A Semi-supervised Transfer Learning Framework for Low Resource Entity and Relation Extraction in Scientific Domain

Hao Wang
hwang20@bit.edu.cn
Beijing Institute of Technology
Beijing, China

Xian-Ling Mao
maoxl@bit.edu.cn
Beijing Institute of Technology
Beijing, China

Heyan Huang
hhy63@bit.edu.cn
Beijing Institute of Technology
Beijing, China

ABSTRACT

With the development of scientific communities, the amount of papers increases quickly. It's important to convert the unstructured scientific papers into structured knowledge base, which relies on Information Extraction (IE) to extract entities and their relationships. Most existing IE methods require abundant annotated data, which is time-consuming and expensive to obtain, especially in scientific domain because it requires annotators with domain knowledge. Recently, several works have been proposed to solve the problem by semi-supervised learning. However, these methods require the input sentence to contain only two entities and simply classify the relationship between these two entities. Obviously, it is far from a realistic application scenarios that both entities and relations need to be extracted from raw text. In this paper, we propose a Semi-supervised Transfer Learning (STL) framework to tackle joint entity and relation extraction problem in a low resource situation. Specifically, STL adopts two main strategies: a rebalancing strategy for alleviating the bias to the majority class during semi-supervised learning, and a transfer learning strategy for transferring knowledge from domains with relatively rich annotation to domains that lack annotated data. Experiment results on two public scientific IE datasets show the effectiveness of the proposed method.

CCS CONCEPTS

• **Computing methodologies** → **Information extraction**.

KEYWORDS

Information Extraction, Semi-supervised Learning, Transfer Learning

1 INTRODUCTION

With the development of scientific communities, the amount of papers increases quickly. It's important to convert the unstructured scientific papers into structured knowledge base, which relies on Information Extraction (IE) to extract entities and their relationships. Two key tasks in IE are Named Entity Recognition (NER) and Relation Extraction (RE). NER aims to identify and classify entities from raw text, while RE aims to decide the relations between entities and generate triplets (h, r, t) where h, r, t are head entity, relationship and tail entity respectively. They are crucial to constructing high-quality knowledge base which can be used for many other downstream tasks in Natural Language Process (NLP), such as question answering, text summarization, dialog system and so on. To build a knowledge base, entities and relations need to be extracted jointly. Most existing joint entity and relation extraction methods require abundant annotated data which is expensive to obtain, especially in scientific domain because of the high requirements for annotators.

Previous works attempt to solve the problem by semi-supervised learning (SSL). Most existing methods focus on generating high-quality pseudo labels by introducing extra information such as template information from labeled data [19] or gradient information [8]. DualRE [12] designs an auxiliary task for unlabeled data. MetaSRE [7] adopts meta-learning to reduce noise in pseudo labels. But these methods simplify the joint entity and relation extraction task to a sentence classification task, i.e. the input sentences contain only two entities and the output is a single relation type. A model trained under such idealized settings won't meet the requirements for realistic application which needs to extract both entities and relations.

To solve the problem above, we proposed a novel Semi-supervised Transfer Learning (STL) framework for joint entity and relation extraction under a low resource condition in scientific domains. The proposed STL framework utilizes a rebalancing strategy and a transfer learning strategy to improve performance. The rebalancing strategy is used to alleviate the influence of data imbalance which is a serious problem in scientific domains. For example, in SciERC [13], a widely used information extraction dataset in artificial intelligence domain, relation "Used-for" accounts for more than half, while relation "compare" only takes up about 5%. The rebalancing strategy makes the data distribution more balanced by selecting unlabeled data predicted as minority class at a higher probability when expanding training set. The transfer learning strategy is designed for transferring common knowledge between domains. For example, most natural science and engineering domains contain entities with types "Method" and "Problem", and relation "Used for" between them. A model is first trained on domains with relatively rich annotated data, e.g. computer science, then its encoder is used to initialize the model to be trained on a new domain. We don't transfer the whole model for two reasons. On the one hand, the type and number of labels between source domain and target domain are different. On the other hand, the classification network may differ depending on the domain and training corpus.

The contributions of the proposed work are as follows:

- We proposed a semi-supervised transfer learning framework for low resource joint entity and relation extraction in scientific domains, which utilize unlabeled data and cross-domain knowledge at the same time.
- We adopt a class rebalancing strategy when expanding training set with pseudo labels to prevent bias to majority classes.

- To the best of our knowledge, we are the first ones to adopt semi-supervised learning and transfer learning simultaneously for low resource scientific information extraction. Experiment results show the effectiveness of our method.

