Exploiting Background Knowledge and Inference for Scientific Relation Extraction
背景知識と推論を利用した科学文献からの関係抽出

TOHOKU
UNIVERSITY

Qin Dai
Graduate School of Information Sciences
Tohoku University

A thesis submitted for the degree of *Doctor of Information Sciences*

November 2019

List of Publications

Journal Paper (Refereed):

1. Qin Dai, Naoya Inoue, Paul Reisert, and Kentaro Inui. Leveraging Unannotated Texts for Scientific Relation Extraction. IEICE Transactions on Information and Systems, Vol. E101-D, No. 12, pp.3209-3217, December 2018.

International Conferences/Workshop Papers (Refereed):

1. Qin Dai, Naoya Inoue, Paul Reisert, Ryo Takahashi and Kentaro Inui. Incorporating Chains of Reasoning over Knowledge Graph for Distantly Supervised Biomedical Knowledge Acquisition. In Proceedings of the 33nd Pacific Asia Conference on Language, Information and Computing (PACLIC33), September 2019.

2. Qin Dai, Naoya Inoue, Paul Reisert, Ryo Takahashi and Kentaro Inui. Distantly Supervised Biomedical Knowledge Acquisition via Knowledge Graph Based Attention. In Proceedings of First Workshop on Extracting Structured Knowledge from Scientific Publications (ESSP), June 2019.

3. Qin Dai, Naoya Inoue, Paul Reisert and Kentaro Inui. Scientific Knowledge Acquisition via the Interaction between Relation Extraction and Knowledge Graph

Completion. In Proceedings of Third International Workshop on SCIentific DOCument Analysis (SCIDOCA), November 2018.

4. Qin Dai, Naoya Inoue, Paul Reisert and Kentaro Inui. Improving Scientific Relation Classification with Task Specific Supersense. In Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computing (PACLIC32), December 2018.

5. Qin Dai, Naoya Inoue, Paul Reisert, and Kentaro Inui. Leveraging Document-specific Information for Identifying Relations in Scientific Articles. In Proceedings of Second International Workshop on SCIentific DOCument Analysis (SCIDOCA), November 2017.

Other Publications(Not refereed):

1. Qin Dai, Naoya Inoue, Paul Reisert, Kentaro Inui. End-to-End Scientific Knowledge Graph Completion via Word Embedding based Entity Type Classification. In Proceedings of The 25th Annual Meeting of the Association for Natural Language Processing, March 2019.

Abstract

A tremendous amount of knowledge is present in the ever-growing scientific literature. In order to efficiently acquire such scientific knowledge, various computational tasks are proposed that train machines to read and analyze scientific documents automatically. One of these tasks, scientific Relation Extraction (RE), aims at automatically capturing scientific semantic relationships among entities in scientific documents. Conventionally, only a limited number of commonly used knowledge bases, such as Wikipedia, are used as a source of background knowledge for scientific RE. In this thesis, we hypothesize that unannotated scientific papers could also be utilized as a source of external background information for scientific RE. Based on the hypothesis, we propose several frameworks that are capable of extracting useful background information from unannotated scientific papers for scientific RE. Our experiments on different scientific corpus prove the effectiveness of the proposed frameworks on RE from scientific articles.

Although most RE frameworks, including ours, achieve reasonable performances, they require large and expensive manually annotated training data. To address this issue, distant supervision is proposed to automatically generate large amounts of labelled sentences via leveraging the alignment between knowledge graphs and texts. In recent years, many distantly supervised RE (DS-RE) frameworks use neural networks with attention mechanism to denoise the automatically labelled sentences and improve performances. To adjust the existing frameworks into scientific domain, we propose a new Knowledge Graph Completion model that significantly enhances our selected state-of-the-art DS-RE model on scientific dataset.

Beside the noise from distant supervision, the brevity of sentences in scientific papers could also hinder the performances of scientific DS-RE. Specifically, authors of scientific papers always omit the background elaboration that they assume is well known and easily inferred by their readers. However, the omitted background elaboration would be essential for a machine to identify relationships between entity pairs in scientific documents. To address this issue, in this thesis, we assume that the textual representation of reasoning paths (or inferences) between entity pairs over both scientific knowledge graph and multiple scientific documents could be utilized as the omitted explanation to fill the "gaps" in scientific documents and thus facilitate scientific DS-RE. Experimental results on biomedical datasets prove the effectiveness of our proposed model for scientific DS-RE, because the proposed model that incorporates the textual representation of reasoning paths achieves significant and consistent improvements as compared with state-of-the-art DS-RE baselines.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor Prof. Kentaro Inui for giving me the precious opportunity to follow my passion and carry out this research. This work would not have been possible without his persistent support and critical feedback. I would also like to extend my heartful gratitude to Dr. Naoya Inoue for his patience and inspiring discussions of this research, as well as all other helpful and motivational members (including former members) of the NLP Lab. Tohoku University, for the great research environment they created.

Furthermore, I would like to be grateful for the love and support from my family and relatives, especially my father, Manibadara, who always has confidence in me and offers me tons of encouragement and support, my younger sister, Sorgog, who always cheers me up and teaches me “when there is a will, there is a way”, and my uncle Chaoketu Gao and his family, who always make me feel at home in the foreign land. **Finally, I dedicate this thesis as an acknowledgment to my late mother, Yulan Gao, who played an integral role in my life so far, my late grandfather, D. and my late grandmother, B. for their immeasurable love and blessings since my childhood. I miss you a lot.**

This work was supported by JST CREST Grant Number JPMJCR1513, Japan and KAKENHI Grant Number 16H06614.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In recent years, scientific publications have become the largest repository of scientific knowledge ever and continue to increase at an unprecedented rate [50]. With the tremendous increase in the number of scientific papers, it is prohibitively time-consuming and laborious for researchers to review and fully-comprehend all papers. To help researchers effectively and quickly access a large amount of scientific papers and acquire useful knowledge, we need a good and practical Relation Extraction (RE) system to automatically recognize and extract useful knowledge from the ever growing scientific papers. For enhancing the scientific RE system, this thesis hypothesizes that it is important to leverage unannotated scientific papers and background knowledge based inferences.

1.1 The Importance of Unanotated Scientific Papers

In order to understand the scientific text and extract knowledge, there is a need to leverage the information that is not written in the given sentence, which we call here background knowledge. Suppose the following sentence[1]:

(1) $\overset{entity}{\underline{\boldsymbol{RTMs}}}_X$ *achieve top performance in automatic, accurate, and language independent* $\overset{entity}{\underline{\boldsymbol{prediction}}}_Y$ *of sentence-level and word-level statistical machine translation (SMT) quality.*

A scientific RE system is expected to extract the knowledge (or relation) APPLY_TO(*RTMs*, *prediction*), which means that *RTMs* is a system or method that is used for the task of *prediction*. For notational convenience, we refer to a sentence where a relation is extracted from as a *target sentence*, and we refer to the related entity pair as a *target entity* pair.

Without the support from background knowledge, such as "what the *RTMs* are" (e.g., "computational models" or "Research Team Members"), a scientific RE system may mistakenly identify the relation as PERFORM(*RTMs*, *prediction*), because if the target entity, *RTMs*, refers to "Research Team Members", it would be the Performers who PERFORM the task of *prediction*, rather than the applied tool for the task.

To address the lack of necessary background knowledge, this thesis hypothesizes that unlabelled scientific papers could be utilized as the source of background knowledge for scientific RE. For instance, from the scientific paper where the target sentence 1 is collected, we could find the following sentence about the target entity *RTMs*:

(2) *Referential translation machines (**RTMs**) provide a **computational model** for quality and semantic similarity judgments using retrieval of relevant training data ...*

Example 1 explicitly describes that the concept *RTMs* refers to the machines that could act as a *computational model*. Therefore, it is essential for a scientific RE system to

[1] This example is taken from W13-2242, ACL anthology (`http://aclanthology.info`).

exploit background knowledge (e.g., *RTMs* act as a computational model) from unlabelled scientific papers to disambiguate the relations (e.g., between PERFORM(*RTMs*, *prediction*) and APPLY_TO(*RTMs*, *prediction*)). There has been much previous work addressing scientific RE. However, most scientific RE systems usually use Wikipedia as the source of background knowledge, despite the high potential of the large number of scientific literatures.

1.2 The Importance of Inferences

Authors of scientific papers always leave out the background elaboration that they assume is well known and easily inferred by their readers. Suppose the following sentence[2]:

(3) *Efficacy and safety of single doses of intramuscular* $\underbrace{\overset{entity}{\textbf{\textit{ketorolac\_tromethamine}}}}_{X}$ *compared with meperidine for postoperative* $\underbrace{\overset{entity}{\textbf{\textit{pain}}}}_{Y}$.

Example 3 does not explain the background connection between *ketorolac_tromethamine* and *pain*, such as the mechanism or logical relationship between the target entity pair, and implicitly conveys that the former *may_treat* the latter. Scientific readers might easily make this assumption based on their inferences over the background knowledge about the target entity pair. However, for a machine, it would be extremely difficult to identify the relationship just from the given sentence without the important inference.

To address the issue of textual brevity in scientific documents, in this thesis, we assume that the inferences (or reasoning paths) between an entity pair over a collection

[2] This example is taken from PMID:2082312, MEDLINE corpus (`http://https://www.nlm.nih.gov/databases/download/pubmed_medline.html`).

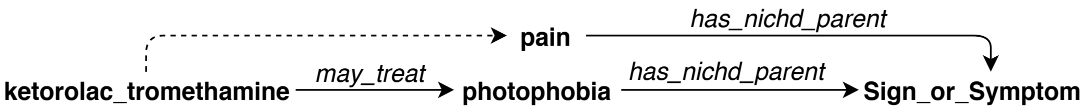


Figure 1.1: An example of reasoning path.

of background knowledge could be applied as the inference to fill the "gaps" and thereby improve the performance of scientific DS-RE. For instance, one reasoning path between *ketorolac_tromethamine* and *pain* is shown in Figure 7.1, where *has_hichd_parent* is similar to the hypernym relationship, the dotted arrow represents the target relation to be identified. By observing the path, we may infer with some likelihood that *may_treat*(*ketorolac_tromethamine*, *pain*), because *ketorolac_tromethamine* could be prescribed to treat some *Sign_or_Symptom* such as *photophobia*, and *pain* is a *Sign_or_Symptom*, therefore *ketorolac_tromethamine* might be used to treat *pain*. By comprehensively considering the path in Figure 7.1 and the sentence in Example 3, we could further prove the assumption. To this end, we propose a DS-RE model that not only encodes the target sentences, but also leverages the background knowledge based inferences, which are encoded as sequences of words.

1.3 Thesis Contributions

This thesis makes following main contributions.

- Exploiting the task specific supersense as a background knowledge for scientific RE, based on the distributional similarity learned from unannotated scientific papers. Experimental results prove the effectiveness of the task specific supersense for scientific RE because the proposed model significantly outperforms a baseline model and achieves competitive results to the state-of-the-art scientific

RE models.

- Developing a comprehensive framework for scientific RE which is capable of identifying relation via automatically collecting background knowledge from unlabelled scientific papers. Results indicate that, without supervision, the proposed model could effectively capture useful background knowledge from unannotated scientific papers, and improve the performances of scientific RE.

- Proposing a new Knowledge Graph Completion (KGC) model for scientific RE, based on the hypothesis that entity type is essential for calculating the plausibility of scientific knowledge. This model not only achieves better performance than most of the existing KGC models on scientific dataset, but also significantly enhances a selected state-of-the-art DS-RE model.

- Exploring the textual representation of inferences over a knowledge graph for scientific RE. Given a knowledge graph, this approach collects multiple shortest paths between a target entity as the background inferences for scientific DS-RE. Evaluations show that the inferences over a knowledge graph significantly outperforms a selected state-of-the-art baseline model.

- Exploring the textual representation of the reasoning paths across multiple documents for scientific RE. In this approach, textual documents are represented as a graph where entities are nodes of this graph while edges encode the textual relation between entity pairs. Shortest paths between a target entity pair are collected as the inferences (or reasoning paths) for scientific RE. Results not only indicate the effectiveness of the textual data based inferences for scientific DS-RE, but also prove the necessity of combining inferences over both knowledge base and multiple texts.

- Developing a novel framework which incorporates the inferences into a state-of-the-art DS-RE model. The proposed model applies Convolutional Neural Network (CNN) and knowledge graph embedding based attention mechanism to encode the inferences, which are represented as sequences of words. Results indicate that the proposed model significantly outperforms the selected baseline model. Furthermore, manual case study shows the proposed model is more capable of recognizing informative target sentences and plausible inferences.

To summarize, the contributions of this thesis are to study the methods for leveraging the large amount of unlabelled scientific publications and background knowledge based inferences for scientific knowledge acquisition.

1.4 Thesis Organization

The rest of the thesis is organized as follows:

- Chapter 2 provides a brief overview of the background in RE and neural networks based frameworks for knowledge acquisition, which includes word embeddings, KGC models and CNN.
- Chapter 3 introduces a new type of supersense (e.g., ANIMAL is a supersense of "*dog*") called task specific supersense for facilitating scientific RE. The task specific supersense could be dynamically defined according to the property of RE task (e.g., "the definitions of target relations in the given task"), and automatically identified via using a small number of seed instances and unlabelled scientific papers.

- Chapter 4 proposes a novel neural networks based framework that enables joint training of scientific relation classification and background knowledge detection from unlabelled scientific papers. This chapter empirically proves the robustness of the proposed model, and also indicates that it is effective and promising for scientific RE to leverage unlabelled scientific papers as the source of background knowledge.

- Chapter 5 proposes a novel framework based on the relationship between RE and KGC. The proposed framework utilizes a RE model to extract KG from collections of unannotated scientific papers, and uses the extracted KG to train a KGC model to learn KG embeddings. Finally, the proposed model extends the selected RE model with the learned KG embeddings. Experiments in this chapter prove the effectiveness of the proposed model on both scientific RE and KGC.

- Chapter 6 describes our work on applying a state-of-the-art Distantly Supervised RE (DS-RE) model on scientific domain. In this work, we focus on adapting the select model to scientific domain. Moreover, we propose a new Knowledge Graph Completion (KGC) model that not only out outperforms most of the existing KGC models, but also significantly enhances the performances of the selected DS-RE model on scientific dataset.

- Chapter 7, 8, and 9 address the task of building a joint DS-RE framework that can extract scientific knowledge via comprehensively considering Knowledge Graph (KG) embedding, multiple target sentences and background knowledge based inferences. We demonstrates that incorporating textual representation of KG based inferences and multi-text based inferences could significantly improve the performance of scientific RE. Moreover, we also observe that our proposed framework is not only capable of recognizing informative target sentences but plausible inferences.

- Chapter 10 concludes this thesis with discussions and presents our future work.

Chapter 2

Background

This chapter introduces central concepts of this thesis for better understanding its task formulation, methodology and real world application. As the first major topic, Section 1 overviews the research on Relation Extraction (RE). I begin with the introduction of the task of Relation Extraction. I then presents the two commonly used RE methods: Supervised RE and Distantly Supervised RE. I conclude this section with an overview of the deep neural network models that recently boost the performances of RE. As the second major topic, Section 2 reviews the basics for Knowledge Graph Completion (KGC) and introduces some representative KGC models, which includes TransE, TransD, ComplEx and SimplE.

2.1 Relation Extraction

Relation Extraction (RE) is the task of capturing predefined relations from text. A relation is a semantic relationship that holds between two or more entities. This thesis focuses on the binary relations, i.e., the relation that holds between two entities. Thus, the task of this thesis consists of the following: given a sentence that has been annotated with entity[3] mentions, we aim towards extracting relations between entities. Suppose the following sentence[4]:

(4) $\overset{entity}{\underline{\textit{This paper}}}$ $\overset{entity}{\underline{explores}}$ $\overset{entity}{\underline{several}}$ $\overset{entity}{\underline{\textit{unsupervised approaches}}}$ *to* $\overset{entity}{\underline{\textit{automatic keyword extraction}}}$
using $\overset{entity}{\underline{\textit{meeting transcripts.}}}$

In Example 4, one of the scientific relations we aim to extract is the relation INPUT(*meeting transcripts*, *automatic keyword extraction*), which means that *meeting transcripts* is the input data of the task of *automatic keyword extraction*. The task of RE for entity pairs can be seen as a classification task. Specifically, given all possible entity pair combinations from a target sentence, the task is to categorize each pair into relation types including predefined relations and non-relation. For example, in Example 4, given the pair (*meeting transcripts*, *automatic keyword extraction*), the output would be INPUT(*meeting transcripts*, *automatic keyword extraction*), while given the entity pair (*several*, *automatic keyword extraction*), it would be non-relation(*several*, *automatic keyword extraction*), which means that they do not belong to any predefined relations. With this level of fine-grained analysis, many applications, such as scientific question answering (QA) and scientific paper summarization, can benefit.

Evaluation Measures of RE includes precision, recall and F-score, which are evaluated based on a gold standard dataset. These measures are used to evaluate whether the relation instances (e.g., INPUT(*meeting transcripts*, *automatic keyword extraction*)) identified by a RE system are correct or incorrect. Precision, recall and F-score are calculated via Equation 2.1, 2.2 and 2.3 respectively, where "gold relations"

[3] In this thesis, *entity* refers not merely to concepts denoted by noun or noun phrase, it could be actions denoted by verb or verb phrase, and evaluation denoted by adjective or adverb etc.

[4] This example is taken from N09-1070, ACL anthology (`http://aclanthology.info`).

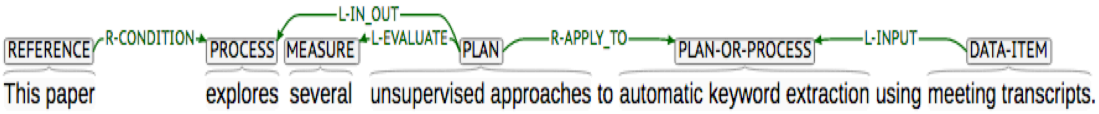


Figure 2.1: Examples of gold relations.

mean the annotated ground true relations as show in Figure 2.1, while “retrieved relations” represent the automatically identified relations.

$$\text{precision} = \frac{|\{\text{gold relations}\} \cap \{\text{retrieved relations}\}|}{|\{\text{retrieved relations}\}|} \tag{2.1}$$

$$\text{recall} = \frac{|\{\text{gold relations}\} \cap \{\text{retrieved relations}\}|}{|\{\text{gold relations}\}|} \tag{2.2}$$

$$\text{F-score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{2.3}$$

2.1.1 Supervised Relation Extraction

The typical supervised relation extraction is fully supervised, which means that a classification model is trained using an fully annotated gold dataset. For example, the fully annotated scientific dataset used here contains the example as shown in Figure 2.1, where entities and the relations among them are marked by a human annotator. The trained classifier is then applied on unseen target entity pairs such as (*bootstrapping methods*, *event extraction*) in Example 5, where the entity type (e.g., *PLAN*) of target entity has been provided. Supervised relation extraction is a hot field in natural language processing since rich annotated corpus are released. However, manually annotating gold dataset is expensive and time-consuming. This would become worse especially when gold dataset needs to be created for a new domain of interest [63]. For instance, the

LOCATED_IN relation might be differently expressed in the newswire domain than the biomedical domain. Due to the limitation, much research has focused on the methods of more inexpensively producing training data. One of the representative approaches is distantly supervised relation extraction.

(5) *This paper investigates two kinds of* $\overset{PLAN}{\underline{\textbf{\textit{bootstrapping methods}}}}_X$ *used for* $\overset{PLAN}{\underline{\textbf{\textit{event extraction}}}}_Y$*...*

2.1.2 Distantly Supervised Relation Extraction

As mentioned, one obstacle that is encountered when building a RE system is the generation of training instances. For coping with this difficulty, [48] proposes distant supervision to automatically generate training samples via leveraging the alignment between Knowledge Bases (KBs) and texts. They assumes that if two entities are connected by a relation in a KB, then all sentences that contain these entity pairs will express the relation. For instance, *may_treat*(*aspirin*, *pain*) is a relation in a biomedical KB. Distant supervision will automatically label all sentences, such as Example 6, Example 7 and Example 8, as positive instances for the relation *may_treat* and use the labelled examples to train a relation classifier as supervised learning. Although distant supervision could provide a large amount of training data at low cost, it always suffers from wrong labelling problem. For instance, comparing to Example 6, Example 7 and Example 8 should not be seen as the evidences to support the *may_treat* relationship between *aspirin* and *pain*, but will still be annotated as positive instances by the distant supervision.

(6) *The clinical manifestations are generally typical nocturnal* ***pain*** *that prevents sleep and that is alleviated with* ***aspirin****.*

(7) *The tumor was remarkably large in size , and* ***pain*** *unrelieved by* ***aspirin****.*

(8) *The level of* ***pain*** *did not change significantly with either* ***aspirin*** *or pentoxifylline , but the walking distance was farther with the pentoxifylline group .*

To automatically alleviate the wrong labelling problem, [55, 28] apply multi-instance learning, which assumes that given a related entity pair (e.g., (*aspirin*, *pain*)), only at-least-one automatically-labelled sentence could express their relation in the KB. Recently, the deep neural networks with attention mechanism are applied to effectively extract features from all of the collected sentences by calculating their contribution (e.g., Example 7 contributes more to identify the relation *may_treat*(*aspirin*, *pain*) than Example 8) [41, 26, 17].

2.2 Deep Neural Networks

In recent years, Deep Neural Networks have revolutionized many application domains of Natural Language Processing (NLP), including machine translation, sentiment analysis and relation extraction. The advantage of deep neural networks is that they are capable of automatically learning representation from raw and complex data such as characters, words and sentences as features. Learned representations often perform much better than the handcrafted feature engineering. This section introduces the building blocks of deep neural networks that are prevalent in NLP: word embeddings and Convolutional Neural Networks (CNNs).

2.2.1 Word Embeddings

Word embeddings are utilized as the input of Deep Neural Networks in NLP, rather than the actual characters or words. The method of word embeddings projects per word in the vocabulary into a real-valued vector space with low dimensionality. The learning of word embeddings is inspired by the linguistic theory of distributional semantic that words appearing in similar contexts tend to have similar semantics. One popular algorithm of word embeddings is called skip-gram [46], which becomes the inspiration of other word embedding algorithms such as GloVe [52] and fasttext [6].

Skip-grap algorithm tries to predict context words (w) that appear around center word (c) within a window size of M. Specifically, skip-gram algorithm optimizes the log probability of observed data:

$$\frac{1}{T}\sum_{i=1}^{T}\sum_{-M\leqslant m\leqslant M, m\neq 0}\log p(w_{i+m}|w_i) \tag{2.4}$$

In Equation 2.4, T represents the number of tokens in training data, M denotes the number of context tokens around the target word w_i. $p(w_{i+m}|w_i)$ is modeled by the softmax function (Equation 2.5), where $\mathbf{u}_w$ and $\mathbf{v}_w$ respectively denote the context and target vector for word w.

$$p(w|c) = \frac{\exp(\mathbf{u}_w^T\mathbf{v}_c)}{\sum_{i=1}^{n}\exp(\mathbf{u}_i^T\mathbf{v}_c)} \tag{2.5}$$

2.2.2 Convolutional Neural Networks

Convolutional neural networks (CNNs) are a type of feed forward artificial neural networks whose main components include convolution operation and pooling operation. Recently, with the prevalence of deep neural networks, CNNs has been effectively applied on RE. A representative CNNs for RE, as shown in Figure 2.2, consists of four main layers: (i) embeddings layer to encode words in sentences into real-valued vectors, (ii) the convolutional layer to generate n-gram level feature, (iii) the pooling layer to determine the most informative features and (iv) a logistic regression layer(a fully connected neural network with a softmax at the end) to perform relation classification.

Embedding layer is calculated via Equations 2.6-2.9, where W^{w}_{emb} is a word embedding projection matrix, W^{et}_{emb} is an entity type (ET) projection matrix, x^{w}_{t} is a one-hot word representation, and x^{et}_{t} is a one-hot entity type representation. The position vector e^{wp}_{t} encodes the relative distance between the current word and the head of target entity pair. For instance, in Example 9, the relative distance of the word *"for"* is [-1, 2].

(9) $\overset{entity}{\underline{We}}$ $\overset{entity}{\underline{introduce}}$ $\overset{entity}{\underline{referential\ translation\ machines}}$ $(\overset{entity}{\underline{RTM}}_{A})$ *for* $\overset{entity}{\underline{quality\ estimation}}_{B}$ ...

