraction. *arXiv preprint arXiv:2010.12812*, 2020.

[4] Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, Vol. 2016, , 2016.

[5] Shanshan Liu, Mutsunori Uenuma, Hiroyuki Shindo, and Yuji Matsumoto. Extraction of the material synthesis procedure. *Fourth International Workshop on SCIentific DOCument Analysis (SCIDOCA2020)*, 2020.

[6] Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. Document-level relation extraction with adaptive thresholding and localized context pooling. *arXiv preprint arXiv:2010.11304*, 2020.

[7] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*, 2019.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).