The remainder of this paper proceeds as follows. Section 2 introduces related work in detail. Section 3 describes the proposed STL framework. Section 4 presents experiment results and analysis. In Section 5, we summarize the main conclusions.

2 RELATED WORK

In this section, we will introduce semi-supervised learning and transfer learning for relation extraction respectively.

Semi-supervised learning for relation extraction. Several semi-supervised learning methods for relation extraction have been proposed. DualRE [12] proposes a complementary dual task of relation extraction, i.e. retrieving sentences expressing a certain relation. NERO [19] combines template method with semi-supervised learning. It generates pseudo labels by calculating the similarity between unlabeled data and relation templates. To reduce the influence of noisy pseudo labels, MetaSRE [7] proposed a label generation network trained only on labeled data. It also adopts a label selection and exploitation scheme to guarantee the quality of selected pseudo labels. GradLRE [8] supposes that labeled data and unlabeled should share gradient direction for updating. It uses reinforcement learning to guide the gradient of unlabeled data to approximate the gradient of labeled data. However, the methods above assume the input sentences only contain two entities and the two entities have been given, which is much simpler than the realistic condition.

Transfer learning for relation extraction. Transfer learning in NLP can be classified into 4 categories: domain adaptation, cross-lingual learning, multi-task learning, and sequential transfer learning [14]. Several sequential transfer learning methods for relation extraction have been proposed. [6] fine-tunes domain-specific pre-trained language model [2, 10] in domain-related dataset. ReTrans [3] generates relation vectors from existing knowledge bases. These methods transfer knowledge from a source domain to a specific domain, which is expensive due to the high cost of building domain-specific language model or knowledge base. [11] trains model on two datasets from the same domain under a multi-task learning framework. However, no previous work explores transfer learning between two different domains with limited data.

3 THE PROPOSED STL FRAMEWORK

We propose a semi-supervised transfer learning framework for low resource information extraction in scientific domains. In this section, we will give the notions first. Then we will introduce the transfer learning strategy, semi-supervised learning strategy and base model in turn.

3.1 Notions

Let $\mathcal{E}$ and $\mathcal{R}$ denote the entities and relations label set respectively. An input instance includes three parts: a sentence $X = \{x_1, x_2, ..., x_N\}$, entity set $E = \{(e_i, y_i^e) : y_i^e \in \mathcal{E}, i \in \{1, 2, ..., N_E\}\}$ where N_E is the number of entities, and relation set $R = \{(h_i, t_i, y_i^r) : h_i, t_i \in E, y_i^r \in \mathcal{R}\}$. Each entity e_i is a span denoted by start position $start(i)$ and end position $end(i)$. The complete input includes the labeled data $L = \{(X_i, E_i, R_i) : i \in \{1, 2, ..., N_L\}\}$ and unlabeled data $U = \{X_i : i \in \{1, 2, ..., N_U\}\}$ where N_L and N_U are the numbers of labeled data and unlabeled data respectively.

3.2 STL Framework

The STL framework is shown in Figure 1. As noted in introduction, it combines transfer learning and semi-supervised learning. We will describe each of them in detail.

3.2.1 Transfer learning. The intuition behind transfer learning is that different scientific domains share some common knowledge. We hope the knowledge learned from source domain can promote the model's performance in target domain. In this work, the encoder is a pre-trained language model in scientific domain SciBERT [2] that consists of 12 Transformer [17] layers, of which each layer encodes different linguistic information. The top layers capture semantic information [9] that is important for entity and relation extraction, so we transfer the top layers of SciBERT between source domain and target domain. Specifically, an entity and relation extraction model M_S is trained on source domain first, then the top SciBERT encoder layers of M_S are used to initialize M_T which is to be trained on target domain.

3.2.2 Self-training. Self-training is a widely used semi-supervised learning method, which trains a model in an iterative manner. Each iteration involves two steps. First, the model is trained on labeled dataset L. Second, the pre-trained model generates pseudo labels for unlabeled dataset U. For each unlabeled data, we can get a predicted entity set $\hat{E} = \{(\hat{e}_i, \hat{y}_i^e) : \hat{y}_i^e \in \mathcal{E}, i \in \{1, 2, ..., \hat{N}_E\}\}$ and relation set $\hat{R} = \{(\hat{h}_i, \hat{t}_i, \hat{y}_i^r) : \hat{h}_i, \hat{t}_i \in \hat{E}, \hat{y}_i^r \in \mathcal{R}\}$, then the pseudo-labeled dataset $\hat{U} = \{(X_i, \hat{E}_i, \hat{R}_i), i \in \{1, 2, ..., \hat{N}_U\}\}$ is sorted by instances score which is the average of all entities and triplets scores. To reduce the noise in pseudo labels, we filter out entities and triplets whose score is under the given thresholds τ_{semi}^e and τ_{semi}^r respectively. Finally, instances with a high score will be added to L for next iteration.