This relative distance will be encoded into position vectors e^{wp1}_{t} and e^{wp2}_{t}, respectively, via Equation 2.8, where W^{wp}_{emb} is a word position embedding projection matrix and x^{wp}_{t} is a one-hot representation of the relative distance. Word embedding e^{w}_{t}, entity type embedding e^{et}_{t} and word position embedding e^{wp1}_{t} and e^{wp1}_{t} are concatenated to create the final word representation e_t.

$$e^{w}_{t} = W^{w}_{emb} x^{w}_{t} \tag{2.6}$$

$$e^{et}_{t} = W^{et}_{emb} x^{et}_{t} \tag{2.7}$$

$$e^{wp}_{t} = W^{wp}_{emb} x^{wp}_{t} \tag{2.8}$$

$$e_t = concat(e^{w}_{t}, e^{et}_{t}, e^{wp1}_{t}, e^{wp2}_{t}) \tag{2.9}$$

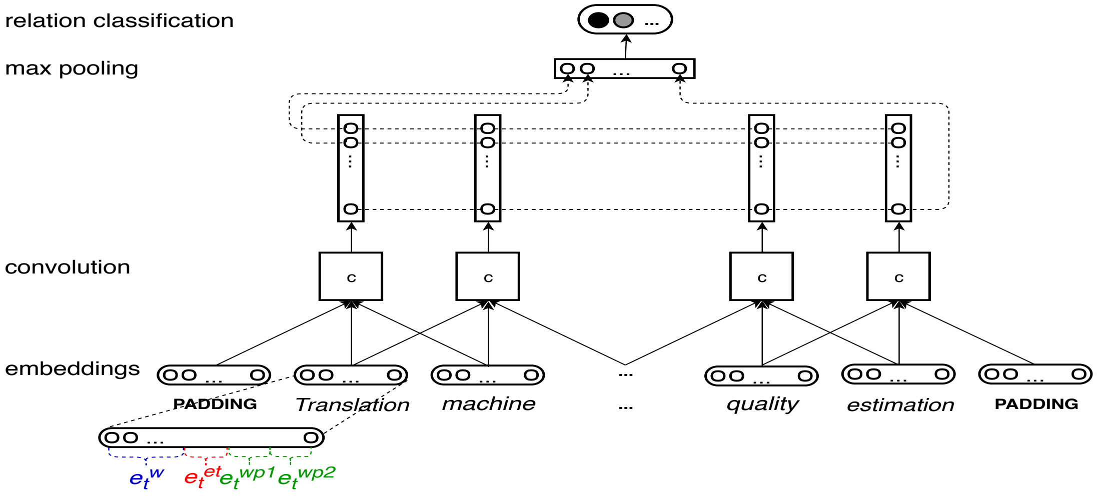


Figure 2.2: CNNs architecture

$$z_t = concat(e_{t-(k-1)/2}, ..., e_{t+(k-1)/2}) \tag{2.10}$$

$$h_t = tanh(Wz_t + b) \tag{2.11}$$

Convolutional layer generates a n-gram level vector h_t. h_t is calculated by Equations 2.10 and 2.11, where z_t is the concatenated embedding of k words in the convolutional window, k is convolutional window size, and W is the weight matrix of the convolutional layer. In order to address the issue of referencing words with indices outside the sentence boundaries, the target sentence is padded with a special **PADDING** token $(k-1)/2$ times at the beginning and the end.

Max pooling layer chooses the maximum value from each dimension of the n-gram level feature and merges them as the sentence level feature r via Equation 2.12, where i indexes feature dimensions, M is the number of feature dimensions.

$$r_i = \max_t\{(h_t)_i\}, \forall i = 1, ..., M \tag{2.12}$$

Logistic regression layer predicts the semantic relationship between a target entity pair in a target sentence x, by computing the score for a class label $c \in C$ via dot product:

$$S_\theta(x)_c = r^T[W^{class}]_c, \tag{2.13}$$

where C is a set of predefined semantic relationships, r is the sentence level feature vector, and W^{class} is the class embedding matrix. The column of W^{class} represents the distributed vector representation of different class labels.

2.3 Knowledge Graph Completion

Knowledge Graphs (KGs), such as Freebase [7] and DBpedia [38], provide large collections of relations between entities, typically stored as (h, r, t) triples, where h = head entity, r = relation and t = tail entity, e.g., (*Tokyo*, *capitalOf*, *Japan*). As distinguished from the task of RE which constructs KGs from raw text, Knowledge Graph Completion (KGC) automatically infers missing facts by examining the latent regularities in existing ones. For example, suppose the triples (*SVM*, APPLY_TO, *recognition*) and (*SVM*, be_INPUT[1], *microblog*) are stored in a KB, as shown in Table 2.1, based on the fact, a KBC model would infer the new plausible triple (*SVM*, APPLY_TO, *classification*) rather than (*SVM*, APPLY_TO, *corpus*), because entity *classification* and entity *recognition* share some latent semantic features.

The latent semantic features are represented by KB embedding, which embeds triple of KB into a continuous vector space, so as to decompose the observed triples into a product of vectors. For a given fact triple (h, r, t) in which head entity h is linked

[1] where (h, be_INPUT, t) equals (t, INPUT, h).

head entity	relation	tail entity
SVM	APPLY_TO	*recognition*
SVM	be_INPUT	*microblog*
SVM	?	*classification*
SVM	?	*corpus*

Table 2.1: Instances for Scientific KGC.

to tail entity t through relation r, the score of plausibility can then be recovered as a multi-linear product between the embedding vectors of h, r and t.

Suppose we have a KG containing a set of fact triplets $O = \{(h, r, t)\}$, where each fact triplet consists of two entities $h, t \in \mathcal{E}$ and their relation $r \in \mathcal{R}$. Here $\mathcal{E}$ and $\mathcal{R}$ stand for the set of entities and relations respectively. KGC model then encodes $h, t \in \mathcal{E}$ and their relation $r \in \mathcal{R}$ into low-dimensional vectors $\mathbf{h}$, $\mathbf{t} \in R^d$ and $\mathbf{r} \in R^d$ respectively, where d is the dimensionality of the embedding space. KGC models define a scoring function $f_r(h, t)$ to evaluate the plausibility of a given fact triplet (h, r, t). The goal of KGC models is to define an effective scoring fuction so that the score of a correct triplet $f_r(h, t)$ is higher than the score of an incorrect triplet $f_r(h', t')$. KGC models minimize a loss function to learn the model parameters (i.e., entity vectors, relation vectors and matrices). The margin-based pairwise ranking loss [8] that defined via Equation 2.14 is conventionally used as the loss function for KGC models.

$$L = \sum_{(h,r,t)\in O,(h',r,t')\in O'} [\gamma - f_r(h, t) + f_r(h', t')]_+ \tag{2.14}$$

In Equation 2.14, $[x]_+ = \max(0, x)$, γ is the margin hyperparameter, O' denotes the set of incorrect triplets obtained by corrupting the set of correct triplets O. This section introduces four representative KGC models, which are TransE [8], TransD [30], ComplEx [68] and SimplE [32].

2.3.1 TransE

Given a fact triplet (h,r,t), TransE then encodes entities h, t and relation r into a real-valued vector $\mathbf{h} \in R^d$, $\mathbf{t} \in R^d$ and $\mathbf{r} \in R^d$ respectively. TransE defines the scoring function via the Equation 2.15.

$$f_r(h,t) = -||\mathbf{h} + \mathbf{r} - \mathbf{t}||_{1/2} \tag{2.15}$$

The score evaluates the distance between $\mathbf{h} + \mathbf{r}$ and $\mathbf{t}$, which is expected to be large if (h, r, t) holds.

2.3.2 TransD

TransD is an extension of TransE and introduces additional mapping vectors $\mathbf{h}_p$, $\mathbf{t}_p$ $\in R^d$ and $\mathbf{r}_p \in R^d$ for h, t and r respectively. TransD defines the scoring function via the Equation 2.16, where $\mathbf{M}_{rh}$ and $\mathbf{M}_{rt}$ are projection matrices for mapping entity embeddings into relation specific spaces.

$$f_r(h,t) = -||\mathbf{h}_r + \mathbf{r} - \mathbf{t}_r||_{1/2} \tag{2.16}$$

$$\begin{aligned}
\mathbf{h}_r &= \mathbf{M}_{rh}\mathbf{h}, \\
\mathbf{t}_r &= \mathbf{M}_{rt}\mathbf{t}, \\
\mathbf{M}_{rh} &= \mathbf{r}_p\mathbf{h}_p^\top + \mathbf{I}^{d\times d}, \\
\mathbf{M}_{rt} &= \mathbf{r}_p\mathbf{t}_p^\top + \mathbf{I}^{d\times d}
\end{aligned}$$

2.3.3 ComplEx

Given a fact triplet (h, r, t), ComplEx then encodes entities h, t and relation r into a complex-valued vector $\mathbf{h} \in \mathbb{C}^d$, $\mathbf{t} \in \mathbb{C}^d$ and $\mathbf{r} \in \mathbb{C}^d$ respectively, where d is the dimensionality of the embedding space. Since entities and relations are represented as complex-valued vector, each $\mathbf{x} \in \mathbb{C}^d$ consists of a real vector component $Re(\mathbf{x})$ and imaginary vector component $Im(\mathbf{x})$, namely $\mathbf{x} = Re(\mathbf{x}) + iIm(\mathbf{x})$. The KG scoring function of ComplEx for a fact triplet (h, r, t) is calculated via Equation 6.11, where $\bar{\mathbf{t}}$ is the conjugate of $\mathbf{t}$; $Re(\cdot)$ (or $Im(\cdot)$) means taking the real (or imaginary) part of a complex value. $\langle u, v, w\rangle$ is defined via Equation 6.12, where $[\cdot]_n$ is the n-th entry of a vector.

$$\begin{aligned} f_r(h,t) = Re(\langle \mathbf{h}, \mathbf{r}, \bar{\mathbf{t}}\rangle) = \\ \langle Re(\mathbf{r}), Re(\mathbf{h}), Re(\mathbf{t})\rangle \\ +\langle Re(\mathbf{r}), Im(\mathbf{h}), Im(\mathbf{t})\rangle \\ +\langle Im(\mathbf{r}), Re(\mathbf{h}), Im(\mathbf{t})\rangle \\ -\langle Im(\mathbf{r}), Im(\mathbf{h}), Re(\mathbf{t})\rangle \end{aligned} \tag{2.17}$$

$$\langle \mathbf{u}, \mathbf{v}, \mathbf{w}\rangle = \sum_{n=1}^{d} [\mathbf{u}]_n [\mathbf{v}]_n [\mathbf{w}]_n \tag{2.18}$$

Since the asymmetry of this scoring function, namely $f_r(h,t) \neq f_r(t,h)$, ComplEx can effectively encode asymmetric relations [68].

2.3.4 SimplE

Given a fact triplet (e_1, r, e_2), SimplE then encodes each entity $e \in \mathcal{E}$ into two vectors $\mathbf{h}_e$, $\mathbf{t}_e \in R^d$ and each relation $r \in \mathcal{R}$ into two vectors $\mathbf{v}_r$, $\mathbf{v}_{r^{-1}} \in R^d$ respectively, where d is the dimensionality of the embedding space. $\mathbf{h}_e$ captures the entity e's behaviour as the *head entity* of a fact triplet and $\mathbf{t}_e$ captures e's behaviour as the *tail entity*. $\mathbf{v}_r$ represents r in a fact triplet (e_1, r, e_2), while $\mathbf{v}_{r^{-1}}$ represents its inverse relation r^{-1} in the triplet (e_2, r^{-1}, e_1). The KG scoring function of SimplE for a fact triplet (e_1, r, e_2) is defined via Equation 6.14.

$$f_r(e_1, e_2) = \frac{1}{2}(\langle \mathbf{h}_{e_1}, \mathbf{v}_r, \mathbf{t}_{e_2} \rangle + \langle \mathbf{h}_{e_2}, \mathbf{v}_{r^{-1}}, \mathbf{t}_{e_1} \rangle) \tag{2.19}$$

Chapter 3

Improving Scientific Relation Extraction with Task Specific Supersense

3.1 Introduction

In this chapter, we propose a new semantic category called the task specific supersense (TSS) for a given RE or Relation Classification (RC). TSS is defined according to the property of a given Relation Classification (RC) task, which includes the definitions of target relations and selectional tendency of target relations. We hypothesize that TSS can be utilized to improve the performance of scientific RC[1].

Suppose the following target sentence taken from the SemEval-2018 task 7 dataset [20]:

(10) *This paper presents a* $\overset{\textit{entity}}{\underline{\textit{critical discussion}}}_X$ *of the various approaches that have*

[1] In this chapter, RE and RC are used interchangeably.

been used in the $\underline{\overset{entity}{\textit{evaluation of Natural Language systems}}}_{Y}$.

In this dataset, the entity mentions are annotated but their types are not tagged. This task asks a RC system to classify the target entity pair into several predefined semantic relations. One of them is TOPIC relation. The relation TOPIC(X, Y) namely means the entity X deals with the topic Y. Therefore, the entity X tends to be a research activity, such as "*analysis*", "*survey*" and "*discussion*" etc. Based on this selectional tendency, we define a TSS to cover these words, called *RESEARCH-PROCESS*. Identifying *RESEARCH-PROCESS* for a given word such as "*discussion*" in Example 19, could help a RC system to correctly classify the target entity pair into TOPIC relation.

Similarly, suppose the following target sentences from the RANIS dataset [66]:

(11) *A* $\underline{\overset{DATA-ITEM}{\textit{verb}}}$ *'s* $\underline{\overset{DATA-ITEM}{\textit{aspectual category}}}_{Y}$ *can be* $\underline{\overset{PROCESS}{\textit{predicted}}}_{X}$ *...*

(12) *...* $\underline{\overset{PLAND-OR-PROCESS}{\textit{statistical generation}}}$*to* $\underline{\overset{PROCESS}{\textit{combine}}}_{X}$ $\underline{\overset{DATA-ITEM}{\textit{common phrases}}}_{Y}$ *into a* $\underline{\overset{DATA-ITEM}{\textit{sentence}}}$.

In this dataset, both entity mentions and entity types (e.g., *PROCESS*, *PLAN*, *DATA-ITEM*) are annotated. The target relations includes relation OUTPUT(X, Y) (as in Example 11), and INPUT(X, Y) (as in Example 12). They namely mean entity Y is the output/input of a process X. Based on the definition, we propose a TSS called *OUTPUT-PROCESS*, verbs like "*show*", "*identify*" and "*extract*" belong to this TSS, because "*a system can show/identify/extract Y*" represents that the system can output Y. If we could correctly identify the *OUTPUT-PROCESS* in a given target sentence, and apply the new specific TSS , it could help a RC system more effectively identify OUTPUT relation, in comparison with only using the original general entity type, *PROCESS*. For instances, in Example 11 and Example 12, both target entities "*predicted*" and "*combine*" belong to the same entity type, *PROCESS*, but the former specifically belongs to the TSS, *OUTPUT-PROCESS*, and the latter does not. Therefore, based on this difference, a RC system could easily distinguish them, and classify the former as OUTPUT relation.

For identifying the TSS, one possibility is to manually annotate the TSS in target sentences. However, manual annotation is time-consuming [33] and expensive [2]. To address this issue, in this work, we propose a minimally supervised approach that utilizes supersense embeddings. Specifically, we manually prepare a small number of seed instance words for the predefined supersense (or TSS) (e.g., "*survey*" for *RESEARCH-PROCESS*) and train the embedding of word and supersense in the same vector space, like the method Flekova and Gurevych [19] proposed, which will be detailed in Section 3.3. By comparing the emebdding between supersense and a given word, we determine its TSS. Our evaluation empirically demonstrates that incorporating the TSS could improve the performance of scientific RC.

3.2 Related Work

Conventional approaches for RC rely on human-designed, complex lexical-syntactic patterns [9], statistical co-occurrences [65] and structuralized knowledge bases such as WordNet [24, 10]. In recent years, exploring Neural Network (NN)-based models has been the dominant approach in the field. Zeng et al. [79] and Xu et al. [77] proposed a Convolutional Neural Network (CNN)-based framework, which depends on sentence-level features collected from an entire target sentence and lexical-level features from lexical resources such as WordNet [18]. Santos et al. [61] proposed a ranking CNN model, which is trained by a pairwise ranking loss function. To improve the ability of sequential modeling, Zhang et al. [81] proposed a recurrent neural network (RNN)-

based model for RC. Other variants of RNN-based models have been proposed, such as Miwa et al. [49], who proposed a bidirectional tree-structured LSTM model.

Additionally, similar NN-based approaches are used in scientific relation classification. For instance, Gu et al. [22] utilized a CNN-based model for identifying *chemical-disease* relations from the abstracts of MEDLINE papers. Hahn-Powell et al. [25] proposed an LSTM-based RNN model for identifying *causal precedence* relationship between two event mentions in biomedical papers. Ammar et al. [1] enhanced Miwa and Bansal [49]'s relation extraction model via extensions such as gazetteer-like information extracted from Wikipedia. Pratap et al. [53] incorporate WordNet hypernyms as the feature for scientific RC. However, none of these approaches leverage the task specific supersense for RC.

Flekova and Gurevych [19] integrated supersense into distributional word representation, and trained supersense embedding and word embedding in the same vector space. They used the similarity between supersense embedding and word embedding as a feature to identify supersense. We applied the similar approach to tag the TSS to enhance the performance of scientific RC.

3.3 Task Specific Supersense Embedding

3.3.1 Preparing Seed TSS Instances

To learn the TSS embedding, we firstly define a TSS according to the property of a given task, such as what kinds of relation are in the given task, what is the definition of the target relation, what type of entity tends to participate in the target relation, etc, as discussed before. We test our hypothesis on different RC tasks in the computational

TSS	**Seed Instances**
SYSTEM or METHOD	*parser, system, learner, decoder, technology, ...*
RESEARCH-PROCESS	*analyze, investigate, study, survey, trial, ...*
OUTPUT-PROCESS	*describe, show, learn, provide, achieve, ...*
INPUT-PROCESS	*combine, compare, convert, transform, divide, ...*

Table 3.1: TSS and corresponding seed instances

linguistic domain in which some RC task, like SemEval-2018 task 7 [20], aims to classify relations, such as USAGE, TOPIC and MEDOL-FEATURE, and other task, like RC on RANIS dataset [66], asks for identifying relations such as INPUT and OUTPUT. Therefore, we come up with four [2] types of TSS, as shown in the first column of Table 3.1, for distinguishing these relations for a given specific task. For instance, tagging *SYSTEM or METHOD* in target sentences could help USAGE relation recognition. After figuring out TSS for a given RC task , we manually prepare a small number of seed instances for the predefined TSS as shown in the second column of Table 3.1.

3.3.2 Building TSS Embeddings

Similar to the method proposed by Flekova and Gurevych [19], we replace each word in a corpus by its corresponding TSS according to seed instances prepared in the previous step. In this way, besides the original corpus (see Table 3.2, first row), we obtain an alternative corpus where each word is replaced by its corresponding TSS (see Table 3.2, second row). We trained the TSS embeddings on the ACL Anthology Reference Corpus [5] and its alternative corpus jointly (e.g., both first and second row in Table 3.2) by the skip-gram NN architecture made available by the Gensim word2vec

[2] As a preliminary study, we only select four representative types of TSS, but in the future, we will investigate more types of TSS for scientific RC.

1	*In the above example , three different* ***analyses*** *have been found.* *Ribas (1994a) reported experimental results obtained from the application of the above technique to* ***learn*** *SRs.*
2	*In the above example , three different RESEARCH-PROCESS have been found.* *Ribas (1994a) reported experimental results obtained from the application of the above technique to OUTPUT-PROCESS SRs.*

Table 3.2: Example of original corpus (1) and alternative corpus (2)

TSS	
SYSTEM or METHOD	*model, models, system, approach, algorithm method, parser, framework, classifier, module*
RESEARCH-PROCESS	*study, work, research, analysis, investigation, experiment, experiments, studies, paper, investigations*
OUTPUT-PROCESS	*obtain, derive, find, provide, describe, give, show, generate, introduce, demonstrate*
INPUT-PROCESS	*compare, combine, integrate, evaluate, convert, incorporate, augment, analyze, transform, apply*

Table 3.3: Top 10 most similar word embeddings for each TSS embedding

tool [3]. Thereby, we produce continuous representation of words and the predefined TSS in one vector space [4]. Table 3.3 shows the most similar word to each of the predefined TSS based on their embeddings' cosine similarity.

3.3.3 Identifying TSS for Given Words

Since the TSS is positioned in the same vector space with original words, we could utilize the embedding cosine similarity between TSS and given words to determine their TSS. Specifically, we tag a given word with the TSS, if the cosine similarity is above a predefined threshold score [5]. For instance, given a target sentence Example 13,

[3] `https://radimrehurek.com/gensim`

[4] The embedding is trained with negative sampling of 25 noise words, minimal word frequency of 10, window size of 2 and alpha of 0.0025, using 15 epochs to generate 300-dimensional vectors.

[5] We set the threshold score as 0.5 for identifying TSS in SemEval2018 Task7 datasets, and set it as 0.3 for RANIS dataset.

Figure 3.1: TSS identification example, where NONE means the word does not belong to any TSS. SYSMETH and INPRO stand for *SYSTEM or METHOD* and *INPUT-PROCESS* respectively.

the TSS identification result would be Figure 3.1.

(13) *large vocabulary continuous speech recognition (LVCSR) , a unified framework based approach is introduced to exploit multi-level linguistic knowledge*

3.4 Proposed Model

3.4.1 Task Setting

In this chapter, we create a task setting where, given definitions of target relations and collections of unannotated scientific papers, we come up with a new entity type called TSS and train TSS embedding on the raw corpus. Based on the embedding cosine similarity between TSS and a given word, we identify the TSS, and incorporate the TSS information into a state-of-the-art RC model, thereby improve its performance on scientific RC. We execute the problem setting in computational linguistic domain, but we believe that this setting can provide useful guide to other domains, such as RC in biomedical domain.

3.4.2 Base Model

We choose the RC model that is proposed by Santos et al. [61] as our base RE model, since it is simple and strong. As shown in Figure 5.3, it is composed of three layers. The first layer is an embedding layer, which maps each word of the target sentence

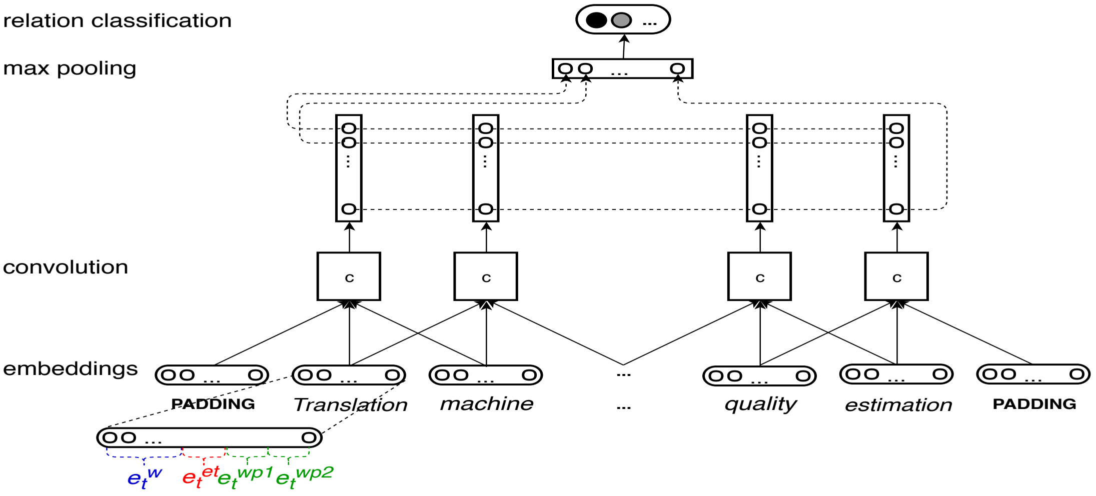


Figure 3.2: Base model architecture

into a low-dimensional word vector representation. The embedding layer is calculated via Equations 6.5-5.6, where W^w_{emb} is a word embedding projection matrix, W^{et}_{emb} is an entity type (ET) projection matrix, x^w_t is a one-hot word representation and x^{et}_t is a one-hot entity type representation. The position vector e^{wp}_t encodes the relative distance between the current word and the head of target entity pair. For instance, in Example 29, the relative distance of the word *"for"* is [-1, 2].

(14) *We introduce referential translation machines* ($\overset{entity}{\underline{RTM}}_A$) *for* $\overset{entity}{\underline{quality\ estimation}}_B$ ...

This relative distance will be encoded into position vectors e^{wp1}_t and e^{wp2}_t, respectively, via Equation 5.5, where W^{wp}_{emb} is a word position embedding projection matrix and x^{wp}_t is a one-hot representation of the relative distance. Word embedding e^w_t, entity type embedding e^{et}_t and word position embedding e^{wp1}_t and e^{wp1}_t are concatenated to create the final word representation e_t. If the dataset does not have entity type information, like SemEval-2018 Task 7 dataset, e^{et}_t will be ignored.

$$e_t^w = W_{emb}^w x_t^w \tag{3.1}$$

$$e_t^{et} = W_{emb}^{et} x_t^{et} \tag{3.2}$$

$$e_t^{wp} = W_{emb}^{wp} x_t^{wp} \tag{3.3}$$

$$e_t = concat(e_t^w, e_t^{et}, e_t^{wp1}, e_t^{wp2}) \tag{3.4}$$

$$z_t = concat(e_{t-(k-1)/2}, ..., e_{t+(k-1)/2}) \tag{3.5}$$

$$h_t = tanh(W z_t + b) \tag{3.6}$$

The next layer is a convolutional layer, which generates a distributed convolutional window level vector h_t. h_t is calculated by Equations 5.7 and 5.8, where z_t is the concatenated embedding of k words in the convolutional window, k is convolutional window size, and W is the weight matrix of the convolutional layer. In order to address the issue of referencing words with indices outside the sentence boundaries, the target sentence is padded with a special **PADDING** token $(k-1)/2$ times at the beginning and the end.

The third layer is a max pooling layer, which chooses the maximum value from each dimension of the convolutional window level feature and merges them as the sentence level feature r via Equation 6.6, where i indexes feature dimensions, M is the number of feature dimensions.

$$r_i = \max_t \{(h_t)_i\}, \; \forall i = 1, ..., M \tag{3.7}$$

Finally, the model predicts the semantic relationship between a target entity pair in a target sentence x, by computing the score for a class label $c \in C$ via dot product:

$$S_\theta(x)_c = r^T[W^{class}]_c \tag{3.8}$$

where C is a set of predefined semantic relationships, r is the sentence level feature vector, and W^{class} is the class embedding matrix. The column of W^{class} represents the distributed vector representation of different class labels. It is worth mentioning that the model uses a logistic loss function, as shown in Equation 5.11:

$$\begin{aligned} L = log(1 + exp(\gamma(m^+ - s_\theta(x)_{y^+})) \\ +log(1 + exp(\gamma(m^- + s_\theta(x)_{c^-})) \end{aligned} \tag{3.9}$$

where $s_\theta(x)_{y^+}$ is the score of correct class label, $s_\theta(x)_{c^-}$ is the score of the most competitive incorrect class label, m^+ and m^- are margins, and γ is a scaling factor. In our experiment, we use $m^+ = 2.5, m^- = 0.5$ and $\gamma = 2$.

3.4.3 Incorporating TSS

We incorporate TSS information via Equations 3.10-3.11, where W^{tss}_{emb} is an TSS projection matrix, and x^{tss}_t is a one-hot TSS representation.

$$e^{tss}_t = W^{tss}_{emb} x^{tss}_t \tag{3.10}$$

$$e_t = concat(e^w_t, e^{et}_t, e^{tss}_t, e^{wp1}_t, e^{wp2}_t) \tag{3.11}$$

3.5 Data

3.5.1 SemEval-2018 Task 7 dataset

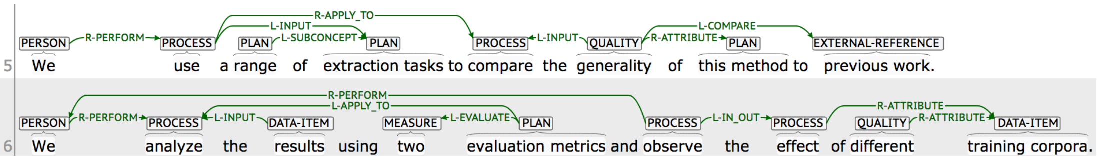
We evaluate the effectiveness of TSS for scientific RC on three different datasets. The first and second dataset we use in evaluation are the SemEval-2018 Task 7.1.1 & 7.1.2 datasets [20], which are in computational linguistic domain. This task handles 6 semantic relations in scientific paper abstracts. The datasets of subtasks 1.1 and 1.2 contains titles and abstracts of papers where entity mentions are either manually annotated (Subtask 1.1), as Example 15, or automatically annotated (Subtask 1.2), as Example 16. The target semantic relations in dataset 1.1 and 1.2 are manually annotated. There are 1228/1248 training examples and 355/255 testing examples in dataset 1.1/1.2. These samples are classified into one of the following semantic relations: USAGE, RESULT, MODEL-FEATURE, PART-WHOLE, TOPIC, COMPARISON. The official evaluation metric is macro-F1 score.

(15) Recently the LATL has undertaken the development of a <entity id="L08-1579.1">multilingual translation system</entity> based on a <entity id="L08-1579.2">symbolic parsing technology</entity> (...)

(16) The aim of this <entity id="L08-1239.17">paper</entity> is at investigating the <entity id="L08-1239.18">relationships</entity> (...)

3.5.2 RANIS dataset

The third dataset we use is RANIS corpus [66], a collection of computer science paper abstracts. The type of entity (referred to as Entity Type (ET) hereafter) and



5 We use a range of extraction tasks to compare the generality of this method to previous work.

6 We analyze the results using two evaluation metrics and observe the effect of different training corpora.

Figure 3.3: Annotation example shown in brat rapid annotation tool. To more clearly illustrate the direction of relation, we add directional tag “L-” and “R-” before each relation tag.

domain specific relation in the RANIS corpus has already been annotated with the annotation scheme proposed by [66], as Figure 5.4. The dataset consists of ETs such as QUALITY, PROCESS and DATA-ITEM and domain specific scientific relations, such as INPUT, OUTPUT and APPLY-TO. In total, the RANIS corpus contains 250 abstracts collected from ACL Anthology (230 abstracts in the development set and 20 abstracts in the test set) and 150 abstracts collected from ACM Digital Library. For training and testing our proposed model, we only use the 250 abstracts from ACL Anthology. From the ACL Anthology abstracts, we extract 11,520 examples from the development set of ACL Anthology and 1,142 examples from the test set of ACL Anthology. These instances are classified into one of the following semantic relations: ORIGIN, COMPARE, EQUIVALENCE, TARGET, OUTPUT, PEFORM, ATTRIBUTE, DESTINATION, RESULT, EVALUATE, APPLY-TO, INPUT, IN-OUT, SUBCONCEPT, POSS, CONDITION, SPLIT and OTHER. We choose the weighted F1 score as the evaluation metric.

Parameter Name	**Value**
Word Emb. size	200
Word Entity Type (or TSS) Emb. size	50
Word Position Emb. szie	100
Convolutional Units	1000
Context Window size	3
Learning Rate	0.01

Table 3.4: Hyperparameters for Relation Classification

3.6 Experiments

3.6.1 Setup

Since the most informative part of text to classify the relation type generally exists between and including target entity pair [37, 78], we only utilize this part of the sentence and disregard the surrounding words for RC.

Previous works have shown that scientific papers specific pre-trained word embeddings can improve training for scientific RC models [60, 27, 31, 43]. Therefore, in this work, we trained the scientific papers specific word embeddings on the ACL Anthology Reference Corpus [5] by the skip-gram NN architecture made available by the Gensim word2vec tool. We initialized [6] the word embedding layer with the pre-trained domain-specific word embedding for RC. We randomly extract 10% training data as validation data and based on the performance on it to select all the hyperparameters. All experiments below use the hyperparameters as shown in Table 5.3.

3.6.2 Result and Discussion

In this paper, we hypothesize that TSS could be used to improve the performance of scientific RC. For testing this hypothesis, we compare the performance of TSS

[6] In experiments on SemEval2018 Task 7 datasets, we didn't tune the word embedding layer, but on RANIS dataset, we tuned it while training.

enhancement with the base model. In other words, we compare the performance before-and-after the automatic TSS tagging, which is mentioned in Section 3.3.

Results for SemEval-2018 Task 7.1.1 are show in Table 4.4. Adding *RESEARCH-PROCESS* proves to be very beneficial compared to the base model alone, as we could improve macro-F1 by more than 5 points. This improvement can be explained by the interdependency between TSS and scientific relations as mentioned in Section 5.1. Thus, even if the number of training samples is small, depending on the corelation, a RC system could correctly classify some relations. While adding the TSS, *SYSTEM or METHOD*, could not enhance the performance on this subtask. This could be because given a specific RC task and its corresponding dataset, some TSS might be redundant when classifying relations. In other words, without the external information from TSS, only the internal information from the dataset itself (e.g., the hint word "*using*" in Example 17) could be enough to identify some relations (e.g., USAGE(X, Y) in Example 17).

(17) $\overset{\textit{entity}}{\underline{\textit{predictor}}}_X$ *pre-selects the phrase candidates*
using $\overset{\textit{entity}}{\underline{\textit{transition rules}}}_Y$

Similar observation can be made for SemEval-2018 Task 7.1.2, as is indicated in Table 4.5. Identification of the TSS, *SYSTEM or METHOD*, could enhance the performance, while adding the *RESEARCH-PROCESS* could decrease the performance. This indicates that, given a specific RC task, different TSS could have different contribution to the overall performance. Therefore, it would be important to select proper TSS for a given RC task.

Figure 5.5 and Figure 3.5 compare some practical results between the TSS enhanced model and **Base** model in SemEval-2018 Task 7.1. Take the second line in Figure 5.5

Model	Precision	Recall	F-score
Base	79.61	64.73	71.40
Base + *SYSTEM or METHOD*	79.99	64.39	71.35
Base + *RESEARCH-PROCESS*	79.97	75.70	77.78
Base + *INPUT-PROCESS* + *OUTPUT-PROCESS*	80.05	62.81	70.39
Base + all	80.65	75.68	**78.09**

Table 3.5: Performance on SemEval-2018 Task 7.1.1

Model	Precision	Recall	F-score
Base	84.18	83.51	83.84
Base + *SYSTEM or METHOD*	84.92	89.04	**86.93**
Base + *RESEARCH-PROCESS*	80.09	82.19	81.12
Base + *INPUT-PROCESS* + *OUTPUT-PROCESS*	83.95	83.91	83.93
Base + all	82.58	88.58	85.48

Table 3.6: Performance on SemEval-2018 Task 7.1.2

as an example, although there is the preposition "*for*", which usually appears in relation USAGE (e.g., "$\underline{\textit{parsing algorithm}}_X$ *for* $\underline{\textit{augmented context-free grammars}}_Y$"), the TSS enhanced model correctly identify the relation as MODEL-FEATURE rather than USAGE, partially because there is no entity marked as *SYSTEM or METHOD*, which is usually associated with USAGE relation.

In Table 3.7 and Table 3.8, we provide our SemEval-2018 Task 7.1 performance in the context of the original task participants. In both subtasks, our model could rank among Top 3, especially in subtask 7.1.2, our system could outperform the second best system. This indicates that, firstly, our selected base model is comparatively strong, secondly, the proposed TSS could boost the performance of the strong base model, so that it could achieve the competitive result to these top ranking models. This again indicates the effectiveness of TSS on scientific RC.

Result on RANIS dataset are shown in Table 3.9. Adding TSS information outperforms the base model. This also proves the effectiveness of TSS on scientific RC. In

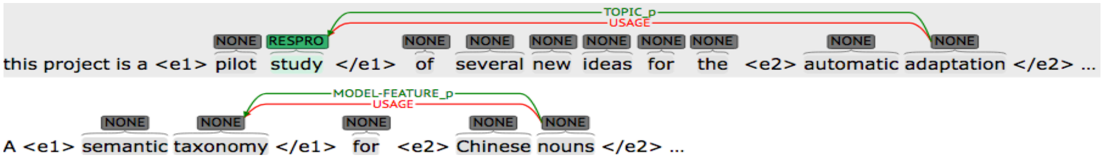

TOPIC_p
USAGE
NONE RESPRO NONE NONE NONE NONE NONE NONE NONE NONE
this project is a <e1> pilot study </e1> of several new ideas for the <e2> automatic adaptation </e2> ...
MODEL-FEATURE_p
USAGE
NONE NONE NONE NONE NONE
A <e1> semantic taxonomy </e1> for <e2> Chinese nouns </e2> ...

Figure 3.4: Comparison between **Base + all** and **Base** in SemEval-2018 Task 7.1.1, where red lines indicate the error from **Base**, while the green lines show the correctly identified relations (which end with "_p") from TSS enhanced model. <e1>, <e2>, </e1> and </e2> are entity boundary marks. RESPRO stands for *RESEARCH-PROCESS*.

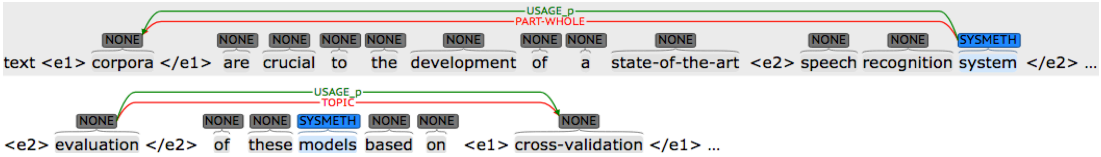

USAGE_p
PART-WHOLE
NONE NONE NONE NONE NONE NONE NONE NONE NONE NONE NONE SYSMETH
text <e1> corpora </e1> are crucial to the development of a state-of-the-art <e2> speech recognition system </e2> ...
USAGE_p
TOPIC
NONE NONE NONE SYSMETH NONE NONE NONE
<e2> evaluation </e2> of these models based on <e1> cross-validation </e1> ...

Figure 3.5: Comparison between **Base + *SYSTEM or METHOD*** and **Base** in SemEval-2018 Task 7.1.2.

addition, as mentioned in Section 7.4.1, RASNIS dataset has been manually annotated with entity types such as PROCESS, PLAN and DATA-ITEM, which have been incorporated in the base model. The enhancement of performance with TSS identification indicates that TSS could be the extension of existing entity type information when classifying semantic relation. Figure 3.6 compares some practical results between **Base + *INPUT-PROCESS* + *OUTPUT-PROCESS*** and **Base** in RANIS dataset. It could be seen that, by adding TSS information, the RC system could correctly distinguish some relations such as INPUT and OUTPUT.

In Comparison with the improvement of performance in SemEval-2018 Task 7 dataset, the increase in RANIS dataset is smaller. This could be because, firstly, the types of target relations in RANIS dataset are more than the ones in SemEval-2018 Task 7 dataset. Secondly, in RANIS dataset, one entity tends to participate in multiple relations in a single sentence. For instance, in the annotation example shown

Rank	Participant	Macro-F1 Score
1	ETH-DS3Lab	81.7
2	UWNLP	78.9
3	SIRIUS-LTG-UiO	76.7
4	ClaiRE	74.9
5	Talla	74.2
	Our model	78.1
	Base model	71.4

Table 3.7: Performance comparison to Top 5 task participants (28 teams) for SemEval-2018 Task 7.1.1

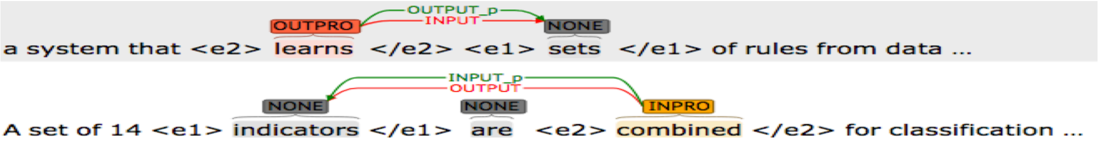
Rank	Participant	Macro-F1 Score
1	ETH-DS3Lab	90.4
2	Talla	84.8
3	SIRIUS-LTG-UiO	83.2
4	MIT-MEDG	80.6
5	GU IRLAB	78.9
	Our model	86.9
	Base model	83.8

Table 3.8: Performance comparison to Top 5 task participants (20 teams) for SemEval-2018 Task 7.1.2

Model	**Precision**	**Recall**	**F-score**
Base	69.34	68.91	67.85
Base + *SYSTEM or METHOD*	70.41	69.70	68.62
Base + *RESEARCH-PROCESS*	69.52	68.83	67.91
Base + *INPUT-PROCESS* + *OUTPUT-PROCESS*	71.12	70.05	**69.34**
Base + all	70.92	69.44	68.71

Table 3.9: Performance on RANIS dataset

in Figure 5.4, the second line, entity "*analyze*" participates in three different relation. Thus, only identifying the entity "*analyze*" as *INPUT-PRCOESS* might not be enough to distinguish them.



a system that <e2> learns </e2> <e1> sets </e1> of rules from data ...

A set of 14 <e1> indicators </e1> are <e2> combined </e2> for classification ...

Figure 3.6: Comparison between **Base + *INPUT-PROCESS* + *OUTPUT-PROCESS*** and **Base** in RANIS dataset, where OUTPRO stands for *OUTPUT-PROCESS*.

3.7 Conclusion

In this work, we address the task of relationship classification in scientific documents by leveraging TSS. We utilize a small number of seed TSS instances to train supersense embeddings and based on the embedding cosine similarity to identify TSS for given words. We extend one of state-of-the-art RC models by the proposed TSS information. Experimental results on three different datasets demonstrated that, firstly, TSS could be used as a feature to improve performance of scientific RC, secondly, the selection of TSS is essential for a given scientific RC task, thirdly, TSS could extend the exiting entity type information.

Chapter 4

Leveraging Unannotated Texts for Scientific Relation Extraction

4.1 Introduction

In recent years, with an increase in the number of scientific papers, it is prohibitively time-consuming for researchers to review and fully-comprehend all papers. To effectively and quickly access a large amount of scientific papers and acquire useful knowledge, a wide variety of computational studies for structuralizing scientific papers has been conducted, such as Argumentative Zoning [67], BioNLP Shared Task [12], and ScienceIE Shared Task [3]. One fundamental study is Relation Extraction (RE). In this paper, we explore the task of RE as an approach for effectively and quickly accessing a large amount of scientific papers and acquiring relevant knowledge.

RE is the task of capturing predefined semantic relations between entities from text. Thus, our task consists of the following: given a sentence that has been annotated with entity[5] mentions, we aim towards extracting relations among entities. Suppose the following sentence[6]:

(18) $\overset{entity}{\underline{\textbf{\textit{RTMs}}}}\ \overset{entity}{\underline{\textit{achieve}}}\ \overset{entity}{\underline{\textit{top}}}\ \overset{entity}{\underline{\textit{performance}}}\ \textit{in}\ \overset{entity}{\underline{\textit{automatic}}}\textit{,}\ \overset{entity}{\underline{\textit{accurate}}}\textit{, and}\ \overset{entity}{\underline{\textit{language independent}}}$
$\overset{entity}{\underline{\textbf{\textit{prediction}}}}\ \textit{of}\ \overset{entity}{\underline{\textit{sentence-level}}}\ \textit{and}\ \overset{entity}{\underline{\textit{word-level}}}$
$\overset{entity}{\underline{\textit{statistical machine translation}}}\ \overset{entity}{\underline{\textit{(SMT)}}}\ \overset{entity}{\underline{\textit{quality}}}\textit{.}$

In Example 26, one of the scientific relations we aim to extract is the relation APPLY_TO(*RTMs*, *prediction*), which means that *RTMs* is the method that is used for the action of *prediction*. For notational convenience, we refer to a sentence where a relation is extracted from as a *target sentence*, and we refer to the related entity pair as a *target entity* pair.

The task of RE for entity pairs can be seen as a classification task. Specifically, given all possible entity pair combinations from a target sentence, the task is to categorize each pair into relation types including predefined relations and non-relation. For example, in Example 26, given the pair (*RTMs*, *prediction*), the output would be APPLY_TO(*RTMs*, *prediction*), and given the entity pair (*RTMs*, *top*), it would be non-relation(*RTMs*, *top*), which means that they do not belong to a predefined relation. With this level of fine-grained analysis, many applications, such as scientific question answering (QA) and scientific paper summarization, can benefit.

Many previous works on RE exist in the general domain [35, 83]. The earlier approaches depend on complex feature engineering such as manually prepared lexical-syntactic patterns [9, 65, 10, etc.]. Recently, Neural Network (NN)-based approaches

[5] In this work, *entity* refers not merely to concepts denoted by noun or noun phrase, it could be actions denoted by verb or verb phrase, and evaluation denoted by adjective or adverb etc.

[6] This example is taken from W13-2242, ACL anthology (`http://aclanthology.info`).

achieve close or even better performance to earlier approaches without complicated manually prepared features [79, 81, 61]. In the context of scientific RE, Ammar et al. [1] enhanced Miwa and Bansal [49]'s end-to-end general relation extraction model by incorporating external knowledge such as gazetteer-like information extracted from Wikipedia. However, no previous work leverages raw scientific documents as a source of background knowledge for RE.

In this work, we hypothesize that unannotated scientific papers can be utilized as a source of background knowledge for scientific RE. We attribute this to the fact that firstly the annotation scheme of scientific relations is based on scientific concepts such as Computer Science (CS) related concepts [66] like “Input” and “Computational_model”, and biochemistry related concepts [59] like “Phosphorylate” and “Myristoylated_by”. This implies that the corpus annotator is required to have external background knowledge about these scientific concepts such as "which entity is a computational_model/corpus/featrue". Secondly, the background information about these concepts are detailed in scientific paper. For instance, CS papers describe the background knowledge [66], which is like *“... proposed Database Semantics as a computational model for natural language semantics ...”*. Therefore, we hypothesize that if a RE system performs similar to the human annotator, the RE system will need to share with the human annotator similar background information about these scientific concepts, which could be extracted from scientific papers. In other words, we hypothesize that the background information about these CS related concepts can be automatically extracted from unannotated CS papers, and the extracted background information can facilitate RE in CS related dataset such as Tateishi et al. [66]'s RANIS corpus, which will be detailed in Section 7.4.1. Suppose the following sentence taken from the RANIS corpus:

(19) *RTMs*$_A$ *achieve top performance in automatic, accurate, and language independent prediction*$_B$ *of sentence-level and word-level statistical machine translation (SMT) quality.*

In Example 19, without any support of background information regarding the concept *RTMs*, such as "what is a *RTM*" (e.g., "computational model", "research team members", or "dataset"), its relation to the entity *prediction* can seem ambiguous. Specifically, if *RTMs* refers to a "computational model", a RE system might extract APPLY_TO(*RTMs*, *prediction*) relation, because the target sentence in Example 19 means that *RTMs* is the method or computational model that is **applied to** the action *prediction*. However, if *RTMs* refers to "research team members", the relation would be extracted as PERFORM(*RTMs*, *prediction*). Finally, if *RTMs* refers to a "corpus", the relation tends to be INPUT(*RTMs*, *prediction*).

Although the target sentence in Example 19 lacks enough background information about the target entity for disambiguating relation extraction, we could find the following sentences about the target entity *RTMs* from other sections of the same paper (Examples 20 and 21):

(20) *Referential translation* ***machines*** *(RTMs) provide a* ***computational model*** *for quality and semantic similarity judgments using retrieval of relevant training data ...*

(21) *... we* ***use*** *RTMs to automatically assess the correctness of student answers to obtain better result than the sate-of-the-art.*

Example 20 describes that the concept *RTMs* refers to a machine that could act as a *computational model*, and Example 21 mentions that *RTMs* could be **used** for some process. As discussed before, this information could be leveraged as background knowledge for disambiguating the relation as APPLY_TO(*RTMs*, *prediction*) rather than PERFORM(*RTMs*, *prediction*) or INPUT(*RTMs*, *prediction*), because *RTMs* is semantically closer to *computational model* rather than *research team members* or *corpus* in Examples 20 and 21.

For utilizing background knowledge, one possibility is to manually annotate useful background information about CS related concepts, such as "*RTMs* are a Computational Model" and "Using *WordNet* as a knowledge base", in scientific papers and apply the annotated scientific papers to RE. However, manual annotation is time consuming [33] and expensive [2].

To address this issue, in this work, we investigate the effectiveness of leveraging unannotated text for RE. Specifically, we propose two methods, term sentence (TS) and semantically related word (SRW), for automatically extracting background knowledge from unannotated scientific papers and utilizing the extracted background information for extending a state-of-the-art neural RE model. Our evaluation empirically demonstrates that incorporating the extracted TS and SRW from unannotated scientific papers improves the performance of RE.

4.2 Related Work

Conventional approaches for RE rely on human-designed, complex lexical-syntactic patterns [9], statistical co-occurrences [65] and structuralized knowledge bases such as WordNet [24, 10]. In recent years, exploring Neural Network (NN)-based models has

Figure 4.1: Annotation example shown in brat rapid annotation tool. To more clearly illustrate the direction of relation, we add directional tag “L-” (means left hand side is the argument B) and “R-” (means right hand side is the argument B) before each relation tag.