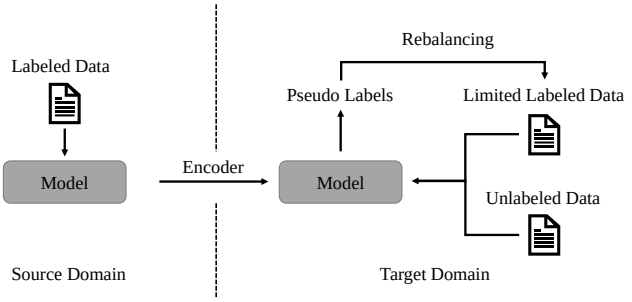

Figure 1: STL Framework.

3.3 Rebalancing Strategy

To solve the class imbalance problem, we follow the rebalancing strategy proposed by [18]. Suppose the instance numbers of each class are sorted in descending order, i.e. $N_1 \geq N_2 \geq N_3 \geq ... \geq N_C$ where N_c denotes the number of instances belonging class c and C

is the label space size, then unlabeled instances predicted as class c are included into training set at the rate of

$$P_c = (\frac{N_{C+1-c}}{N_1})^{\alpha} \tag{1}$$

Where α is a hyperparameter to control the number of instances added to training set. This distribution guarantees that the smaller portion of a class in the total dataset, the more likely it is to be added into the training set for unlabeled data predicted as this class.

We assign a representative entity label and relation label to each instance. First, the class distribution in the dataset is counted. For NER, we select the pseudo entity label with the lowest portion as the representative label. So does RE. Then we sample the unlabeled instances based on their representative entity label or relation label.

3.4 Base Model

The proposed STL framework can be adapted to any entity and relation extraction model. In this work, we select SpERT [4], a powerful span-based model for entity and relation extraction, as the base model. SpERT identifies all entities first, then assign relation to all entity pairs. Let $S = \{s_1, s_2, ..., s_{N_S}\}$ be the set of spans up to length l. For a given span $s = \{x_i, x_{i+1}, ..., x_{i+k}\}$ whose length is $k + 1$, the representation of s is made up of three parts:

- **Span Embedding** : the semantic information of a span depends on the tokens, so the representation should incorporate the tokens representation $f(x_i, x_{i+1}, ..., x_{i+k})$ where f is max-pooling function in SpERT.
- **Size Embedding**: size is an auxiliary clue to decide whether a span is an entity. Each size embedding w_{k+1} is a learnable vector.
- **Sentence Embedding**: the context information is also important for named entity recognition, so the [CLS] token embedding h_{CLS} of SciBERT encoder is added.

The final span representation is:

$$h^s = f(x_i, x_{i+1}, ..., x_{i+k}); w_{k+1}; h_{CLS} \tag{2}$$

where ; means concatenation operation. Then the representation is fed into a softmax classifier:

$$y^s = softmax(W^e \cdot h^s + b^e) \tag{3}$$

where $y^s \in \mathcal{E} \cup \{\epsilon\}$. ϵ means a span is not an entity. Then each entity pairs will be fed into a relation classifier. Given 2 entity spans s_1 and s_2, the input representation is made up by two parts:

- **Span Embedding** : head entity embedding h^{s_1} and tail entity embedding h^{s_2}.
- **Context Embedding**: context embedding for relation extraction is the representation of tokens between two entities. The context embedding c_{s_1,s_2} is also generated by max-pooling.

Since some relations are asymmetric, so two representations of an entity pairs are fed into the classifier:

$$h^r_{forward} = h^{s_1}; c_{s_1,s_2}; h^{s_1} \tag{4}$$

$$h^r_{backward} = h^{s_2}; c_{s_1,s_2}; h^{s_1} \tag{5}$$

Different from NER, SpERT regards relation extraction as a multi-label classification. Classifier for a specific relation $r_i \in \mathcal{R}$ is:

$$y^{r_i}_{forward/backward} = \sigma(W^{r_i} \cdot h^r_{forward/backward} + b^{r_i}) \tag{6}$$

where σ is sigmod function. Only relations whose score is greater than a confidence threshold τ_r will be assigned to entity pairs.