Table 4.1: Frequently Appeared Relation Tags

Type	**Definition**	**Example**
ATTRIBUTE(A, B)	B is an attribute or a characteristic of A	*accuracy*$_{\text{A}}$ *of the tagger*$_{\text{B}}$
OUTPUT(A, B)	B is the output of a system or a process A; B is generated by A	*an image*$_{\text{B}}$ *displayed*$_{\text{A}}$ *on a palm*
APPLY_TO(A, B)	a method A is applied to achieve the purpose B	*CRF*$_{\text{A}}$*-based tagger*$_{\text{B}}$
INPUT(A, B)	B is the input of a system or a process A; B is consumed by A	*corpus*$_{\text{A}}$ *for training*$_{\text{B}}$
EVALUATE(A, B)	A is evaluated as B	*experiment shows an increase*$_{\text{B}}$ *in F-score*$_{\text{A}}$ *compared to the baseline*
SUBCONCETP(A, B)	A is-a, or is a part-of B	*a corpus*$_{\text{B}}$ *such as PTB*$_{\text{A}}$
CONDITION(A, B)	The condition A holds in situation B, e.g, time, location, experimental condition	*a survey*$_{\text{B}}$ *conducted in India*$_{\text{A}}$
EQUIVALENCE(A, B)	terms A and B refer to the same entity: definition, abbreviation, or coreference	*DoS*$_{\text{B}}$ *(denial-of-service*$_{\text{A}}$*) attack*
PERFORM(A, B)	A is the agent of an intentional action B	*a frustrated player*$_{\text{A}}$ *of a game*$_{\text{B}}$
IN_OUT(A, B)	B is simultaneously INPUT and OUTPUT and is changed by a system or a process A	*a modified*$_{\text{A}}$ *annotation schema*$_{\text{B}}$

been the dominant approach in the field. Zeng et al. [79] proposed a deep Convolutional Neural Network (CNN)-based framework, which depends on sentence-level features collected from an entire target sentence and lexical-level features from lexical resources such as WordNet [18]. Santos et al. [61] proposed a ranking CNN model, which is trained by a pairwise ranking loss function. To improve the ability of sequential modeling, Zhang et al. [81] proposed a recurrent neural network (RNN)-based model for RE. Other variants of RNN-based models have been proposed, such as Miwa et al. [49], who proposed a bidirectional tree-structured LSTM model. Additionally, similar NN-based approaches are used in scientific relation extraction. For instance, Gu et al. [22] utilized a CNN-based model for identifying *chemical-disease* relations from the abstracts of MEDLINE papers. Hahn-Powell et al. [25] proposed an LSTM-based RNN model for identifying *causal precedence* relationship between two event mentions in biomedical papers. Ammar et al. [1] enhanced Miwa and Bansal [49]'s relation extraction model via extensions such as gazetteer-like information extracted from Wikipedia. However, none of these approaches leverage unannotated scientific papers for RE.

4.3 Data

We evaluate the performance of RE using the RANIS corpus [66], a collection of computer science paper abstracts. The type of entity (referred to as Entity Type (ET) hereafter) and domain specific relation in the RANIS corpus has already been annotated with the annotation scheme proposed by [66], as shown Fig. 5.4. The corpus consists of ETs such as QUALITY, PROCESS and DATA-ITEM and domain specific scientific relations, such as INPUT, OUTPUT and APPLY_TO. Table 4.1 summarizes frequently appearing domain specific relations and provides both definitions and examples.

In total, the RANIS corpus contains 250 abstracts collected from ACL Anthology (230 abstracts in the development set and 20 abstracts in the test set) and 150 abstracts collected from ACM Digital Library. For training and testing our proposed model, we only use the 250 abstracts from ACL Anthology. From the ACL Anthology abstracts, we extract 11,520 relations from the development set of ACL Anthology and 1,142 relations from the test set of ACL Anthology. The distribution of relation types for both sets is shown in Figure 4.2. For each ACL anthology abstract in the RANIS corpus, we collect its corresponding unannotated paper body from ACL Anthology Reference Corpus [5] as the source of background information for RE.

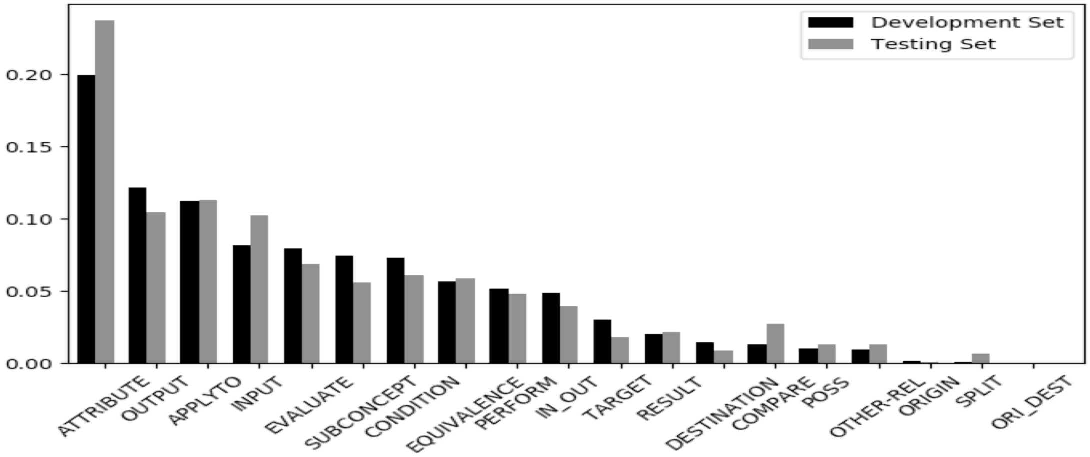


Figure 4.2: Distribution of relation types.

4.4 Proposed Model

In this paper, we hypothesize that unannotated scientific papers can be utilized as a source of background information for RE. Therefore, we create a problem setting where we consider an annotated sentence in a paper abstract as a target sentence, and the corresponding unannotated paper body of the abstract (henceforth, *paper body*) as the source of background information. We hypothesize that the background information extracted from the paper body could facilitate relation extraction in paper abstracts. We believe that this setting can be easily adapted to a more general task setting, e.g. analyzing semantic relation in a whole document (not just in an abstract) via considering a collection of unannotated scientific papers as a source of background information.

Based on this hypothesis, we propose a new relation classification model that categorizes relations not only based on the target sentence, but also on the background information acquired from unannotated scientific papers, as illustrated in Section 5.1. To create such a model, we need to address the following questions:

1. From the perspective of knowledge acquisition, how do we extract the background

information from unannotated scientific papers?

2. From the perspective of NN, how do we encode the extracted information into a vector representation for relation classification?

4.4.1 Retrieving Background Information from Unannotated Scientific Papers

For acquiring background knowledge from unannotated scientific papers, we propose two methods.

Method 1: extract all of the sentences containing the target entity of interest in the unannotated paper body as a representation of background information (henceforth, referred to as *Term Sentence(TS)*)[7]. Formally, $TS_A = w_{A1}, ...ent_A, ..., w_{Ai}, ...w_{An}$ and $TS_B = w_{B1}, ...ent_B, ..., w_{Bi}, ...w_{Bn}$, where ent_A and ent_B are target entities, w_{Ai} (w_{Bi}) is the word of the sentence in which the target entity ent_A (ent_B) exists. For example, given a target entity *RTM*, we could find the following TSs in its corresponding paper body:

(22) ***RTM** is a computational model for identifying the acts of translation for translating between any given two data sets with respect to a reference corpus selected in the same domain.*

(23) ***RTM** can be used for predicting the quality of translation outputs.*

Given multiple TSs for a target entity, this method simply concatenates all of the individual TSs (e.g., Examples 22 and 23) into an overall representation of TS and feeds it to the proposed model.

[7] In this work, we only choose the noun phrase target entity to extract TS.

The intuition behind the method is that a TS could contain domain-specific background information about target entity for relationship analysis. For instance, Example 23 clearly mentions that "***RTM*** *can be used for predicting the quality ...*" and this is effective evidence for the existence of the scientific relationship APPLY_TO($\underline{RTM}_A$, $\underline{quality\ estimation}_B$) relationship in the target sentence (Example 24).

(24) *We introduce referential translation machines ($\underline{RTM}_A$) for $\underline{quality\ estimation}_B$ of translation outputs of sentence-level and word-level statistical machine translation (SMT) quality.*

Method 2: extract Semantically Related Word as a representation of background information for RE. In this work, we define SRW as the set of content words (e.g., nouns, verbs and adjectives) from a paper body that are semantically close to a given target entity.

The process of extracting SRW in this work is similar to the approach proposed by [45]. Specifically, based on word embeddings, we calculate cosine similarity between a given target entity (from a paper abstract) and each content word from its corresponding paper body, and then use a predefined criteria to select the member for its SRW. We manually set the SRW criteria (SRW_c) as 0.35, and only collect the word whose cosine similarity with the target entity is larger than the SRW_c as the member of SRW. The effect of SRW_c on RE performance will be discussed in Section 4.5.2. Formally, $SRW_A = \{w_{A1}, ..., w_{Ai}, ...w_{An} | cos(e_{ent_A}, e_{w_{Ai}}) > \text{SRW\_c}\}$ and $SRW_B = \{w_{B1}, ..., w_{Bi}, ...w_{Bn} | cos(e_{ent_B}, e_{w_{Bi}}) > \text{SRW\_c}\}$, where SRW_A (SRW_B) is the SRW for entity A (B), w_{Ai} (w_{Bi}) is the content words from the paper body, $e_{w_{Ai}}$ ($e_{w_{Bi}}$) is its word embedding and e_{ent_A} (e_{ent_B}) is the word embedding of the target entity A (B).

The following example is a practical case of SRW extraction applied in this work.

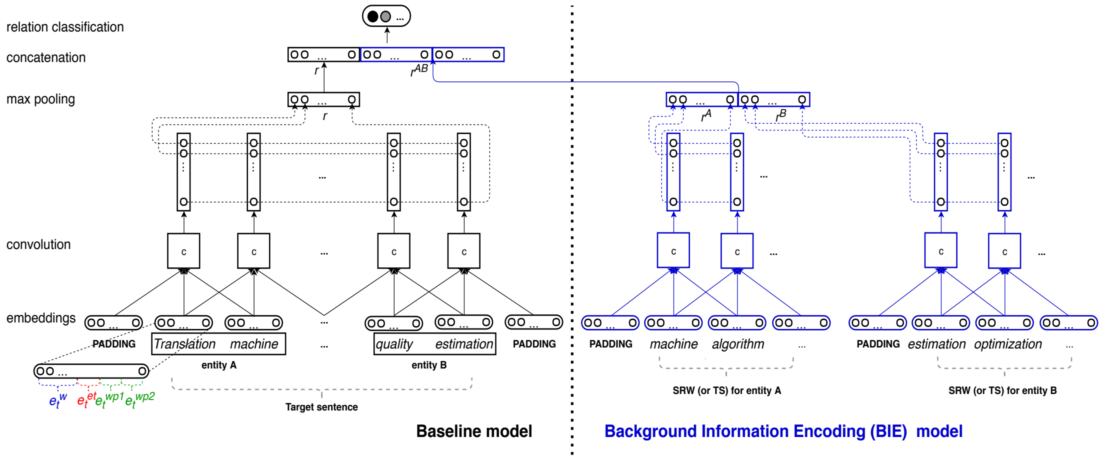


Figure 4.3: The architecture of the proposed model enhanced by LC (or TS) encoding.

Given a target sentence (e.g. Example 25) with a marked target entity pair[8], the method automatically extracts SRW_A and SRW_B, from its corresponding paper body for target entity pair, "***extraction***" and "***collections***"[9], respectively.

(25) *We are interested in the problem of* *<u>word extraction</u>*$_A$ *from* *<u>Chinese text collections</u>*$_B$.

SRW_A: *extraction, extracting, identification, retrieval, filtering*

SRW_B: *collections, corpora, sets, texts, corpus, data*

The intuition behind applying SRW for RE is inspired by its usage in word sense disambiguation [44]. Specifically, given an entity, its entity type might differ in distinct texts. For instance, the specific entity type for *"collections"* in Text1[10] is different with the one in Text2[11]. In Text1, *"collections"* belongs to the type of *corpus*, but in

[8] This example is taken from J04-1004, ACL anthology (`http://aclanthology.info`).

[9] In this work, we only select the noun (phrase), verb (phrase) and adjective target entity and simply use its head word to extract SRW.

[10] This example is taken from D09-1074, ACL anthology (`http://aclanthology.info`).

[11] This example is taken from A94-1009, ACL anthology (`http://aclanthology.info`).

Text2, it refers to *parameters*. This difference could be illustrated by extracting SRW of *"collections"* from each Text, which is denoted in parenthesis. Since entity type information closely interacts with relation classification [72, 49], we hypothesize that SRW could illustrate the entity type information about target entity, thereby facilitating RE.

Text1: *Typically, a parallel training corpus is comprised of collections*$_A$ *of varying quality and relevance to the translation problem of interest.*
(*SRW*$_A$*: collections, corpus*)

Text2: *The model is defined by two collections*$_A$ *of parameters: the transition probabilities, which express the probability that a tag follows the preceding one (or two for a second order model); and the lexical probabilities,*
(*SRW*$_A$*: collections, parameters*)

For instance, suppose we intend to classify the relation between *"collections*$_A$*"* and *"model*$_B$*"* in the target sentence, *"We apply these collections*$_A$ *to train the model*$_B$*"*. In the context of Text1, the relation would be INPUT, because the SRW in Text1 indicates that *"collections"* is semantically similar to the entity *corpus*, and *corpus* is usually used as the input data for training a NLP model. In contrast, in the context of Text2, they have a low tendency to hold INPUT relation, when in fact, have high tendency to hold ATTRIBUTE relation, because in Text2, *"collections"* belongs to the type of *parameters*, and *parameters* is not the input data, but the attribute of the *"model"*. Similarly in Example 25, SRW$_B$ contains *"corpus"*, therefore the target entity, *"collections"*, has high tendency to participate in INPUT relation, which is the gold standard relation in RANIS corpus [66].

4.4.2 Architecture

The proposed NN model, in general, contains two main parts: Baseline model and Background Information Encoding model (BIE model, for short) as shown in Figure 4.3. The former converts the target sentence into a vector representation, and the latter is responsible for converting the acquired TS pair and SRW pair into a vector representation.

The Baseline model is the CNN-based baseline model that has been described in Chapter 3. The BIE model, as shown in Figure 4.3, is used for encoding SRW (or TS) of entity A and SRW (or TS) of entity B, thus having a parallel structure. The parallel CNN-model for each SRW (or TS) has independent convolutional weight matrix W_1 and W_2 but shares word embedding projection matrix W^w_{emb}. As shown in Figure 4.3, BIE model consists of 3 layers: the first layer is the word embedding layer that maps each word from SRW or from TS into word vector via Equation 4.1, where $X^{w_A}_t$ ($X^{w_B}_t$) is the one-hot of the word from SRW_A (SRW_B) or from TS_A (TS_B). The second layer is the convolutional layer, which generate the convolutional filter level vector z^A_t and z^B_t via Equation 4.2-4.4, where k is the convolutional window size. The third layer is max pooling layer, which chooses a maximum value from each SRW (or TS) via Equation 4.5, where i indexes feature dimensions, m is the number of feature dimensions. The final output of BIE model is calculated via Equation 4.6.

$$e^{w_{A(orB)}}_t = W^w_{emb} x^{w_{A(orB)}}_t \tag{4.1}$$

$$z^{A(orB)}_t = concat(e^{w_{A(orB)}}_{t-(k-1)/2}, ..., e^{w_{A(orB)}}_{t+(k-1)/2}) \tag{4.2}$$

$$h^A_t = tanh(W_1 z^A_t + b_1) \tag{4.3}$$

$$h_t^B = tanh(W_2 z_t^B + b_2) \tag{4.4}$$

$$r_i^{A(orB)} = \max_t\{(h_t^{A(orB)})_i\}, \forall i = 1, ..., m \tag{4.5}$$

$$r^{AB} = concat(r^A, r^B) \tag{4.6}$$

Finally, the final vector representation of a SRW pair (or TS pair), r^{AB}, and the final output vector of the Baseline model, r, are concatenated and fed to a semantic relation classifier.

We use the back-propagation algorithm for training the model and choose the logistic loss function in Equation 5.11 as the objective function.

4.5 Experiments

4.5.1 Setup

From the RANIS corpus, we extract 67,929 possible intra-sentence entity pairs from the ACL development set and 6,674 intra-sentence entity pairs from the ACL testing set. From the development set, we randomly select 90% of samples as training data and the rest as validation data for tuning hyper parameters such as the number of hidden layer dimensions, the number of epochs, learning rate, etc. In Table 4.2, we show the distribution of the RELATED entity pairs, which means that the entity pair belongs to a predefined relation such as INPUT. In Table 5.3, we show the selected hyper parameter values.

Previous works have shown that pre-trained word embeddings can improve training for relation extraction models [79, 81, 61]. Therefore, in this work, we trained scientific paper specific word embeddings on the ACL Anthology Reference Corpus [5] (in total:

Table 4.2: Distribution of RELATED entity pairs.

Data type	**Percentage (RELATED/all)**
training data	17.0% (10,391/61,137)
validation data	16.6% (1,129/6,792)
testing data	17.1% (1,142/6,674)

Table 4.3: Hyperparameters for Relation Classification

Parameter Name	**Value**
Word Emb. size	200
Word Entity Type Emb. size	50
Word Position Emb. szie	100
Convolutional Units (Baseline model)	1000
Context Window size (Baseline model)	3
Convolutional Units (BIE model)	100
Context Window size (BIE model)	3
The Number of Epoch	25
Learning Rate	0.003

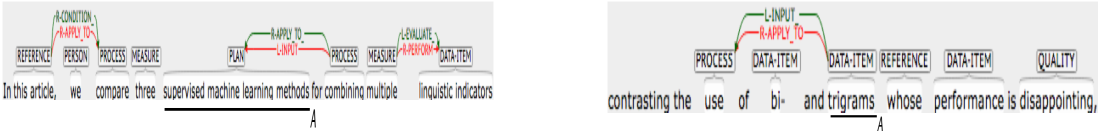
about 3 million sentences) by the skip-gram NN architecture made available by the Gensim word2vec tool[12]. We initialized the word embedding layer with the pre-trained domain-specific word embedding for RE.

We implemented the baseline model, proposed NN model, and the back-propagation algorithm with Theano [4]. To minimize the influence of random initialization of model parameters on RE, we ran each evaluation 5 times and took their mean value for comparison.

4.5.2 Result

In this work, we hypothesize that unannotated scientific papers could be used as a source of background information for scientific RE. We propose two methods for extracting background information: i) Term Sentence (TS), and ii) Semantically Related Word (SRW). For testing this hypothesis, we compare the performance of each method with the baseline approach, the CNN baseline model introduced in the previous section.

[12] https://radimrehurek.com/gensim


R-CONDITION
R-APPLY_TO
REFERENCE
PERSON
PROCESS
MEASURE
PLAN
R-APPLY_TO
L-INPUT
PROCESS
MEASURE
L-EVALUATE
R-PERFORM
DATA-ITEM
In this article, we compare three supervised machine learning methods for combining multiple linguistic indicators
A
L-INPUT
R-APPLY_TO
PROCESS
DATA-ITEM
DATA-ITEM
REFERENCE
DATA-ITEM
QUALITY
contrasting the use of bi- and trigrams whose performance is disappointing,
A

(a) SRW_A: *methods, techniques, algorithms systems, models, ...*

(b) SRW_A: *bigram, trigram, unigram, tokens, words, ...*

Figure 4.4: Comparison between **Baseline + SRW** and **Baseline**, where red lines indicate the error from **Baseline**, while the green lines show the correctly identified relations from **Baseline + SRW**.

Tables 4.4 presents the overall performance of baseline model and each extension. It can be seen that all extension from our proposed method gets better performance than the baseline approach. Table 4.5 detects the influence of our proposed method on each individual relationship. It can be seen that the proposed methods perform better than the baseline approach over a majority of the relationships. The better performance indicates the following: unannotated scientific papers are useful resource of background information for RE, and for the two proposed methods, TS and SRW, especially the combination of TS and SRW, which achieved the highest scores, is effective method for extracting background information from unannotated scientific papers for scientific RE. Additionally, all of the proposed methods are unsupervised, and the results also confirm the feasibility of unsupervised method on tapping the potential of unannotated scientific papers for scientific RE.

Figure 4.4 compares some practical results between **Baseline + SRW** and **Baseline**. Take (b) as an example, although there is the target entity "use", which usually appears in relation APPLY_TO, the proposed system correctly identify the relation as INPUT, because SRW of "trigrams" contains such informative words like "tokens" and "words" that are frequently used as input data for some process.

In addition to comparing the performance over the relations that include non-relation,

Table 4.4: Performance of RE (mean ± standard deviation)

Model	**Precision**	**Recall**	**F-score**
Baseline	62.79±1.22	50.58±0.46	54.5±0.45
Baseline + TS	62.88±0.42	50.75±0.48	54.96±0.34
Baseline + SRW	63.02±0.7	51.67±0.52	55.56±0.46
Baseline + TS + SRW	**65.14**±0.63	**52.08**±0.58	**56.47**±0.44

Table 4.5: Performance (F-score) over selected relationship

	Baseline	**Proposed Method**		
Relationship	Baseline	Baseline+TS	Baseline+SRW	Baseline+TS+SRW
ATTRIBUTE	75.09±0.5	73.73±0.74	**75.35**±1.05	74.65±0.7
APPLY_TO	53.08±0.56	53.81±1.95	**55.75**±1.56	55.53±1.2
OUTPUT	49.58±2.49	52.06±1.57	51.03±1.65	**52.3**±1.48
INPUT	38.83±2.54	40.56±1.54	41.34±1.17	**43.27**±2.44
EVALUATE	93.36±1.15	92.26±1.18	92.87±0.92	**93.78**±0.54
CONDITION	38.47±3.92	37.54±3.97	36.41±3.71	**39.27**±2.64
EQUIVALENCE	56.0±2.28	56.6±1.85	56.4±1.74	**57.0**±1.1
SUBCONCEPT	22.47±5.64	22.95±2.74	24.81±3.39	**32.4**±4.64
PERFORM	89.4±0.8	89.8±0.98	88.6±0.8	**90.2**±0.75
IN_OUT	45.96±1.6	**47.49**±2.0	46.82±4.15	46.93±1.32
RESULT	5.34±4.88	6.81±4.1	9.38±3.26	**12.14**±4.74
TARGET	20.54±2.18	19.92±2.15	**20.71**±3.28	20.21±1.49

we also detect the influence of our proposed method when omitting the non-relation. Table 4.6 and Table 4.7 present the result on the setting that excludes non-relation. As shown in Table 4.6 and Table 4.7, the proposed methods outperform the baseline approach. Again, this comparison indicates the effectiveness of the proposed model for RE in scientific documents.

As mentioned in Section 8.3, we utilize a cosine similarity based criteria, SRW_c, to extract SRW from unannotated scientific papers. In Table 4.8, we compare the impact

Table 4.6: Performance of RE on the setting that **excludes** non-relation

Model	**Precision**	**Recall**	**F-score**
Baseline	68.88±1.3	66.88±0.77	66.34±1.23
Baseline + TS	68.75±0.74	67.18±0.79	66.78±0.61
Baseline + SRW	69.1±0.89	68.97±0.53	68.35±0.59
Baseline + TS + SRW	**70.23**±0.44	**70.19**±0.48	**69.6**±0.43

Table 4.7: Performance (F-score) over selected relationship on the setting that **excludes** non-relation

	Baseline	**Proposed Method**		
Relationship	Baseline	Baseline+TS	Baseline+SRW	Baseline+TS+SRW
ATTRIBUTE	80.36±1.33	80.55±0.66	81.55±0.7	**81.88**±0.82
APPLY_TO	74.79±1.42	73.95±1.85	76.92±1.58	**77.91**±1.72
OUTPUT	60.83±3.06	62.35±1.31	63.46±0.94	**65.7**±1.43
INPUT	52.39±1.26	54.66±2.56	56.5±2.12	**58.7**±3.01
EVALUATE	97.67±0.59	97.35±0.56	**98.33**±0.48	96.86±0.84
CONDITION	45.58±4.26	46.36±3.8	45.1±2.22	**48.34**±1.51
EQUIVALENCE	77.8±5.53	82.2±0.75	**85.0**±1.1	82.4±1.85
SUBCONCEPT	44.38±3.51	43.56±2.88	49.66±4.06	**51.48**±3.0
PERFORM	**92.4**±1.96	90.4±2.06	91.2±0.4	91.4±0.8
IN_OUT	47.8±1.51	**49.69**±2.83	48.89±1.69	47.78±1.55
RESULT	42.32±7.97	40.09±8.7	47.26±4.94	**58.88**±3.9
TARGET	23.31±3.98	25.06±3.37	25.67±2.78	**27.82**±3.05

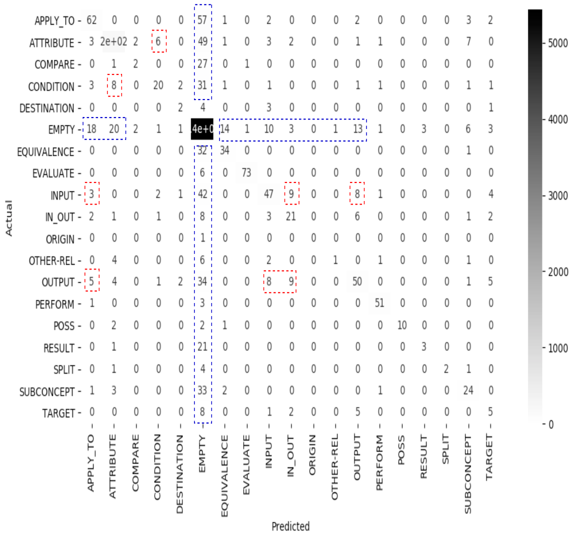
Table 4.8: Impact of using different SRW_c on RE

SRW_c	**Precision**	**Recall**	**F-score**
0.15	65.0±1.53	50.26±0.51	54.44±0.65
0.25	65.84±1.82	49.84±0.61	54.67±0.76
0.35	63.02±0.7	**51.67**±0.52	**55.56**±0.46
0.45	65.56±0.8	50.81±0.5	55.37±0.54
0.55	**66.3**±1.18	50.65±0.74	55.22±0.47

of using different SRW_c on the performance of scientific RE. It can be seen that, the best performance on RE is obtained with a moderate SRW_c like 0.35 and 0.45. This is understandable as the high CRW_c might limit the extraction of informative SRW and the low CRW_c might allow the extraction of noisy and irrelevant SRW from scientific papers, this could negatively affect the performance of RE.

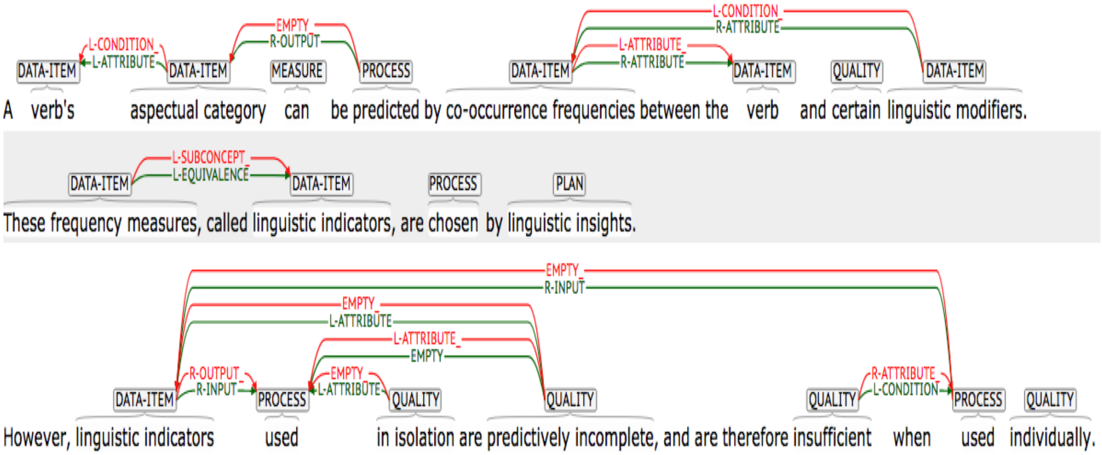
4.5.3 Error Analysis and Discussion

Towards understanding the disadvantage of our proposed method and improve the performance for future work, we randomly select 5 abstracts from the testing data and manually analyze the types of errors from the result of TS and SRW extension (**Baseline**



	APPLY_TO	ATTRIBUTE	COMPARE	CONDITION	DESTINATION	EMPTY	EQUIVALENCE	EVALUATE	INPUT	IN_OUT	ORIGIN	OTHER-REL	OUTPUT	PERFORM	POSS	RESULT	SPLIT	SUBCONCEPT	TARGET
APPLY_TO	62	0	0	0	0	57	1	0	2	0	0	0	2	0	0	0	0	3	2
ATTRIBUTE	3	2e+02	2	6	0	49	1	0	3	2	0	0	1	1	0	0	0	7	0
COMPARE	0	1	2	0	0	27	0	1	0	0	0	0	0	0	0	0	0	0	0
CONDITION	3	8	0	20	2	31	1	0	1	0	0	0	1	1	0	0	0	1	1
DESTINATION	0	0	0	0	2	4	0	0	3	0	0	0	0	0	0	0	0	0	1
EMPTY	18	20	2	1	1	4e+0	14	1	10	3	0	1	13	1	0	3	0	6	3
EQUIVALENCE	0	0	0	0	0	32	34	0	0	0	0	0	0	0	0	0	0	1	0
EVALUATE	0	0	0	0	0	6	0	73	0	0	0	0	0	0	0	0	0	0	0
INPUT	3	0	0	2	1	42	0	0	47	9	0	0	8	1	0	0	0	0	4
IN_OUT	2	1	0	1	0	8	0	0	3	21	0	0	6	0	0	0	0	1	2
ORIGIN	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
OTHER-REL	0	4	0	0	0	6	0	0	2	0	0	1	0	1	0	0	0	1	0
OUTPUT	5	4	0	1	2	34	0	0	8	9	0	0	50	0	0	0	0	1	5
PERFORM	1	0	0	0	0	3	0	0	0	0	0	0	0	51	0	0	0	0	0
POSS	0	2	0	0	0	2	1	0	0	0	0	0	0	0	10	0	0	0	0
RESULT	0	1	0	0	0	21	0	0	0	0	0	0	0	0	0	3	0	0	0
SPLIT	0	1	0	0	0	4	0	0	0	0	0	0	0	0	0	0	2	1	0
SUBCONCEPT	1	3	0	0	0	33	2	0	0	0	0	0	0	1	0	0	0	24	0
TARGET	0	0	0	0	0	8	0	0	1	2	0	0	5	0	0	0	0	0	5

Figure 4.5: Confusion Matrix from **Baseline + TS + SRW**.



A verb's aspectual category can be predicted by co-occurrence frequencies between the verb and certain linguistic modifiers.

These frequency measures, called linguistic indicators, are chosen by linguistic insights.

However, linguistic indicators used in isolation are predictively incomplete, and are therefore insufficient when used individually.

Figure 4.6: Relationship identification error from **Baseline + TS + SRW**, where red lines indicate the error while the green line shows the gold standard relation.

+ TS + SRW), which is visualized like Figure 4.6. Based on the difference between the predicted relation and actual relation, we categorize the error into two types. The first type of error occurs between a relationship with high frequency and the one with low frequency, specifically, the model tends to confuse between EMTPY (means non-relation) and predefined relations such as INPUT and ATTRIBUTE, as shown in the third sentence in Figure 4.6. This observation is also supported by the confusion matrix in Figure 4.5, where this kind of error is marked by a blue rectangle. The second type of error is the error between definitionally similar relationships, which are frequently observed between INPUT and OUTPUT, INPUT and IN_OUT, APPLY_TO and INPUT, ATTRIBUTE and CONDITION etc. as shown in the first sentence of Figure 4.6. This observation is also supported by the confusion matrix in Figure 4.5, where this kind of error is marked by a red rectangle.

There are several optional solutions for addressing these errors. In order to deal with the non-relation bias, we assume that it would be effective to utilize syntactic information between target entities, because syntactically related entities might tend to be in some relation rather than in non-relation. Therefore by incorporating the syntactic path, the system might decrease the non-relation bias. For overcoming the definitionally similar relationships, we assume that it would be effective to extract the information of selectional preference to distinguish these definitionally similar relationships. For instance, for distinguishing between INPUT and APPLY_TO, if one target entity involved in the relation is frequently observed as the OBJECT of the predicate "*apply*" and rarely observed as the OBJECT of "*generate*", the relation might have higher tendency to be in an APPLY_TO than INPUT. This is because the entity, such as "*method*", "*model*" and "*algorithm*", has such selectional preference and usually participates in APPLY_TO relation.

4.6 Conclusion

In this work, we address the task of relationship extraction in scientific documents by leveraging background information extracted from unannotated scientific papers. We design a novel neural network model that not only collects feature from target sentence, but also extracts background information from unannotated scientific papers. We proposed two unsupervised methods: Term Sentence (TS) and Semantically Related Word (SRW). Experimental results on the RANIS corpus demonstrated that unannotated scientific papers could be used as a source of background knowledge for scientific relationship extraction. The proposed unsupervised methods are also proven to be effective for acquiring background information from unannotated scientific papers for relation extraction. An error analysis showed that the proposed model had difficulty for identifying some relationships such as definitionally similar relationships. We assume that this will be improved by incorporating other background information, such as syntactic information and selectional preference information.

Chapter 5

Scientific Knowledge Acquisition via the Interaction between Relation Extraction and Knowledge Graph Completion

5.1 Introduction

The task of RE for entity pairs can be seen as a classification task. Specifically, given all possible entity pair combinations from a target sentence, the task is to categorize each pair into relation types including predefined relations and. In Example 26, given the pair (*RTMs*, *prediction*), the output would be APPLY_TO(*RTMs*, *prediction*), and given the entity pair (*RTMs*, *top*), it would be non-relation(*RTMs*, *top*), which means that they do not belong to a predefined relation. With this level of fine-grained analysis,

head entity	relation	tail entity
SVM	APPLY_TO	*recognition*
SVM	be_INPUT	*microblog*
SVM	?	*classification*
SVM	?	*corpus*

Table 5.1: Instances for Scientific KGC.

many applications, such as scientific question answering (QA) and scientific paper summarization, can benefit.

(26) ***RTMs***$_X$ *achieve* ***top***$_Z$ *performance in automatic, accurate, and language independent* ***prediction***$_Y$ *of sentence-level and word-level statistical machine translation (SMT) quality.*[1]

After extracting useful knowledge, we could use the stored knowledge base to complete the missing knowledge, which is the task of Knowledge Graph Completion. Knowledge Bases (KBs) such as Freebase [7] and DBpedia [38] are extremely crucial for many natural language processing tasks [62]. They provide large collections of relations between entities, typically stored as (h, r, t) triples, where h = head entity, r = relation and t = tail entity, e.g., (*Tokyo*, *capitalOf*, *Japan*). However, the sparsity of KBs impedes their usefulness in real world applications.

KB completion or Knowledge Graph Completion (KGC) automatically infers missing facts by examining the latent regularities in existing ones [68]. For example, suppose the triples (*SVM*, APPLY_TO, *recognition*) and (*SVM*, be_INPUT[2], *microblog*) are stored in a KB, as shown in Table 5.1, based on the fact, a KBC model would infer the new plausible triple (*SVM*, APPLY_TO, *classification*) rather than (*SVM*,

[1] This example is taken from W13-2242, ACL anthology (`http://aclanthology.info`).

[2] where (h, be_INPUT, t) equals (t, INPUT, h).

APPLY_TO, *corpus*), because entity *classification* and entity *recognition* share some latent semantic features.

The latent semantic features are represented by KB embedding, which embeds triple of KB into a continuous vector space, so as to decompose the observed triples into a product of vectors. For a given fact triple (h, r, t) in which head entity h is linked to tail entity t through relation r, the score of plausibility can then be recovered as a multi-linear product between the embedding vectors of h, r and t.

Most successful RE approaches [79, 77, 61, 81, 49] extract salient relational triples mainly based on local lexical-syntactic patterns. Given the Example 27, a RE model could identify the relation triple (*genetic algorithms*, APPLY-TO, *optimization models*) based on the local pattern *"... using ..."*. However, given the Example 28, it might be insufficient to solely consider the local lexical-syntactic pattern. This is because, although target sentences have similar local pattern (e.g., Example 27 and Example 28), the relation triple could vary with their global semantic features of target entities. In the Example 28, the actual triple to be identified is (*corpora*, INPUT, *statistical model*) rather than (*corpora*, APPLY-TO, *statistical model*), because the target entities expressing the semantic meaning of data, such as *corpora*, is not an algorithm but an input data for a natural language processing model.

(27) *In this study,* $\underline{\textit{optimization models}}_X$ *using* $\underline{\textit{genetic algorithms}}_Y$ *(GAs) are proposed to study ...* [3]

(28) *Corpus-based approach trains a probabilistic or* $\underline{\textit{statistical model}}_X$ *using sense-tagged or raw* $\underline{\textit{corpora}}_Y$ *...* [4]

[3] This example is taken from J03-1001, ACL anthology (`http://aclanthology.info`).

[4] This example is taken from R13-2017, ACL anthology (`http://aclanthology.info`).

Data type	**Entity**	**Relation**	**Train**	**Test**
FB15k	14,951	1,345	483,142	59,071
WN18	40,943	18	141,442	5,000
RANIS	5,577	34	11,520	1,142

Table 5.2: Comparison of statistics of KBs.

The global semantic feature of entities, as mentioned, could be learned by a KGC model. Therefore, we assume that by leveraging the learned KB embedding from KGC, we could extend and enhance a RE model so that it would not only utilize the local lexical-syntactic pattern, but also the global semantic information of entities. In addition, we also assume that the KGC model could in turn be facilitated by a RE model.

Various large-scale KBs such as Freebase [7] and DBpedia [38], are available. Their huge volume allows a KGC model to encode every element (entities and relations) of a KB into a low-dimensional embedding vector space. However, comparing to the size of KB in general domain, the size of scientific KB, such as the KB in computational linguistic domain is extremely small. Table 5.2 compares the statistics of some KBs, where both FB15k and WN18 are first introduced by [8] and have been commonly used in KGC researches. The RANIS corpus is created by Tateishi et al. [66], a scientific semantic relationship-annotated corpus collected from computational linguistic paper abstracts. The small volume of scientific KB might hinder the performance of an existing KGC model in scientific domain. For increasing the size of training data and achieving the full potential of a KGC model, one possibility is to manually annotate relation triples such as (*microblog*, INPUT, *sentiment analysis*) in scientific papers and apply the annotated scientific relation triples to scientific KGC. However, manual annotation is time consuming [33] and expensive [2].

To address this issue, in this work, we investigate the effectiveness of leveraging unannotated scientific papers and a RE model for scientific KGC. Specifically, we train a scientific RE model and extract knowledge triples from collections of raw scientific papers, and then use the extracted knowledge triples to enlarge the existing training data for scientific KGC.

Based on the discussion above, we hypothesize that, for acquiring scientific knowledge with high quality and quantity, it would be effective to launch scientific RE and KGC interactively. Therefore we propose a pipeline architecture, which will be detailed in the next section.

5.2 Proposed Model

5.2.1 Framework Formulation

In this paper, for acquiring scientific knowledge, we create a new framework where given a small scientific KB and collections of unannotated scientific papers, we come up with the new pipeline architecture that connects scientific RE and KGC. Specifically, based on raw scientific papers and a trained RE model, we extract new knowledge triples and enrich the original training data of scientific KGC. Thereby we could improve the performance of a scientific KGC model. This, in turn, would enhance the performance of scientific RE by incorporating the embedding learned from the scientific KGC model. The overview of the architecture is illustrated in Figure 5.1. We execute the framework in computational linguistic domain, but we believe that this setting can be easily adapted to other domains, such as knowledge acquisition in biomedical domain.

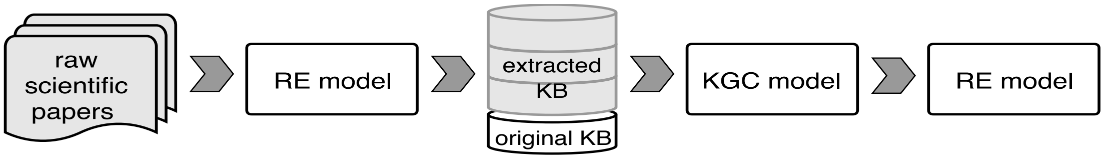


Figure 5.1: Overview of the proposed pipeline architecture

5.2.2 Base Model for Scientific KGC

We select ComplEx [68] as our base scientific KGC model, since it is simple and strong, achieving state-of-the-art predictive performance in general domain. Specifically, suppose we have a KB containing a set of relation triples $\mathcal{O} = \{(e_i, l_k, e_j)\}$, where each relation triple consists of two entities $e_i, e_j \in \mathcal{E}$ and their relation $l_k \in \mathcal{L}$. Here $\mathcal{E}$ and $\mathcal{L}$ stand for the set of entities and relations respectively. ComplEx then encodes each entity $e \in \mathcal{E}$ and relation $l \in \mathcal{L}$ into a complex-valued vector $\mathbf{e} \in \mathbb{C}^d$ and $\mathbf{l} \in \mathbb{C}^d$ respectively, where d is the dimensionality of the embedding space. Since entities and relations are represented as complex-valued vector, each $x \in \mathbb{C}^d$ consists of a real vector component $Re(x)$ and imaginary vector component $Im(x)$, namely $x = Re(x) + iIm(x)$. For a given relation triple $(e_i, l_k, e_j) \in \mathcal{E} \times \mathcal{L} \times \mathcal{E}$, the plausibility of that triple is calculated via Equation 6.3, where $\mathbf{e}_i, \mathbf{l}_k, \mathbf{e}_j \in \mathbb{C}^d$ are vector representations associated with head entity, relation and tail entity respectively; $\bar{\mathbf{e}}_j$ is the conjugate of $\mathbf{e}_j$; $Re(\cdot)$ (or $Im(\cdot)$) means taking the real (or imaginary) part of a complex value. $\langle u, v, w \rangle$ is defined via Equation 6.4, where $[\cdot]_n$ is the n-th entry of a vector.

$$\begin{aligned}\phi(e_i, l_k, e_j) = Re(\langle \mathbf{e}_i, \mathbf{l}_k, \bar{\mathbf{e}}_j \rangle) = \\ \langle Re(\mathbf{l}_k), Re(\mathbf{e}_i), Re(\mathbf{e}_j)\rangle \\ +\langle Re(\mathbf{l}_k), Im(\mathbf{e}_i), Im(\mathbf{e}_j)\rangle \\ +\langle Im(\mathbf{l}_k), Re(\mathbf{e}_i), Im(\mathbf{e}_j)\rangle \\ -\langle Im(\mathbf{l}_k), Im(\mathbf{e}_i), Re(\mathbf{e}_j)\rangle\end{aligned} \tag{5.1}$$

$$\langle u, v, w\rangle = \sum_{n=1}^{d} [u]_n [v]_n [w]_n \tag{5.2}$$

Triple with higher $\phi(\cdot, \cdot, \cdot)$ means more plausible. Since the asymmetry of this scoring function, namely $\phi(e_i, l_k, e_j) \neq \phi(e_j, l_k, e_i)$, ComplEx can effectively encode asymmetric relations [68].

5.2.3 Proposed Model for scientific KGC

The scientific KGC, as mentioned, suffers from the shortage of training data. For increasing the size of training data and achieving the full potential of the base KGC model in scientific domain, we hypothesize that unannotated scientific papers can be utilized as a source of training data. Specifically, we train a RE model, which will be detailed in Section 5.2.4, on a scientific relation annotated corpus, such as RANIS corpus, and apply the trained RE model to extract scientific relation triples from collections of raw scientific papers, and then enlarge an existing training data via these extracted triples. The proposed method for scientific KGC is summarized in Figure 5.2.

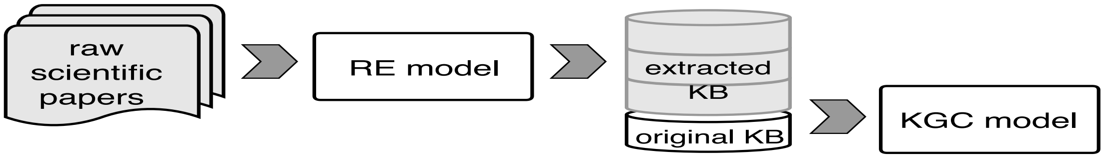


Figure 5.2: Overview of the proposed scientific KGC model

5.2.4 Base Model for Scientific RE

The RC base model is proposed by Santos et al. [61]. As shown in Figure 5.3, it is composed of three layers. The first layer is an embedding layer, which maps each word of the target sentence into a low-dimensional word vector representation. The embedding layer is calculated via Equations 6.5-5.6, where W_{emb}^{w} is a word embedding projection matrix, W_{emb}^{et} is an entity type (ET) projection matrix, x_t^w is a one-hot word representation and x_t^{et} is a one-hot entity type representation. The position vector e_t^{wp} encodes the relative distance between the current word and the head of target entity pair. For instance, in Example 29, the relative distance of the word *"for"* is [-1, 2].

(29) *We introduce referential translation machines* $\overset{entity}{\underline{RTM}}_A$ *for* $\overset{entity}{\underline{quality\ estimation}}_B$ *...*

This relative distance will be encoded into position vectors e_t^{wp1} and e_t^{wp2}, respectively, via Equation 5.5, where W_{emb}^{wp} is a word position embedding projection matrix and x_t^{wp} is a one-hot representation of the relative distance. Word embedding e_t^w, entity type embedding e_t^{et} and word position embedding e_t^{wp1} and e_t^{wp1} are concatenated to create the final word representation e_t. If the dataset does not have entity type information, like SemEval-2018 Task 7 dataset, e_t^{et} will be ignored.

$$e_t^w = W_{emb}^w x_t^w \tag{5.3}$$

$$e_t^{et} = W_{emb}^{et} x_t^{et} \tag{5.4}$$

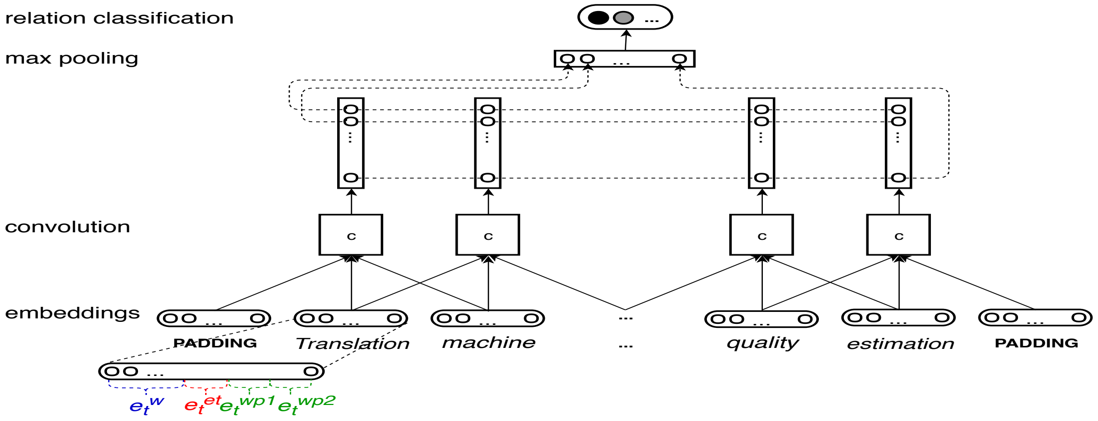


Figure 5.3: Base model architecture

$$e_t^{wp} = W_{emb}^{wp} x_t^{wp} \tag{5.5}$$

$$e_t = concat(e_t^{w}, e_t^{et}, e_t^{wp1}, e_t^{wp2}) \tag{5.6}$$

$$z_t = concat(e_{t-(k-1)/2}, ..., e_{t+(k-1)/2}) \tag{5.7}$$

$$h_t = tanh(W z_t + b) \tag{5.8}$$

The next layer is a convolutional layer, which generates a distributed convolutional window level vector h_t. h_t is calculated by Equations 5.7 and 5.8, where z_t is the concatenated embedding of k words in the convolutional window, k is convolutional window size, and W is the weight matrix of the convolutional layer. In order to address the issue of referencing words with indices outside the sentence boundaries, the target sentence is padded with a special **PADDING** token $(k - 1)/2$ times at the beginning and the end.