4 EXPERIMENT

In this section, baselines, datasets, experiment and evaluation settings will be introduced first. Then results on STL and baselines will be compared. Finally, an analysis of results will be presented.

4.1 Baselines

Since no existing methods are designed for our settings, we only select the representative semi-supervised learning method self-training [15] as baseline.

4.2 Datasets

We evaluate the proposed STL framework and baselines on two widely used public datasets:

- **SciERC** [13]: The SciERC dataset is constructed on 500 abstracts in artificial intelligence domain, which contains three subtasks: named entity recognition, relation extraction and coreference resolution. It annotates 6 entity types (*Task, Method, Metric, Material, OtherScientificTerm, Generic*) and 7 relations types (*Used-for, Feature-of, Hyponym-of, Part-of, Evaluate-for, Compare, Conjunction*).
- **ADE** [5]: The ADE dataset is an information extraction dataset in biomedicine domain, which consists of 4,722 sentences. It only annotated a single relation *AdverseEffect* and two entity types (*AdverseEffect, Drug*).

4.3 Experiment Settings

4.3.1 Data Split. For SciERC, we follow the data split in [13]. For ADE, it doesn't provide an official split, so we divide the corpus into training set, validation set and test set in a ratio of 80/10/10%. For both SciERC and ADE datasets, we randomly sample half of the training set as unlabeled data by removing all labels and 20% of the rest as labeled training set. To eliminate the effect of randomness, we repeat this process 5 times and report the average results.

4.3.2 Implementation Details. τ^e_{semi} and τ^r_{semi} are set to 0.2 and 0.6 respectively. Only the last layer of SciBERT encoder is transferred to target domain. An adam optimizer with $2e-5$ peak learning rate is adopted. For fair comparison, only iterations for labeled data are considered when calculating warmup steps of learning rate. Other hyperparameters of SpERT follow the settings reported in [4].

For transfer learning module, ADE and SciERC is the source domain for each other, i.e. when ADE is the target domain, SciERC is regarded as the source domain, and vice versa. All labeled data is available when training model on source domain.

Since the relation imbalance is more serious in SciERC, we sample pseudo labels according to relation type for SciERC. For ADE, we sample by entity type because there is only one relation type. Entities with type *generic* are removed from SciERC since extracting pronouns is not our target.

4.4 Evaluation Settings

Following the evaluation settings in [16], a predicted entity will be regarded as correct if both the boundaries and type are correct. For relation extraction, there are two criteria: boundaries settings and strict settings [1]:

- **Boundaries Settings** (denoted as **RE**) : A predicted triplet will be regarded as correct if the relation type and the boundaries of two entities are correct. Whether the type of two entities is correct doesn't matter.
- **Strict Settings** (denoted as **RE+**) : Based on boundaries setting, the entity type of two entities also must be correct.

4.5 Main Results

Results of baselines and proposed STL framework are shown in Table 1. It can be observed that:

- STL outperforms the self-training baseline and supervised learning on most metrics. Compared to supervised learning, it gains 1.1/1.2/1.2 F1 improvements in NER/RE/RE+ on ADE and 0.3/0.7 F1 improvements in NER/RE on SciERC.
- Self-training performs worse than supervised learning on almost all metrics. We argue it's caused by the noise in pseudo labels. Unlike conventional classification tasks, one instance in joint entity and relation extraction contains multiple entities and triplets. Although we only add pseudo labels with a high average confidence score to training set, they still include wrong labels with a high probability.

Table 1: F1 in ADE and SciERC. "supervised" means the model is trained only on the labeled training set without any semi-supervised or transfer learning techniques.

Model	ADE			SciERC		
	NER	RE	RE+	NER	RE	RE+
Supervised	85.5	72.4	72.4	59.4	31.5	**21.5**
Self-training	84.9	71.1	71.1	58.4	31.9	21.4
STL	**86.6**	**73.6**	**73.6**	**59.7**	**32.2**	21.2

4.6 Hyperparameter Selection

The main hyperparameters in STL are the thresholds to filter entities and triplets. We search thresholds from 0.1 to 0.9 to find the best threshold based on the average of three metrics mentioned in section 4.4. As shown in Figure 2, we get the same result when setting entity threshold to 0.1 and 0.2. 0.2 is selected as the optimal threshold to reduce noise.

To reduce the search cost, we search entity threshold first with a fixed relation threshold (set to 0.4 in our experiments), then search relation threshold with the optimal entity threshold.