The third layer is a max pooling layer, which chooses the maximum value from each dimension of the convolutional window level feature and merges them as the sentence level feature r via Equation 6.6, where i indexes feature dimensions, M is the number of feature dimensions.

$$r_i = \max_t \{(h_t)_i\}, \ \forall i = 1, ..., M \tag{5.9}$$

Finally, the model predicts the semantic relationship between a target entity pair in a target sentence x, by computing the score for a class label $c \in C$ via dot product:

$$S_\theta(x)_c = r^T [W^{class}]_c \tag{5.10}$$

where C is a set of predefined semantic relationships, r is the sentence level feature vector, and W^{class} is the class embedding matrix. The column of W^{class} represents the distributed vector representation of different class labels. It is worth mentioning that the model uses a logistic loss function, as shown in Equation 5.11:

$$\begin{aligned} L = log(1 + exp(\gamma(m^+ - s_\theta(x)_{y^+})) \\ +log(1 + exp(\gamma(m^- + s_\theta(x)_{c^-})) \end{aligned} \tag{5.11}$$

where $s_\theta(x)_{y^+}$ is the score of correct class label, $s_\theta(x)_{c^-}$ is the score of the most competitive incorrect class label, m^+ and m^- are margins, and γ is a scaling factor. In our experiment, we use $m^+ = 2.5, m^- = 0.5$ and $\gamma = 2$.

5.2.5 Proposed Model for Scientific RE

Since a KGC model could learn latent semantic meaning of target entities for relation triple prediction, we hypothesize that incorporating the learned embedding could enhance the performance of a RE model. Therefore, we extend the RE base model via the Equation 5.12 and 5.13, where r is the feature vector from the base RE model, which is calculated via Equation 6.6, e_i and e_j are the target entity pair, $\mathbf{e}_i$ and $\mathbf{e}_j$ are their corresponding complex-valued embedding, which is acquired by the base KGC model. In Equation 5.13, $W^{\mathcal{L}}_{Re}$ (or $W^{\mathcal{L}}_{Im}$) is the projection matrix of real (or imaginary) part of relation embedding learned by the base KGC model. The column of $W^{\mathcal{L}}_{Re}$ (or $W^{\mathcal{L}}_{Im}$) represents the real (or imaginary) part embedding of different relations. $\odot$ is the element-wise product operation. Finally, we replace the original feature vector r in the base RE model with the new final feature vector r_{new} for scientific RE.

$$r_{new} = concat(r, sigmoid(v(e_i, e_j))) \tag{5.12}$$

$$\begin{aligned} v(e_i, e_j) = {} & W^{\mathcal{L}}_{Re}(Re(\mathbf{e}_i) \odot Re(\mathbf{e}_j)) \\ & +W^{\mathcal{L}}_{Re}(Im(\mathbf{e}_i) \odot Im(\mathbf{e}_j)) \\ & +W^{\mathcal{L}}_{Im}(Re(\mathbf{e}_i) \odot Im(\mathbf{e}_j)) \\ & -W^{\mathcal{L}}_{Im}(Im(\mathbf{e}_i) \odot Re(\mathbf{e}_j)) \end{aligned} \tag{5.13}$$

5.3 Experiments

5.3.1 Data

The dataset we use for evaluation in this paper is RANIS corpus [66], a collection of computer science paper abstracts. The type of entity (referred to as Entity Type (ET) hereafter) and domain specific relation in the RANIS corpus has already been annotated with the annotation scheme proposed by [66], as Figure 5.4. The dataset consists of ETs such as QUALITY, PROCESS and DATA-ITEM and domain specific scientific relations, such as INPUT, OUTPUT and APPLY-TO. In total, the RANIS corpus contains 250 abstracts collected from ACL Anthology (230 abstracts in the development set and 20 abstracts in the test set) and 150 abstracts collected from

Figure 5.4: Annotation example shown in brat rapid annotation tool. To more clearly illustrate the direction of relation, we add directional tag “L-” and “R-” before each relation tag.

ACM Digital Library. For training and testing our proposed model, we only use the 250 abstracts from ACL Anthology. From the ACL Anthology abstracts, we extract 11,520 examples[5] from the development set of ACL Anthology and 1,142 examples from the test set of ACL Anthology. These instances are classified into one of the following semantic relations: ORIGIN, COMPARE, EQUIVALENCE, TARGET, OUTPUT, PEFORM, ATTRIBUTE, DESTINATION, RESULT, EVALUATE, APPLY-TO, INPUT, IN-OUT, SUBCONCEPT, POSS, CONDITION, SPLIT and OTHER. We use the ACL Anthology Reference Corpus [5] as the source of training data for scientific KGC. We extract about 400 thousand new relation triples, which achieve high prediction score [6] from the trained RE model and share the same entity pair mentions with RANIS corpus.

5.3.2 Setup

Scientific KGC:

We use the default setting of the base KGC model. Specifically, we sample 1 negative entity for each ground truth entity and use the loss function defined in Equation 6.3. We updates parameters, relation embeddings and entity embeddings (both dimensionality of 200) using $\lambda = 0.1$ for L^2 regularization, and AdaGrad with initial learning rate of 0.5 and mini-batch size of 500.

[5] To our knowledge, there is no large scientific KB available for KGC, therefore, we simply treat each relation example in RANIS corpus as a normal relation triple like the one in Freebase.

[6] We manually set the threshold score as 1.0.

Parameter Name	**Value**
Word Emb. size	200
Word Entity Type Emb. size	50
Word Position Emb. szie	100
Convolutional Units	1000
Context Window size	3
Learning Rate	0.01

Table 5.3: Hyperparameters for Scientific RE

Scientific RE:

Since the most informative part of text to classify the relation type generally exists between and including target entity pair [37, 78], we only utilize this part of the sentence and disregard the surrounding words.

Previous works have shown that scientific papers specific pre-trained word embeddings can improve training for scientific RE models [60, 27, 31, 43]. Therefore, in this work, we trained the scientific papers specific word embeddings on the ACL Anthology Reference Corpus [5] by the skip-gram NN architecture made available by the Gensim word2vec tool. We initialized the word embedding layer with the pre-trained domain-specific word embedding for RE. We randomly extract 10% training data as validation data and based on the performance on it to select all the hyperparameters. All experiments below use the hyperparameters as shown in Table 5.3.

5.3.3 Result and Discussion

Scientific KGC

In this paper, we hypothesize that unannotated scientific papers could be used as a source of training data for a scientific KGC model. We propose a pipeline architecture for scientific KGC, which is illustrated in Figure 5.1. For testing this hypothesis, we compare the performance of the base KGC model trained by enlarged training data with the one only trained by the original training data. We choose the Link prediction task to evaluate the performance of KGC. Link prediction deals with knowledge graph completion: given an entity and a relation, the KGC models predict the other missing entity. Specifically, the task predicts tail entity t given head entity h and relation r, e.g., $(h, r, *)$, or predict head entity h given $(*, r, t)$.

We report the raw MRR (RMRR) for the evaluated models.MRR is defined as: $MRR = \frac{1}{2*|tt|} \sum_{(h,r,t) \in tt} (\frac{1}{rank_h} + \frac{1}{rank_t})$, where tt represents the test triplets. Hit@N is the proportion of the correctly predicted entities (h or t) in top N ranked entities. Table 5.4 presents the performance of the selected KGC model and each extension, where “Original” means the KGC model is only trained by the original training data, “Original+Extracted” means the KGC model is not only trained by the original training data, but also by the relation triples that is extracted from unannotated scientific papers. The percentage indicates the ratio of the extracted triples that is used as training data. It can be seen that the evaluation metric increases with the size of training data. We believe that this is because limited amounts of training data can lead to a problem of low coverage in that many entity pairs encountered at run-time are not observed in training data therefore their embedding for KGC will not be learned. However, by adding in the extracted triples we enlarge the coverage of entity pairs therefore the model could learn their embedding for KGC. This proves the effectiveness of the proposed pipeline architecture for scientific KGC. Specifically, utilizing a trained RE model and collections of raw scientific papers is an effective approach to improve the performance of scientific KGC, especially when the training data is comparatively small.

Training Data	**RMRR**
Original	0.061
Original + Extracted(25%)	0.193
Original + Extracted(50%)	0.229
Original + Extracted(75%)	0.243
Original + Extracted(100%)	0.259

Table 5.4: Link prediction result on RANIS dataset

Scientific RE

In this work, we also hypothesize that the embedding learned by a KGC model could be used to enhance the performance of scientific RE. For testing this hypothesis, we compare the performance of the base RE model with the one that is extended by the learned embedding via Equation 5.12.

Tables 5.5 presents the overall performance of baseline model and each extension. It can be seen that all extension from KGC embedding get better performance than the baseline approach. Table 5.6 detects the influence of our proposed method on each individual relationship. It can be seen that the proposed methods perform better than the baseline approach over a majority of the relationships.

Figure 5.5 compare some actual results between the KGC embedding enhanced model and **Base** model in RE. Take the first line in Figure 5.5 as an example, although there is the preposition “*in*”, which usually appears in relation CONDITION(X,Y) (e.g., “$\underline{\textit{problem}}_Y$ *in* $\underline{\textit{English to Indian language Machine Translation}}_X$”), the KGC embedding enhanced model correctly identify the relation as APPLY_TO rather than CONDITION.

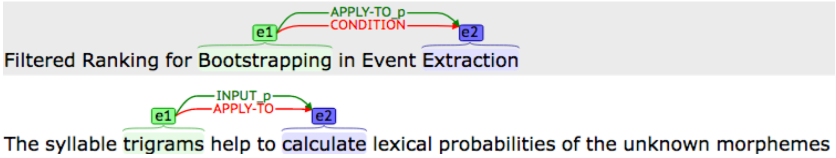
The better performance indicates the following: the entity embedding trained by the selected KGC model could improve the performance of the base RE model, especially

Model	Precision	Recall	F-score
Base	69.34	68.91	67.85
Base + KGC embedding(original)	69.35	68.39	67.97
Base + KGC embedding(original+25%)	71.37	70.75	69.95
Base + KGC embedding(original+50%)	71.81	70.49	70.13
Base + KGC embedding(original+75%)	73.91	70.84	**70.80**
Base + KGC embedding(original+100%)	73.47	70.11	70.40

Table 5.5: RE performance on RANIS dataset

Relationship	Base	Base + KGC embedding				
		original	(+25%)	(+50%)	(+75%)	(+100%)
ATTRIBUTE	79.6	80.1	84.3	83.8	84.0	**85.9**
APPLY_TO	75.2	74.6	76.5	77.2	75.8	**77.6**
OUTPUT	**64.4**	62.5	61.5	61.9	63.2	59.2
INPUT	54.2	54.3	58.5	60.9	**61.3**	57.9
EVALUATE	**97.5**	96.9	96.9	96.9	96.9	96.9
CONDITION	47.1	47.8	50.0	50.4	51.2	**54.2**
EQUIVALENCE	83.7	83.0	83.2	79.7	**84.0**	78.2
SUBCONCEPT	42.9	45.1	47.4	47.8	**49.0**	44.8
PERFORM	93.8	92.2	89.8	87.6	95.5	**97.3**
IN_OUT	53.1	53.6	46.6	**58.6**	50.8	46.5

Table 5.6: RE performance (F-score) over selected relationship



Filtered Ranking for Bootstrapping in Event Extraction

The syllable trigrams help to calculate lexical probabilities of the unknown morphemes

Figure 5.5: Comparison between **Base + KGC embedding (original+75%)** and **Base** in RE, where red lines indicate the error from **Base**, while the green lines show the correctly identified relations (which end with "_p") from KGC embedding enhanced model. e1 and e2 are entity marks.

the embedding obtained by larger training data achieves better performance than the smaller one. Additionally, the results also confirm the feasibility of jointly training the scientific KGC model and the scientific RE model.

5.4 Related Work

Recently, KGC researches have been growing interest in learning vector representations for entities and relations in KB called Knowledge Graph (KG) embedding. [51, 8, 73, 68, 42] prose KG embedding models to predict new facts in a given KB using information from existing entities and relations. Aside from the existing relation triples, external information is applied to improve the KG embedding for KGC. The external information includes surrounding text [56, 73, 82], entity type and relation domain [23, 11], logical rules [70, 57] and cross-lingual triples[34]. However, these methods have not utilized the relation triples that are extracted from unannotated scientific papers via a trained RE model, especially when the training data is comparatively small.

Conventional approaches for RE rely on human-designed, complex lexical-syntactic patterns [9], statistical co-occurrences [65] and structuralized knowledge bases such as WordNet [24, 10]. In recent years, exploring Neural Network (NN)-based models has been the dominant approach in the field. Zeng et al. [79] and Xu et al. [77] proposed a Convolutional Neural Network (CNN)-based framework, which depends on sentence-level features collected from an entire target sentence and lexical-level features from lexical resources such as WordNet [18]. Santos et al. [61] proposed a ranking CNN model, which is trained by a pairwise ranking loss function. To improve the ability of sequential modeling, Zhang et al. [81] proposed a recurrent neural network (RNN)-based model for RC. Other variants of RNN-based models have been proposed, such as Miwa et al. [49], who proposed a bidirectional tree-structured LSTM model. Additionally, similar NN-based approaches are used in scientific relation classification. For instance, Gu et al. [22] utilized a CNN-based model for identifying *chemical-disease* relations from the abstracts of MEDLINE papers. Hahn-Powell et al. [25]

proposed an LSTM-based RNN model for identifying *causal precedence* relationship between two event mentions in biomedical papers. Ammar et al. [1] enhanced Miwa and Bansal [49]'s relation extraction model via extensions such as gazetteer-like information extracted from Wikipedia. Pratp et al. [53] incorporate WordNet hypernyms as the feature for scientific RC. However, none of these approaches leverage the embedding that is trained by a KGC model for RE.

5.5 Conclusion

In this work, we address scientific knowledge acquisition via the collaboration of two sub tasks: scientific KGC and scientific RE. Since scientific KGC and scientific RE are complementary to one another, we propose a pipeline architecture to solve both tasks interdependently. For scientific KGC, we extract new relation triples from a collection of raw scientific papers with a trained RE model, and then enrich the original training data for KGC with the extracted relation triples. Experimental results demonstrated that, firstly, raw scientific papers could be used as a source of training data for scientific KGC, secondly, the proposed pipeline architecture is an effective approach to improve the performance of scientific KGC. For scientific RE, we utilize the learned embedding from the selected KGC model to extend a state-of-the-arts RE model. Experimental results prove that incorporating the embedding from the KGC model could enhance the performance of scientific RE.

Chapter 6

Distantly Supervised Biomedical Knowledge Acquisition via Knowledge Graph Based Attention

6.1 Introduction

Scientific Knowledge Graph (KG), such as Unified Medical Language System (UMLS) [1], is extremely crucial for many scientific Natural Language Processing (NLP) tasks such as Question Answering (QA), Information Retrieval (IR), Relation Extraction (RE), etc. Scientific KG provides large collections of relations between entities, typically stored as (h, r, t) triplets, where h = *head entity*, r = relation and t = *tail entity*, e.g., (*acetaminophen*, *may_treat*, *pain*). However, as with general KGs such as Freebase [7] and DBpedia [38], scientific KGs are far from complete and this would impede their usefulness in real-world applications. Scientific KGs, on the one hand, face the

[1] `https://www.nlm.nih.gov/research/umls/`

data sparsity problem. On the other hand, scientific publications have become the largest repository ever for scientific KGs and continue to increase at an unprecedented rate [50]. Therefore, it is an essential and fundamental task to turn the unstructured scientific publications into well organized KG, and it belongs to the task of RE.

In RE, one obstacle that is encountered when building a RE system is the generation of training instances. For coping with this difficulty, [48] proposes distant supervision to automatically generate training samples via leveraging the alignment between KGs and texts. They assumes that if two entities are connected by a relation in a KG, then all sentences that contain these entity pairs will express the relation. For instance, (*aspirin*, *may_treat*, *pain*) is a fact triplet in UMLS. Distant supervision will automatically label all sentences, such as Example 33, Example 34 and Example 35, as positive instances for the relation *may_treat*. Although distant supervision could provide a large amount of training data at low cost, it always suffers from wrong labelling problem. For instance, comparing to Example 33, Example 34 and Example 35 should not be seen as the evidences to support the *may_treat* relationship between *aspirin* and *pain*, but will still be annotated as positive instances by the distant supervision.

(30) *The clinical manifestations are generally typical nocturnal **pain** that prevents sleep and that is alleviated with **aspirin**.*

(31) *The tumor was remarkably large in size , and**pain** unrelieved by **aspirin**.*

(32) *The level of**pain** did not change significantly with either **aspirin** or pentoxifylline , but the walking distance was farther with the pentoxifylline group .*

To automatically alleviate the wrong labelling problem, [55, 28] apply multi-instance learning. In order to avoid the handcrafted features and errors propagated from NLP tools, [80] proposes a Convolutional Neural Network (CNN), which incorporate mutli-instance learning with neural network model, and achieves significant improvement in distantly supervised RE. Despite the impressive achievement in RE, this model still has the limitation that it only selects the most informative sentence and ignores the rest, thereby loses the rich information stored in those neglected sentences, For instance, among Example 33, Example 34 and Example 35, Example 33 is undoubtedly the most informative one for detecting relation *may_treat*, but it unnecessarily means other sentences such as Example 35 could not contribute to the relation detection. In Example 35, entity *aspirin* and entity *pentoxifylline* have alternative relation, and the latter is a drug to treat muscle pain, therefore the former is also likely to be a pain-killing drug. To address this issue, recently, attention mechanism is applied to extract features from all collected sentences. [41] proposes a relation vector based attention mechanism for distantly supervised RE. [26] proposes a novel joint model that leverages the KG-based attention mechanism and achieves better performance than [41] on distantly supervised RE from New York Times (NYT) corpus.

The success that the joint model [26] has attained in the newswire domain (or non-scientific domain) inspires us to choose the strong model as our base model and assess its feasibility on biomedical domain. Specifically, the first question of this research is how the joint model behaves when the system is trained on biomedical KG (e.g., UMLS) and biomeical corpus (e.g., Medline corpus). [26] indicates that the performance of the base model could be affected the representation ability of KGC model. The representation ability of a KGC model also varies with dataset [71].

Therefore, given a new dataset (e.g., a biomedical dataset), it is necessary to extend the base model with other competitive KGC models, and choose the best fit for the given dataset. However, the base model only implements two KGC models, which are based on TransE [8] and TransD [30] respectively. Thus, the second question of this work is how other competitive KGC models such as ComplEx [68] and SimplE [32] influence the performance of the base model on biomedical dataset. At last but not least, in biomedical KG, a relation is scientifically restricted by entity type (ET). For instance, in the relation (*h*, *may_treat*, *t*), the ET of *t* should be `Disease or Syndrome`. Therefore, ET information is an important feature for biomedical RE and KGC. For leveraging the ET information, which the base model lacks, in this work, we propose an end-to-end KGC model to enhance the base model. The proposed KGC model is capable of identifying ET via the word embedding of target entity and incorporating the predicted ET into a state-of-to-art KGC model to evaluate the plausibility of potential fact triplets.

We conduct evaluation on biomedical datasets in which KG is collected from UMLS and textual data is extracted from Medline corpus. The experimental results not only show the feasibility of the base model on the biomedical domain, but also prove the effectiveness of our proposed extensions for the base model.

6.2 Related Work

RE is a fundamental task in the NLP community. In recent years, Neural Network (NN)-based models have been the dominant approaches for non-scientific RE, which include Convolutional Neural Network (CNN)-based frameworks [79, 77, 61] Recurrent Neural Network (RNN)-based frameworks [81, 49, 84]. NN-based approaches are also used in scientific RE. For instance, [22] utilizes a CNN-based model for identifying *chemical-disease* relations from Medline corpus. [25] proposes an LSTM-based model for identifying *causal precedence* relationship between two event mentions in biomedical papers. [1] applies [49]'s model for scientific RE.

Although remarkably good performances are achieved by the models mentioned above, they still train and extract relations on sentence-level and thus need a large amount of annotation data, which is expensive and time-consuming. To address this issue, distant supervision is proposed by [48]. To alleviate the noisy data from the distant supervision, many studies model distant supervision for RE as a Multiple Instance Learning (MIL) problem [55, 28, 80], in which all sentences containing a target entity pair (e.g.,*aspirin* and *pain*) are seen as a bag to be classified. To make full use of all the sentences in the bag, rather than just the most informative one, [41] proposes a relation vector based attention mechanism to extract feature from the entire bag and outperforms the prior approaches. [26] proposes a joint model that adopts a KG-based attention mechanism and achieves better performance than [41] on distantly supervised RE from NYT corpus.

In this work, we are primarily interested in applying distant supervision techniques to extract biomedical fact triplets from scientific publications. To validate and enhance the efficacy of the previous techniques in biomedical domain, we choose the strong joint model proposed by [26] as the base model and make some necessary extension for our scientific RE task. Since from the two main groups of KGC models [71]: translational distance models and semantic matching models, the base model only implements the translational distance models, TransE [8] and TransD [30], we thus extend the base model with the semantic matching models, ComplEx [68] and SimplE [32], for

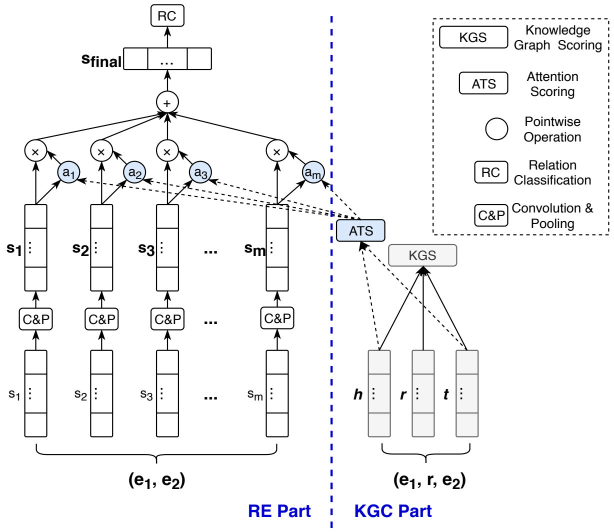


Figure 6.1: Overview of the base model.

selecting the best fit for our task. In addition, the base model has not incorporated the ET information, which we assume is crucial for scientific RE. Therefore, we propose an end-to-end KGC model to enhance the base model. Different from the work [75], which utilizes an ET look-up dictionary to obtain ET, the end-to-end KGC is capable of identifying ET via the word embedding of a target entity and thus is free of the attachment to an incomplete ET look-up dictionary.

6.3 Base Model

The architecture of the base model is illustrated in Figure 6.1. In this section, we will introduce the base model proposed by [26] in two main parts: KGC part, RE part.

6.3.1 KGC Part

Suppose we have a KG containing a set of fact triplets $O = \{(e_1, r, e_2)\}$, where each fact triplet consists of two entities $e_1, e_2 \in \mathcal{E}$ and their relation $r \in \mathcal{R}$. Here $\mathcal{E}$ and $\mathcal{R}$ stand for the set of entities and relations respectively. KGC model then encodes $e_1, e_2 \in \mathcal{E}$ and their relation $r \in \mathcal{R}$ into low-dimensional vectors $\mathbf{h}, \mathbf{t} \in R^d$ and $\mathbf{r} \in R^d$ respectively, where d is the dimensionality of the embedding space. As mentioned above, the base model adopts two representative translational distance models Prob-TransE and Prob-TransD, which are based on TransE [8] and TransD [30] repectively, to score a fact triplet. Specifically, given an entity pair (e_1, e_2), Prob-TransE defines its latent relation embedding $\mathbf{r}_{ht}$ via the Equation 6.1.

$$\mathbf{r}_{ht} = \mathbf{t} - \mathbf{h} \tag{6.1}$$

Prob-TransD is an extension of Prob-TransE and introduces additional mapping vectors $\mathbf{h}_p$, $\mathbf{t}_p \in R^d$ and $\mathbf{r}_p \in R^d$ for e_1, e_2 and r respectively. Prob-TransD encodes the latent relation embedding via the Equation 6.2, where $\mathbf{M}_{rh}$ and $\mathbf{M}_{rt}$ are projection matrices for mapping entity embeddings into relation spaces.

$$\mathbf{r}_{ht} = \mathbf{t}_r - \mathbf{h}_r, \tag{6.2}$$

$$\begin{aligned}
\mathbf{h}_r &= \mathbf{M}_{rh}\mathbf{h}, \\
\mathbf{t}_r &= \mathbf{M}_{rt}\mathbf{t}, \\
\mathbf{M}_{rh} &= \mathbf{r}_p\mathbf{h}_p^\top + \mathbf{I}^{d\times d}, \\
\mathbf{M}_{rt} &= \mathbf{r}_p\mathbf{t}_p^\top + \mathbf{I}^{d\times d}
\end{aligned}$$

The conditional probability can be formalized over all fact triplets O via the Equations 6.3 and 6.4, where $f_r(e_1, e_2)$ is the KG scoring function, which is used to evaluate the plausibility of a given fact triplet. For instance, the score for (*aspirin*, *may_treat*, *pain*) would be higher than the one for (*aspirin*, *has_ingredient*, *pain*), because the former is more plausible than the latter. $\theta_{\mathcal{E}}$ and $\theta_{\mathcal{R}}$ are parameters for entities and relations respectively, b is a bias constant.

$$P(r|(e_1, e_2), \theta_{\mathcal{E}}, \theta_{\mathcal{R}}) = \frac{\exp(f_r(e_1, e_2))}{\sum_{r' \in \mathcal{R}} \exp(f_{r'}(e_1, e_2))} \tag{6.3}$$

$$f_r(e_1, e_2) = b - \|\mathbf{r}_{ht} - \mathbf{r}\| \tag{6.4}$$

6.3.2 RE Part

Sentence Representation Learning. Given a sentence s with n words $s = \{w_1, ..., w_n\}$ including a target entity pair (e_1, e_2), CNN is used to generate a distributed representation $\mathbf{s}$ for the sentence. Specifically, vector representation $\mathbf{v}_t$ for each word w_t is calculated via Equation 6.5, where $\mathbf{W}^{w}_{emb}$ is a word embedding projection matrix [46], $\mathbf{W}^{wp}_{emb}$ is a word position embedding projection matrix, $\mathbf{x}^{w}_{t}$ is a one-hot word representation and $\mathbf{x}^{wp}_{t}$ is a one-hot word position representation. The word position describes the relative distance between the current word and the target entity pair [79]. For instance, in the sentence *"Patients recorded* $\underline{\text{pain}}_{e_2}$ *and* $\underline{\text{aspirin}}_{e_1}$ *consumption in a daily diary"*, the relative distance of the word *"and"* is [1, -1].

$$\mathbf{v}_t = [\mathbf{v}^{w}_{t}; \mathbf{v}^{wp1}_{t}; \mathbf{v}^{wp2}_{t}], \tag{6.5}$$

$$\mathbf{v}_t^{w} = \mathbf{W}_{emb}^{w}\mathbf{x}_t^{w},$$
$$\mathbf{v}_t^{wp1} = \mathbf{W}_{emb}^{wp}\mathbf{x}_t^{wp1},$$
$$\mathbf{v}_t^{wp2} = \mathbf{W}_{emb}^{wp}\mathbf{x}_t^{wp2}$$

The distributed representation $\mathbf{s}$ is formulated via the Equation 6.6, where, $[\mathbf{s}]_i$ and $[\mathbf{h}_t]_i$ are the i-th value of $\mathbf{s}$ and $\mathbf{h}_t$, M is the dimensionality of $\mathbf{s}$, $\mathbf{W}$ is the convolution kernal, $\mathbf{b}$ is a bias vector, and k is the convolutional window size.

$$[\mathbf{s}]_i = \max_t\{[\mathbf{h}_t]_i\},\ \forall i = 1, ..., M \tag{6.6}$$

$$\mathbf{h}_t = \tanh(\mathbf{W}\mathbf{z}_t + \mathbf{b}),$$
$$\mathbf{z}_t = [\mathbf{v}_{t-(k-1)/2}; ...; \mathbf{v}_{t+(k-1)/2}]$$

KG-based Attention. Suppose for each fact triplet (e_1, r, e_2), there might be multiple sentences $S_r = \{s_1, ..., s_m\}$ in which each sentence contains the entity pair (e_1, e_2) and is assumed to imply the relation r, m is the size of S_r. As discussed before, the distant supervision inevitably collect noisy sentences, the base model adopts a KG-based attention mechanism to discriminate the informative sentences from the noisy ones. Specifically, the base model use the latent relation embedding $\mathbf{r}_{ht}$ from Equation 6.1 (or Equation 6.2) as the attention over S_r to generate its final representation $\mathbf{s}_{final}$. $\mathbf{s}_{final}$ is calculated via Equation 6.7, where $\mathbf{W}_s$ is the weight matrix, $\mathbf{b}_s$ is the bias vector, a_i is the weight for $\mathbf{s}_i$, which is the distributed representation for the i-th sentence in S_r.

$$\mathbf{s}_{final} = \sum_{i=1}^{m} a_i\mathbf{s}_i, \tag{6.7}$$

$$a_i = \frac{\exp(\langle \mathbf{r}_{ht}, \mathbf{x}_i \rangle)}{\sum_{k=1}^{m} \exp(\langle \mathbf{r}_{ht}, \mathbf{x}_k \rangle)},$$

$$\mathbf{x}_i = \tanh(\mathbf{W}_s \mathbf{s}_i + \mathbf{b}_s)$$

Finally, the conditional probability $P(r|S_r, \theta)$ is formulated via Equation 6.8 and Equation 6.9, where, θ is the parameters for RE, which includes $\{\mathbf{W}^{w}_{emb}, \mathbf{W}^{wp}_{emb}, \mathbf{W}, \mathbf{b}, \mathbf{W}_s, \mathbf{b}_s, \mathbf{M}, \mathbf{d}\}$, $\mathbf{M}$ is the representation matrix of relations, $\mathbf{d}$ is a bias vector, $\mathbf{o}$ is the output vector containing the prediction probabilities of all target relations for the input sentences set S_r, and n_r is the total number of relations.

$$P(r|S_r, \theta) = \frac{\exp(\mathbf{o}_r)}{\sum_{c=1}^{n_r} \exp(\mathbf{o}_c)} \tag{6.8}$$

$$\mathbf{o} = \mathbf{M}\mathbf{s}_{final} + \mathbf{d} \tag{6.9}$$

6.3.3 Optimization

The base model defines the optimization function as the log-likelihood of the objective function in Equation 6.10.

$$P(G, D|\theta) = P(G|\theta_{\mathcal{E}}, \theta_{\mathcal{R}}) + P(D|\theta_S) \tag{6.10}$$

where, G and D are KG and textual data respectively. The base model applies Stochastic Gradient Descent (SGD) and L_2 regularization. In practice, the base model optimizes the KG Encoding Part and Sentence Encoding Part in parallel.

6.4 Extensions

The base model opens the possibility to jointly train RE models with KGC models for distantly supervised RE. The empirical results of the base model on NYT corpus indicate that the performance of distantly supervised RE varies with KGC models [26]. In addition, the performance of KGC models depends on a given dataset [71]. Therefore, we assume that it is necessary to attempt multiple competitive KGC models for the joint framework so as to find the optimal combination for our biomedical dataset. However, the base model only implements translational distance models: TransE and TransD, but not the semantic matching models, and this, we assume, might hinder its performance in the new dataset. To address this, we select two representative semantic matching models: ComplEx [68] and SimplE [32] as the alternative KGC part.

As discussed in Section 8.1, in scientific KGs, a fact triplet is severely restricted by ET information (e.g., ET of e_2 should be `Disease or Syndrome` in the fact triplet (e_1, may_treat, e_2)). Therefore, for leveraging ET information, which the base model lacks, we also propose an end-to-end KGC model to extend the base model. Since the proposed KGC model is build on SimplE and is capable of Named Entity Recognition (NER), we call it SimplE_NER.

6.4.1 ComplEx based Attention

Given a fact triplet (e_1, r, e_2), ComplEx then encodes entities e_1, e_2 and relation r into a complex-valued vector $\mathbf{e}_1 \in \mathbb{C}^d$, $\mathbf{e}_2 \in \mathbb{C}^d$ and $\mathbf{r} \in \mathbb{C}^d$ respectively, where d is the dimensionality of the embedding space. Since entities and relations are represented as complex-valued vector, each $\mathbf{x} \in \mathbb{C}^d$ consists of a real vector component $Re(\mathbf{x})$ and imaginary vector component $Im(\mathbf{x})$, namely $\mathbf{x} = Re(\mathbf{x}) + iIm(\mathbf{x})$. The KG scoring function of ComplEx for a fact triplet (e_1, r, e_2) is calculated via Equation 6.11, where $\bar{\mathbf{e}}_2$ is the conjugate of $\mathbf{e}_2$; $Re(\cdot)$ (or $Im(\cdot)$) means taking the real (or imaginary) part of a complex value. $\langle u, v, w \rangle$ is defined via Equation 6.12, where $[\cdot]_n$ is the n-th entry of a vector.

$$\begin{aligned} f_r(e_1, e_2) = Re(\langle \mathbf{e}_1, \mathbf{r}, \bar{\mathbf{e}}_2 \rangle) = \\ \langle Re(\mathbf{r}), Re(\mathbf{e}_1), Re(\mathbf{e}_2) \rangle \\ + \langle Re(\mathbf{r}), Im(\mathbf{e}_1), Im(\mathbf{e}_2) \rangle \\ + \langle Im(\mathbf{r}), Re(\mathbf{e}_1), Im(\mathbf{e}_2) \rangle \\ - \langle Im(\mathbf{r}), Im(\mathbf{e}_1), Re(\mathbf{e}_2) \rangle \end{aligned} \tag{6.11}$$

$$\langle \mathbf{u}, \mathbf{v}, \mathbf{w} \rangle = \sum_{n=1}^{d} [\mathbf{u}]_n [\mathbf{v}]_n [\mathbf{w}]_n \tag{6.12}$$

Since the asymmetry of this scoring function, namely $f_r(e_1, e_2) \neq f_r(e_2, e_1)$, ComplEx can effectively encode asymmetric relations [68]. For calculating the attention, the $\mathbf{r}_{ht}$ in Equation 6.7 is defined via Equation 6.13, where $\odot$ represents the element-wise multiplication.

$$\mathbf{r}_{ht} = Re(\mathbf{e}_1) \odot Re(\mathbf{e}_2) + Im(\mathbf{e}_1) \odot Im(\mathbf{e}_2) \tag{6.13}$$

6.4.2 SimplE based Attention

Given a fact triplet (e_1, r, e_2), SimplE then encodes each entity $e \in \mathcal{E}$ into two vectors $\mathbf{h}_e$, $\mathbf{t}_e \in R^d$ and each relation $r \in \mathcal{R}$ into two vectors $\mathbf{v}_r$, $\mathbf{v}_{r^{-1}} \in R^d$ respectively, where d is the dimensionality of the embedding space. $\mathbf{h}_e$ captures the entity e's behaviour as the *head entity* of a fact triplet and $\mathbf{t}_e$ captures e's behaviour as the *tail entity*. $\mathbf{v}_r$

represents r in a fact triplet (e_1, r, e_2), while $\mathbf{v}_{r^{-1}}$ represents its inverse relation r^{-1} in the triplet (e_2, r^{-1}, e_1). The KG scoring function of SimplE for a fact triplet (e_1, r, e_2) is defined via Equation 6.14.

$$f_r(e_1, e_2) = \frac{1}{2}(\langle \mathbf{h}_{e_1}, \mathbf{v}_r, \mathbf{t}_{e_2} \rangle + \langle \mathbf{h}_{e_2}, \mathbf{v}_{r^{-1}}, \mathbf{t}_{e_1} \rangle) \tag{6.14}$$

Similar to the attention from ComplEx, the $\mathbf{r}_{ht}$ in Equation 6.7 is defined via Equation 6.15.

$$\mathbf{r}_{ht} = \frac{1}{2}(\mathbf{h}_{e_1} \odot \mathbf{h}_{e_2} + \mathbf{t}_{e_1} \odot \mathbf{t}_{e_2}) \tag{6.15}$$

6.4.3 SimplE_NER based Attention

The proposed end-to-end KGC model is based on SimplE, because SimplE outperforms several state-of-the-art models including ComplEx [32]. The proposed model is illustrated in Figure 6.2. It includes ET classification part (below) and KG Scoring part (above). In ET classification part, a multi-layer perceptron (MLP) with two hidden layers are applied to identify ET based on word embedding of target entity. In KG Scoring part, *head entity* and *tail entity* along with their predicted ETs and their relation are projected into corresponding KG embeddings, which are then fed to a KG scoring function.

ET Classification Part. In this work, we use a MLP network to classify ET for *head entity* and *tail entity*. The architecture of our MLP network is as bellow:

$$
\begin{aligned}
\mathbf{h}_w &= \tanh(\mathbf{W}^w_{emb}\mathbf{x}^w), \\
\mathbf{h}_1 &= \mathrm{sigmoid}(\mathbf{W}_1\mathbf{h}_w + \mathbf{b}_1), \\
\mathbf{h}_2 &= \mathrm{sigmoid}(\mathbf{W}_2\mathbf{h}_1 + \mathbf{b}_2), \\
\mathbf{y} &= \mathrm{sigmoid}(\mathbf{W}_{ET}\mathbf{h}_2 + \mathbf{b}_{ET})
\end{aligned}
\tag{6.16}
$$

where $\mathbf{W}^w_{emb}$ is a word embedding projection matrix, which is initialized by the pre-trained word embedding that is trained on Medline corpus via Gensim word2vec tool, $\mathbf{x}^w$ is a one-hot entity representation, $\mathbf{y}$ is the output vector containing the prediction probabilities of all target ETs. $\mathbf{W}_1$, $\mathbf{b}_1$, $\mathbf{W}_2$, $\mathbf{b}_2$, $\mathbf{W}_{ET}$ and $\mathbf{b}_{ET}$ are parameters to optimize.

KG Scoring Part. Given fact triplet and predicted ET pair ET_1 (for e_1) and ET_2 (for e_2), the proposed model project them into their corresponding KG embeddings namely $\mathbf{h}_{e_1}$, $\mathbf{t}_{e_1}$, $\mathbf{v}_r$, $\mathbf{v}_{r^{-1}}$, $\mathbf{h}_{e_2}$, $\mathbf{t}_{e_2}$, $\mathbf{h}_{ET_1}$, $\mathbf{t}_{ET_1}$, $\mathbf{h}_{ET_2}$ and $\mathbf{t}_{ET_2}$ respectively, where $\mathbf{h}_{ET_1}$ (or $\mathbf{t}_{ET_1}$) represents the KG embedding of ET for e_1 when e_1 acts as the *head entity* (or *tail entity*) in a fact triplet. The KG scoring function is defined via Equation 6.17. Since the proposed KGC model is build on SimplE, we apply Equation 6.15 to calculate $\mathbf{r}_{ht}$.

$$
\begin{aligned}
f_r(e_1, e_2) = \frac{1}{4}(&\langle \mathbf{h}_{e_1}, \mathbf{v}_r, \mathbf{t}_{e_2} \rangle \\
&+\langle \mathbf{h}_{e_2}, \mathbf{v}_{r^{-1}}, \mathbf{t}_{e_1} \rangle \\
&+\langle \mathbf{h}_{ET_1}, \mathbf{v}_r, \mathbf{t}_{ET_2} \rangle \\
&+\langle \mathbf{h}_{ET_2}, \mathbf{v}_{r^{-1}}, \mathbf{t}_{ET_1} \rangle)
\end{aligned}
\tag{6.17}
$$

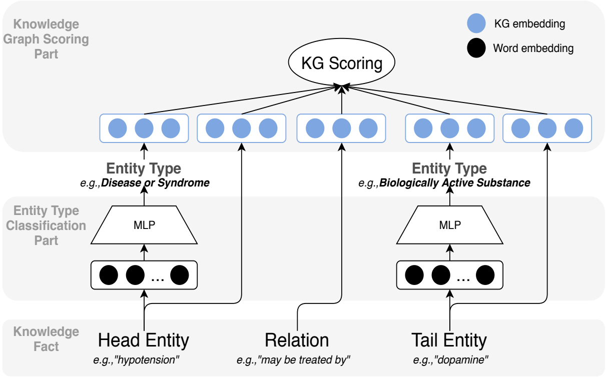


Figure 6.2: Overview of the proposed end-to-end KGC model.

6.5 Experiments

Our experiments aim to demonstrate that, (1) the base model proposed by [26] is feasible for biomedical dataset, such as UMLS and Medline corpus, and (2) in order to improve the performance on the given biomedical dataset, it is necessary to extend the base model with other competitive KGC models, such as ComplEx and SimplE, and (3) the proposed end-to-end KGC model is effective for distantly supervised RE from biomedical dataset.

6.5.1 Data

The biomedical datasets used for evaluation consist of biomedical knowledge graph and biomedical textual data, which will be detailed as follows.

Knowledge Graph. We choose the UMLS as the KG. UMLS is a large biomedical knowledge base developed at the U.S. National Library of Medicine. UMLS contains millions of biomedical concepts and relations between them. We follow [73], and only collect the fact triplet with RO relation category (RO stands for “has Relationship Other than synonymous, narrower, or broader”), which covers the interesting relations

#Entity	#Relation	#Train	#Test
25,080	360	53,036	11,810

Table 6.1: Statistics of KG in this work.

like *may_treat*, *my_prevent*, etc. From the UMLS 2018 release, we extract about 60 thousand such RO fact triplets (i.e., (e_1, r, e_2)) under the restriction that their entity pairs (i.e., e_1 and e_2) should coexist within a sentence in Medline corpus. They are then randomly divided into training and testing sets for KGC. Following [74], we keep high entity overlap between training and testing set, but zero fact triplet overlap. The statistics of the extracted KG is shown in Table 7.1. For training the ET Classification Part in Section 6.4.3, we also collect about 35 thousand entity-ET pairs (e.g., *heart rates*-`Clinical Attribute`) from the UMLS 2018 release.

Textual Data. Medline corpus is a collection of bimedical abstracts maintained by the National Library of Medicine. From the Medline corpus, by applying a string matching model [2], we extract $732,771$ sentences that contain the entity pairs (i.e., e_1 and e_2) in the KG mentioned above as our textual data, in which $592,605$ sentences are for training and $140,166$ sentences for testing. For identifying the NA relation, besides the "related" sentences, we also extract the "unrelated" sentences based on a closed world assumption: pairs of entities not listed in the KG are regarded to have NA relation and sentences containing them considered to be the "unrelated" sentences. By this way, we extract $1,738,801$ "unrelated" sentences for the training data, and $431,212$ "unrelated" sentences for the testing data. Table 7.2 presents some sample sentences in the training data.

[2] We adopt the NER model that is available at `https://github.com/mpuig/spacy-lookup`.

Fact Triplet	Textual Data
(insulin, gene_plays_role_in_process, lipid_metabolism)	s_1 : *It is unknown whether short - term angiotensin_receptor blocker therapy can improve glucose and lipid_metabolism*$_{e_2}$ *in insulin*$_{e_1}$ *- resistant subjects.* s_2 : *Adipocyte lipid_metabolism*$_{e_2}$ *is primarily regulated by insulin*$_{e_1}$ *and the catecholamines norepinephrine and epinephrine.* s_3 : ...
(insulin, NA, TPA)	s_1 : *M wortmannin resulted in 80% and 20% decreases of glucose uptake stimulated by insulin*$_{e_1}$ *and TPA*$_{e_2}$ *, respectively.* s_2 : *The effects of insulin*$_{e_1}$ *, IGF1 and TPA*$_{e_2}$ *were also observed in the presence of cycloheximide.* s_3 : ...

Table 6.2: Examples of textual data extracted from Medline corpus.

6.5.2 Parameter Settings

We base our work on [26] and extend their implementation available at `https://github.com/thunlp/JointNRE`, and thus adopt identical optimization process. We use the default settings of parameters [3] provided by the base model. Since we address the distantly supervised RE in biomedical domain, we use the Medline corpus to train the domain specific word embedding projection matrix $\mathbf{W}^{w}_{emb}$.

6.5.3 Result and Discussion

[26] evaluates the base model on non-scientific dataset. In this work, we firstly plan to assess its feasibility on scientific dataset, and secondly, to investigate the effectiveness of our extensions, which is discussed in Section 6.4, with respect to enhancing the distantly supervised RE from scientific dataset.

Relation Extraction We follow [48, 74, 41, 26] and conduct the held-out evaluation, in which the model for distantly supervised RE is evaluated by comparing the fact triplets identified from textual data (i.e., the bag of sentences containing the target entity pairs)

[3] As a preliminary study, we only adopt the default hyperparameters, but we will tune them in the furture.

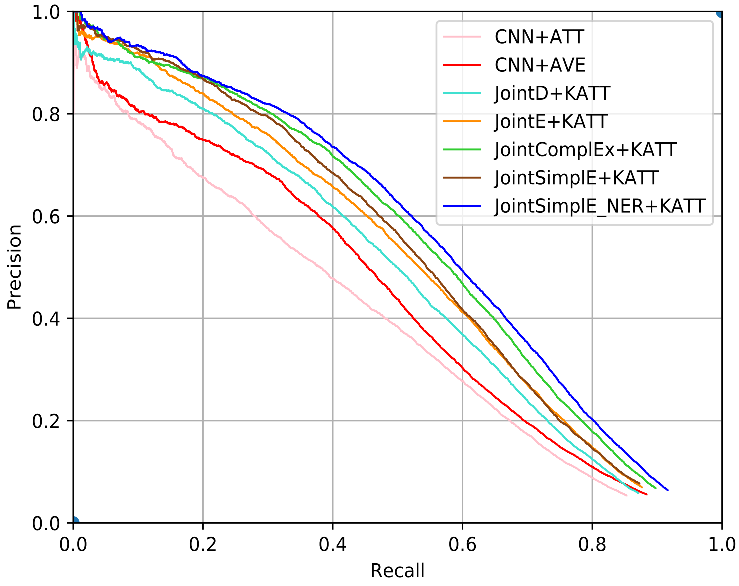


Figure 6.3: Aggregate precision/recall curves for different RE models.

with those in KG. We report precision-recall curves and Precision@N (P@N) as well in our evaluation.

The precision-recall curves are shown in Figure 6.3, where "JointD+KATT" and "JointE+KATT" represent the RE model with the KG-based attention obtained from Prob-TransD and Prob-TransE respectively, which are our base models and trained on both KG and textual data. Similarly, "JointComplEx+KATT", "JointSimplE+KATT" and "JointSimplE_NER+KATT" represent the RE model with the KG-based attention obtained from ComplEx, SimplE and SimplE_NER respectively, which are our extensions. "CNN+AVE" and "CNN+ATT" represent the RE model with average attention and relation vector based attention [41] respectively, which are not joint models and only trained on textual data. The results show that:

(1) All RE models with KG-based attention, such as "JointE+KATT", outperform those models without it, such as "CNN+ATT". This observation is in line with [26].

This demonstrates that not just for non-scientific dataset , jointly training a KGC model with a RE model is also an effective approach to improve the performance of distantly supervised RE for biomedical dataset. In other words, the outperformance proves the feasibility of the base model proposed by [26] on biomedical dataset. The comparison between [26]'s results on non-scientific dataset and ours on scientific dataset also indicates that the performance of base model could differ according to the dataset. Specifically, on scientific dataset, "JointE+KATT" performs better than "JointD+KATT" but in non-scientific dataset the latter outperforms the former.

(2) Our extended models, "JointComplEx+KATT", "JointSimplE+KATT" and "JointSimplE_NER+KATT", achieve better precision than the base model over the major range of recall. It could be attributed to their better capability of modeling asymmetric relations (e.g., may_treat and $may\_prevent$), because their KG scoring functions are asymmetry (i.e., $f_r(e_1, e_2) \neq f_r(e_2, e_1)$). The superior performance indicates the necessity of our extensions on the base model. Specifically, given the frequently used biomedical dataset, UMLS and Medline corpus, it would be an effective method to switch the translational distance models, such as TransE and TransD, with the semantic matching models, such as ComplEx and SimplE, for increasing the performance of distantly supervised RE. The effect of different KGC models on the distantly supervised RE will be discussed later.

(3) The model enhanced by our proposed KGC model, "JointSimplE_NER+KATT", achieves the highest precision over almost entire range of recall compared with the models that apply the existing KGC models. This proves the effectiveness of our proposed KGC model for the distantly supervised RE. Additionally, different from the exiting KGC models, the proposed end-to-end KGC model is capable of identifying ET

information from word embedding of target entity. This indicates that the incorporation of semantic information of entity, such as ET, is a promising approach for enhancing the base model.

Effect of KGC on RE. [26] indicates that KGC models could affect the performance of distantly supervised RE. For investigating the influence of KGC models on our specific RE task, we compare their link prediction results on our KG with their corresponding Precision@N (P@N) results on our RE task. Link prediction is the task that predicts *tail entity* t given both *head entity* h and relation r, e.g., $(h, r, *)$, or predict *head entity* h given $(*, r, t)$. We report the mean reciprocal rank (MRR) and mean Hit@N scores for evaluating the KGC models. MRR is defined as: $MRR = \frac{1}{2*|tt|} \sum_{(h,r,t)\in tt}(\frac{1}{rank_h} + \frac{1}{rank_t})$, where tt represents the test triplets. Hit@N is the proportion of the correctly predicted entities (h or t) in top N ranked entities. Table 6.3 and Table 6.4 represent the RE precision@N and link prediction results respectively. This comparison indicates that given a biomedical dataset, the performance of a KGC model on the link prediction task could predict its effectiveness on its corresponding distantly supervised RE task. This observation also instruct us how to select the best KGC model for the base model. In addition, Table 6.3 and Table 6.4 indicate that ET is not only effective for distantly supervised RE task, but also for KGC task, and this observation will inspire us to explore other useful semantic feature of entity, such as the definition of entity, for our task.