4.7 Effectiveness of Rebalancing

The target of rebalancing is to promote the performance on minority classes. The results and ratios of different relation types are shown in Table 2. We choose classes whose proportion is less than 10% as minority classes. STL gains 1.0 average improvement on minority classes after adopting rebalancing strategy. However, we find that not all minority classes can benefit from rebalancing, which is an interesting problem for future work.

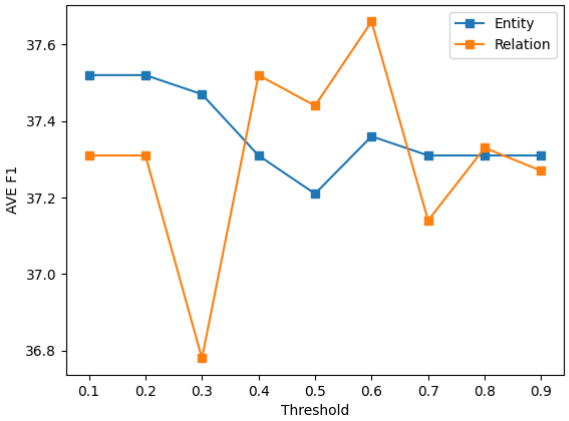

Figure 2: Average F1 of Three Metrics on SciERC with different entity an relation confidence threshold.

Table 2: F1 of RE for Each Class in SciERC.

Type	Method		Proportion
	w/o rebalancing	STL	
Used-for	31.0	32.2	52.5%
Feature-of	9.6	10.6	5.4%
Hyponym-of	37.6	36.8	9.3%
Evaluate-for	4.2	12.0	9.7%
Part-of	7.2	4.0	5.6%
Compare	3.8	3.8	5.2%
Conjunction	53.7	55.9	12.4%

5 CONCLUSION

We proposed a semi-supervised transfer learning framework for low resource entity and relation extraction in scientific domains. A rebalancing strategy is adopted to solve the class imbalance problem. Extensive experiments prove the effectiveness of the proposed method.

REFERENCES

[1] Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. 2018. Adversarial training for multi-context joint entity and relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2830–2836.

[2] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3615–3620.

[3] Shimin Di, Yanyan Shen, and Lei Chen. 2019. Relation extraction via domain-aware transfer learning. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & Data Mining*. 1348–1357.

[4] Markus Eberts and Adrian Ulges. 2019. Span-based joint entity and relation extraction with transformer pre-training. *arXiv preprint arXiv:1909.07755* (2019).

[5] Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of biomedical informatics* 45, 5 (2012), 885–892.

[6] Walid Hafiane, Joel Legrand, Yannick Toussaint, and Adrien Coulet. 2020. Experiments on transfer learning architectures for biomedical relation extraction. *arXiv preprint arXiv:2011.12380* (2020).

[7] Xuming Hu, Chenwei Zhang, Fukun Ma, Chenyao Liu, Lijie Wen, and S Yu Philip. 2021. Semi-supervised Relation Extraction via Incremental Meta Self-Training. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. 487–496.

[8] Xuming Hu, Chenwei Zhang, Yawen Yang, Xiaohe Li, Li Lin, Lijie Wen, and S Yu Philip. 2021. Gradient Imitation Reinforcement Learning for Low Resource Relation Extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 2737–2746.

[9] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language?. In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*.

[10] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36, 4 (2020), 1234–1240.

[11] Joël Legrand, Yannick Toussaint, Chedy Raïssi, and Adrien Coulet. 2021. Syntax-based transfer learning for the task of biomedical relation extraction. *Journal of Biomedical Semantics* 12, 1 (2021), 1–11.

[12] Hongtao Lin, Jun Yan, Meng Qu, and Xiang Ren. 2019. Learning dual retrieval module for semi-supervised relation extraction. In *The World Wide Web Conference*. 1073–1083.

[13] Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 3219–3232.

[14] Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: Tutorials*. 15–18.

[15] Henry Scudder. 1965. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory* 11, 3 (1965), 363–371.

[16] Bruno Taillé, Vincent Guigue, Geoffrey Scoutheeten, and Patrick Gallinari. 2020. Let's Stop Incorrect Comparisons in End-to-end Relation Extraction! *arXiv preprint arXiv:2009.10684* (2020).

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[18] Chen Wei, Kihyuk Sohn, Clayton Mellina, Alan Yuille, and Fan Yang. 2021. Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10857–10866.

[19] Wenxuan Zhou, Hongtao Lin, Bill Yuchen Lin, Ziqi Wang, Junyi Du, Leonardo Neves, and Xiang Ren. 2020. Nero: A neural rule grounding framework for label-efficient relation extraction. In *Proceedings of The Web Conference 2020*. 2166–2176.