Model	**P@2k**	**P@4k**	**P@6k**	**Mean**
JointE+KATT	0.876	0.786	0.698	0.786
JointD+KATT	0.848	0.725	0.528	0.700
JointComplEx+KATT	0.892	0.819	0.741	0.817
JointSimplE+KATT	0.900	0.808	0.721	0.809
JointSimplE_NER+KATT	**0.913**	**0.829**	**0.753**	**0.831**

Table 6.3: P@N for different RE models, where k=1000.

	MRR		**Hit@**		
Model	**Raw**	**Filter**	**1**	**3**	**10**
TransE	0.156	0.200	0.113	0.244	0.356
TransD	0.138	0.149	0.098	0.160	0.245
ComplEx	0.278	0.457	0.380	0.507	0.587
SimplE	0.273	0.455	0.368	0.516	0.598
SimplE_NER	**0.339**	**0.538**	**0.473**	**0.578**	**0.651**

Table 6.4: Link prediction results for different KGC models.

6.6 Conclusion and Future Work

In this work, we tackle the task of distantly supervised RE from biomedical publications. To this end, we apply the strong joint framework proposed by [26] as the base model. For enhancing its performance on our specific task, we extend the base model with other competitive KGC models. What is more, we also propose a new end-to-end KGC model, which incorporates word embedding based entity type information into a sate-of-the-art KGC model. Experimental results not only show the feasibility of the base model on the biomedical domain, but also indicate the effectiveness of our extensions. Our extended model achieves significant and consistent improvements on the biomedical dataset as compared with baselines. Since the semantic information of target entity, such as ET information, is effective for our task, in the future, we will explore other useful semantic features, such as the definition of target entity and fact triplet chain between entities (e.g., cancer→disease_has_associated_gene→ Ku86→gene_plays_role_in_process→NHEJ), for our task.

Chapter 7

Incorporating Chains of Reasoning over Knowledge Graph for Distantly Supervised Biomedical Knowledge Acquisition

7.1 Introduction

Scientific Knowledge Graph (KG), such as Unified Medical Language System (UMLS) [1], is extremely crucial for many scientific Natural Language Processing (NLP) tasks such as Question Answering (QA), Information Retrieval (IR) and Relation Extraction (RE). Scientific KG provides large collections of relations between entities, typically stored as (h, r, t) triplets, where h = *head entity*, r = relation and t = *tail entity*, e.g.,

[1] https://www.nlm.nih.gov/research/umls/

(*acetaminophen*, *may_treat*, *pain*). However, KGs are often highly incomplete [47]. Scientific KGs, as with general KGs such as Freebase [7] and DBpedia [38], are far from complete and this would impede their usefulness in real-world applications. Scientific KGs, on the one hand, face the data sparsity problem. On the other hand, scientific publications have become the largest repository ever for scientific KGs and continue to increase at an unprecedented rate [50]. Therefore, it is an essential and fundamental task to turn the unstructured scientific publications into well organized KG, and it belongs to the task of RE.

One obstacle that is encountered when building a RE system is the generation of training instances. For coping with this difficulty, [48] proposes distant supervision to automatically generate training samples via leveraging the alignment between KGs and texts. They assume that if two entities are connected by a relation in a KG, then all sentences that contain those entity pairs will express the relation. For instance, (*ketorolac_tromethamine*, *may_treat*, *pain*) is a fact triplet in UMLS. Distant supervision will automatically label all sentences, such as Example 33, Example 34 and Example 35, as positive instances for the relation *may_treat*. Although distant supervision could provide a large amount of training data at low cost, it always suffers from wrong labelling problem. For instance, comparing to Example 33, Example 34 and Example 35 should not be seen as the convincing evidences to support the *may_treat* relationship between *ketorolac_tromethamine* and *pain*, but will still be annotated as positive instances by the distant supervision.

(33) *The analgesic effectiveness of **ketorolac_tromethamine** was compared with hydrocodone and acetaminophen for **pain** from an arthroscopically assisted patellar-tendon autograft anterior cruciate ligament reconstruction.*

(34) *This double-blind, split-mouth, and randomized study was aimed to compare the efficacy of dexamethasone and* ***ketorolac_tromethamine****, through the evaluation of* ***pain****, edema, and limitation of mouth_opening.*

(35) *A loading dose of parental* ***ketorolac_tromethamine*** *was administered and subjects were later given two staged doses of the same "unknown" drug with* ***pain*** *evaluations conducted after each dose.*

To automatically alleviate the wrong labelling problem, [55, 28] apply multi-instance learning. In order to avoid the handcrafted features and errors propagated from NLP tools, [80] proposes a Convolutional Neural Network (CNN), which incorporate mutli-instance learning with neural network model, and achieves significant improvement in distantly supervised RE (DS-RE). Recently, attention mechanism is applied to effectively extract features from all collected sentences, rather than from the most informative one that previous work has focused on. [41] proposes a relation vector based attention mechanism for DS-RE. [26] proposes a novel joint model that leverages a KG-based attention mechanism and achieves significant improvement than [41].

Although the KG-based model outperforms several state-of-the-art DS-RE models, the brevity of textual information would inevitably hinder its performance. Specifically, authors always leave out information that they assume is known to their readers. For instance, Example 34 omits the background connection between *ketorolac_tromethamine* and *pain* and implicitly conveys that the former *may_treat* the latter. Human readers could easily make this inference based on their Background Knowledge (BK) about the target entity pair. However, for a machine, it would be extremely difficult to identify the relationship just from the given sentence without the important BK.

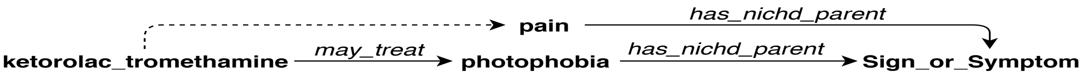


Figure 7.1: An example of reasoning path.

To address the issue of textual brevity, in this work, we assume that the paths (or reasoning paths) between an entity pair over a KG could be applied as the BK to fill the "gaps" and thereby improve the performance of DS-RE. For instance, one reasoning path between *ketorolac_tromethamine* and *pain* over UMLS is shown in Figure 7.1. By observing the path, we may infer with some likelihood that (*ketorolac_tromethamine*, *may_treat*, *pain*), because *ketorolac_tromethamine* could be prescribed to treat some *Sign_or_Symptom* such as *photophobia*, and *pain* is a *Sign_or_Symptom*, therefore *ketorolac_tromethamine* might be used to treat *pain*. By comprehensively considering the path in Figure 7.1 and the sentence in Example 34, we could further prove the inference. To this end, we propose the DS-RE model that not only encodes the sentences containing target entity pairs, but also the reasoning paths between them over a KG.

We conduct evaluation on biomedical datasets in which KG is collected from UMLS and textual data is extracted from Medline corpus. The experimental results prove the effectiveness of the incorporation of reasoning paths for improving DS-RE from biomedical datasets.

7.2 Related Work

RE is a fundamental task in the NLP community. In recent years, Neural Network (NN)-based models have been the dominant approaches for non-scientific RE, which include

Convolutional Neural Network (CNN)-based frameworks [79, 77, 61] Recurrent Neural Network (RNN)-based frameworks [81, 49, 84]. NN-based approaches are also used in scientific RE. For instance, [22] utilizes a CNN-based model for identifying *chemical-disease* relations from Medline corpus. [25] proposes an LSTM-based model for identifying *causal precedence* relationship between two event mentions in biomedical papers. [1] applies [49]'s model for scientific RE.

Although remarkably good performances are achieved by the models mentioned above, they still train and extract relations on sentence-level and thus need a large amount of annotation data, which is expensive and time-consuming. To address this issue, distant supervision is proposed by [48]. To alleviate the noisy data from the distant supervision, many studies model DS-RE as a Multiple Instance Learning (MIL) problem [55, 28, 80], in which all sentences containing a target entity pair (e.g., *ketorolac_tromethamine* and *pain*) are seen as a bag to be classified. To make full use of all the sentences in the bag, rather than just the most informative one in the bag, researchers apply attention mechanism in deep NN-based models for DS-RE. [41] proposes a relation vector based attention mechanism to extract feature from the entire bag and outperforms the prior approaches. [17] proposes multi-level structured self-attention mechanism. [26] proposes a joint model that adopts a KG-based attention mechanism and achieves significant improvement than [41] on DS-RE.

The attention mechanism in deep NN-based models has achieved significant progress on DS-RE. However, the brevity of input sentences could still negatively affect the performance. To address this issue, we assume that the reasoning paths between target entity pairs over a KG could be applied as BK to fill the "gaps" of input sentences and thus promote the efficiency of DS-RE. [58] uses some inference pattern learned from

UMLS for eliminating potentially related entity pairs from negative training data for DS-RE. [29] applies entity descriptions generated form Freebase and Wikipedia as BK, [40] utilizes multilingual text as BK and [69] uses relation alias information (e.g., $founded$ and $co\text{-}founded$ are aliases for the relation $founderOfCompany$) as BK for DS-RE. However, none of these existing approaches mentioned above comprehensively consider multiple sentences containing entity pairs and multiple reasoning paths between them for DS-RE.

7.3 Proposed Model

As discussed before, the sentences containing the entity pairs of interest tend to omit the BK that the authors assume is known to the readers. However, the omitted BK would be extremely important for a machine to identify the relation between the entity pairs. To fill the "gaps" and improve the efficacy of DS-RE, we assume that the reasoning paths between the entity pairs over a KG could be utilized as BK to compensate for the brevity of the sentences. Motivated by this issue, we propose the DS-RE model that integrates both reasoning paths and sentences.

7.3.1 Architecture

The proposed model consists of three parts: KG Encoding Part, Sentence Encoding Part and Path Encoding Part, as shown in Figure 7.2. The KG Encoding Part and Sentence Encoding Part are identical to the base model introduced in Chapter 6, except that the final input to the relation classification layer. The Path Encoding Part takes as input a set of reasoning paths, $P_r = \{p_1, ..., p_m\}$, between two entities of interest

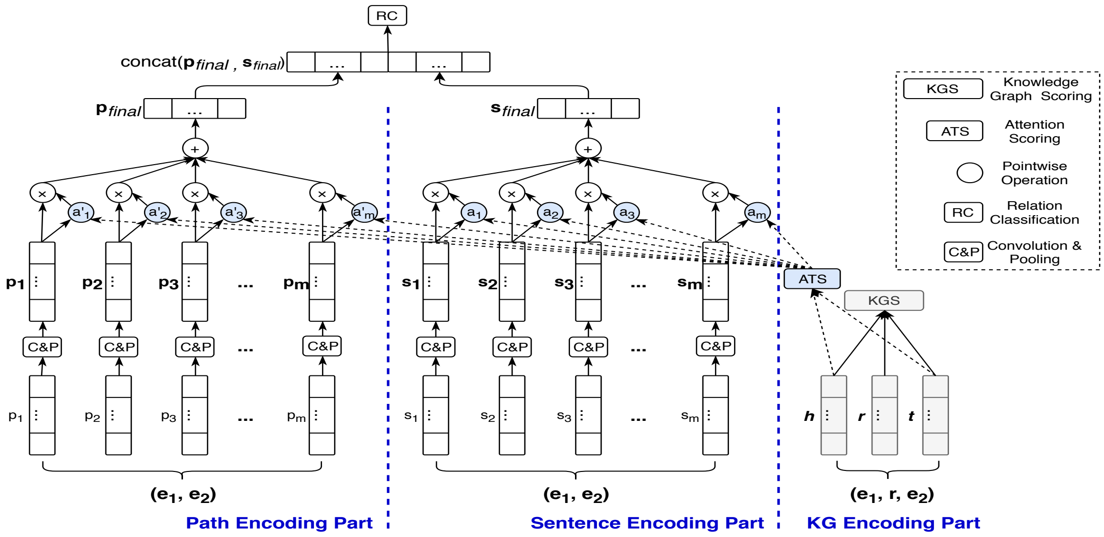


Figure 7.2: Overview of the proposed model.

(e_1, e_2), and encodes them into the final representation of KG based reasoning paths, $\mathbf{kp}_{final}$. Specifically, let $p = \{e_1, r_1, e_{r_1}, r_2, e_{r_2}, ..., r_i, e_{r_i} ..., e_2\}$ denote a path between (e_1, e_2). To express the semantic meaning of a relation in a path, we represent r_i by its component words, rather than treat it as an unit. Therefore, a path will be represented as $p = \{e_1, w_1^{r_1}, w_2^{r_1}, ..., e_{r_1}, , w_1^{r_2}, w_2^{r_2}, ..., e_{r_2}, ..., e_2\}$, where $w_2^{r_1}$ denotes the second word of r_1 (e.g., *treat* in *may_treat* relation).

Since a path is represented as a sequence of words, or a special sentence, we apply the similar CNN model used in the Sentence Encoding Part to encode the path into vector representation $\mathbf{p}_i$. The Path Encoding Part and Sentence Encoding Part share the word embedding projection matrix $\mathbf{W}_{emb}^{w}$, and word position projection matrix $\mathbf{W}_{emb}^{wp}$ in Equation 6.5 except the convolutional kernal $\mathbf{W}$ and its corresponding bias vector $\mathbf{b}$ in Equation 6.6. To utilize evidence from all the paths between target entity pair, we also adopt the KG-based attention mechanism applied in Sentence Encoding Part to calculate the final representation of paths $\mathbf{kp}_{final}$. We calculate $\mathbf{kp}_{final}$ via

Equation 8.1, where $\mathbf{W}_s$ is the weight matrix, $\mathbf{b}_s$ is the bias vector, a'_i is the weight for $\mathbf{p}_i$, which is the distributed representation for the i-th path in P_r.

$$\mathbf{kp}_{final} = \sum_{i=1}^{m} a'_i \mathbf{p}_i, \tag{7.1}$$

$$a'_i = \frac{\exp(\langle \mathbf{r}_{ht}, \mathbf{x'}_i \rangle)}{\sum_{k=1}^{m} \exp(\langle \mathbf{r}_{ht}, \mathbf{x'}_k \rangle)},$$
$$\mathbf{x'}_i = \tanh(\mathbf{W}_s \mathbf{p}_i + \mathbf{b}_s)$$

Finally, we concatenate the resulting representation $\mathbf{s}_{final}$ and $\mathbf{kp}_{final}$ for S_r (the set of input sentences) and P_r (the set of reasoning paths) respectively as the input to the relation classification layer. The conditional probability $P(r|S_r, P_r, \theta_S, \theta_P)$ is formulated via Equation 8.2 and Equation 8.3, where, θ_P is the parameters in Path Encoding Part, $\mathbf{M}$ is the representation matrix of relations, $\mathbf{d}$ is a bias vector, $\mathbf{o}$ is the output vector containing the prediction probabilities of all target relations for both input sentences set S_r and input paths set P_r. n_r is the total number of relations.

$$P(r|S_r, P_r, \theta_S, \theta_P) = \frac{\exp(\mathbf{o}_r)}{\sum_{c=1}^{n_r} \exp(\mathbf{o}_c)} \tag{7.2}$$

$$\mathbf{o} = \mathbf{M}[\mathbf{s}_{final}; \mathbf{kp}_{final}] + \mathbf{d} \tag{7.3}$$

Similar to the base model, we define the optimization function as the log-likelihood of the objective function in Equation 8.4.

$$P(G, D|\theta) = P(G|\theta_{\mathcal{E}}, \theta_{\mathcal{R}}) + P(D|\theta_S, \theta_P) \tag{7.4}$$

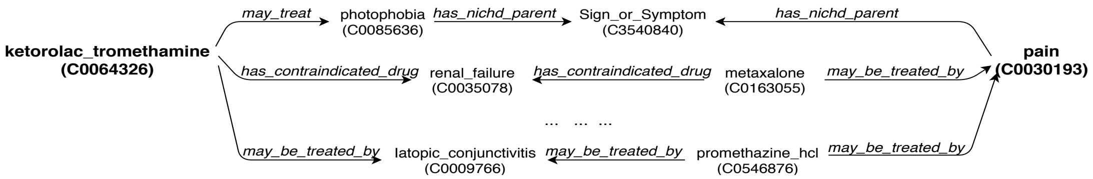


Figure 7.3: Multiple reasoning paths between *ketorolac_tromethamine* and *pain*.

7.3.2 Reasoning Paths Generation

Let (e_1, e_2) be an entity pair of interest. The set of reasoning paths P_r is obtained by computing all shortest paths in a KG starting from e_1 till e_2. For simulating the situation where the direct relation between a target entity pair is unavailable in a sparse KG, we remove the triplet that directly connect the target entity pair of interest from the KG. Each reasoning path, thus, is at least a two-hop path, namely $p = \{e_1, r_1, e_{r_1}, r_2, e_2\}$. However, if the shortest path is not found due to the sparsity of KG, we will use a padding path to represent the missing path $p = \{r_{padding}\}$. Figure 8.3 shows the generated paths between *ketorolac_tromethamine* and *pain*.

7.4 Experiments

Our experiments aim to demonstrate the effectiveness of the proposed model, which is discussed in Section 8.3, for DS-RE from biomedical datasets.

7.4.1 Data

The biomedical datasets used for evaluation consist of knowledge graph, textual data and reasoning path, which will be detailed as follows.

Knowledge Graph. We choose the UMLS as the KG. UMLS is a large biomedical

#Entity	#Relation	#Train (triplet)	#Test (triplet)
16,049	295	34,378	12,502

Table 7.1: Statistics of KG in this work.

knowledge base developed at the U.S. National Library of Medicine. UMLS contains millions of biomedical concepts and relations between them. We follow [73], and only collect the fact triplet with RO relation category (RO stands for “has Relationship Other than synonymous, narrower, or broader”), which covers the interesting relations such as *may_treat* and *my_prevent*. From the UMLS 2018 release, we extract about 50 thousand such RO fact triplets (i.e., (e_1, r, e_2)) under the restriction that their entity pairs (i.e., e_1 and e_2) should coexist within a sentence in Medline corpus. They are then randomly divided into training and testing sets for KGC. Following [74], we keep high entity overlap between training and testing set, but zero fact triplet overlap. The statistics of the extracted KG is shown in Table 7.1.

Textual Data. Medline corpus is a collection of bimedical abstracts maintained by the National Library of Medicine. From the Medline corpus, by applying the UMLS entity recognizer, QuickUMLS [64], we extract $682,093$ sentences that contain UMLS entity pairs as our textual data, in which $485,498$ sentences are for training and $196,595$ sentences for testing. For identifying the NA relation, besides the “related” sentences, we also extract the “unrelated” sentences based on a closed world assumption: pairs of entities not listed in the KG are regarded to have NA relation and sentences containing them considered to be the “unrelated” sentences. By this way, we extract $1,394,025$ “unrelated” sentences for the training data, and $598,154$ “unrelated” sentences for the testing data. Table 7.2 presents some sample sentences in the training data.

Reasoning Path. Following the Section 8.3.1, we extract $197,396$ paths for not NA

Fact Triplet	Textual Data
(insulin, gene_product_plays_role_in_biological_process, energy_expenditure)	s_1 : *These results indicate that hyperglucagonemia during* $\underline{\textit{insulin}}_{e_1}$ *deficiency results in an increase in* $\underline{\textit{energy\_expenditure}}_{e_2}$*, which may contribute to the catabolic_state in many conditions.* s_2 : *It was hypothesized that the waxy maize treatment would result in a blunted and more sustained glucose and* $\underline{\textit{insulin}}_{e_1}$ *response, as well as* $\underline{\textit{energy\_expenditure}}_{e_2}$ *and appetitive responses.* s_3 : ...
(IRI, NA, insulin)	s_1 : *Plasma insulin immunoreactivity (*$\underline{\textit{IRI}}_{e_1}$*) results from high molecular weight substances with insulin immunoreactivity (HWIRI), proinsulin (PI) and* $\underline{\textit{insulin}}_{e_2}$ *(I).* s_2 : *The beads method demonstrated high* $\underline{\textit{IRI}}_{e_1}$ *values in both* $\underline{\textit{insulin}}_{e_2}$ *fractions and the fractions containing serum_proteins bigger than 40,000 molecular weight.* s_3 : ...

Table 7.2: Examples of textual data extracted from Medline corpus.

triplets (139, 224 / 58, 172 for training / testing) and 679, 408 for NA triplets (474, 263 / 205, 145 for training / testing), under the restriction that each entity in a path should be observed in Medline corpus.

7.4.2 Parameter Settings

We base our work on [26] and its implementation available at `https://github.com/thunlp/JointNRE`, and thus adopt identical optimization process. We use the default settings of parameters [2] provided by the base model. Since we address the DS-RE in biomedical domain, we use the Medline corpus to train the domain specific word embedding projection matrix $\mathbf{W}^{w}_{emb}$ in Equation 6.5.

7.4.3 Result and Discussion

We investigate the effectiveness of our proposed model with respect to enhancing the DS-RE from biomedical datasets. We follow [48, 74, 41, 26] and conduct the held-out evaluation, in which the model for DS-RE is evaluated by comparing the fact triplets

[2] As a preliminary study, we only adopt the default hyperparameters, but we will tune them for our task in the furture.

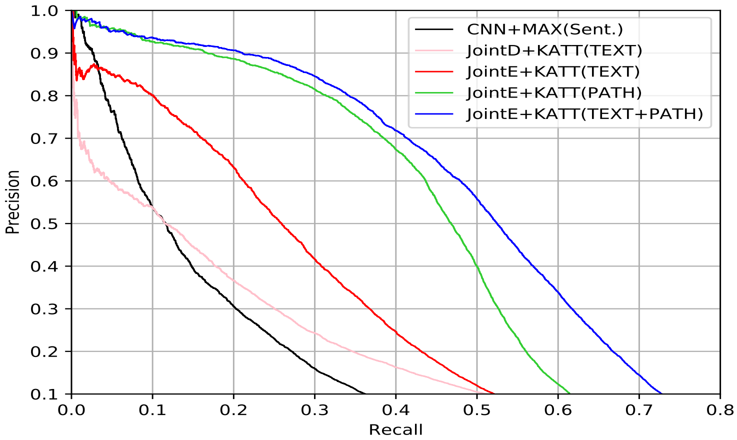


Figure 7.4: Precision-Recall curves.

identified from textual data (i.e., the set of sentences containing the target entity pairs) with those in KG. Following the evaluation of previous works, we draw Precision-Recall curves and report the micro average precision (AP) score, which is a measure of the area under the Precision-Recall curve (higher is better), as well as Precision@N (P@N) metrics, which gives the percentage of correct triplets among top N ranked candidates.

Precision-Recall Curves. The Precision-Recall (PR) curves are shown in Figure 7.4, where "CNN+MAX" represents that the DS-RE model uses max-polling over the vector of sentences as $\mathbf{s}_{final}$ in Equation 6.7. "JointE+KATT" (or "JointD+KATT") represents that the DS-RE model applies Prob-TransE (or Prob-TransD) as its KG Encoding Part for attention calculation. "(TEXT)" indicates that the model only takes the textual data as input (i.e., the set of sentences containing target entity pairs). "(PATH)" indicates the DS-RE model only takes the reasoning paths between entity pairs as its input. "(TEXT+PATH)" indicates the DS-RE model takes both the textual data and reasoning paths as its input. The results show that:

(1) The proposed model (i.e., "JointE+KATT(PATH+TEXT)") significantly outperform the base model (i.e., "JointE+KATT(TEXT)"), proving that reasoning paths are useful BK for biomedical DS-RE. This result inspires us to explore other reasoning strategy such as by reasoning across multiple documents. (2) "JointE+KATT(PATH+TEXT)" achieves better overall performance than "JointE+KATT(PATH)", demonstrating the mutual complementary relationship between the sentences containing entity pairs and the reasoning paths between them. Specifically, on the one hand, as discussed in Section 8.1, reasoning paths could provide BK for interpreting the implicitly expressed relation in sentences. On the other hand, due to the sparsity of KG, it is by no means certain that all entity pairs are fully connected by plausible reasoning paths in the KG. In that case, the sentences could provide the informative evidence to identify the relation between them.

AP and P@N Evaluation. The results in terms of P@1k, P@2k, P@3k, P@4k, P@5k, the mean of them and AP are shown in Table 7.3. From the table, we have similar observation to the PR curves: (1) The proposed model (i.e., "JointE+KATT(TEXT+PATH)") significantly outperforms the base model for all measures. (2) "JointE+KATT(TEXT+PATH)" outperforms "JointE+KATT(PATH)" in most of the metrics.

Model	P@1k	P@2k	P@3k	P@4k	P@5k	Mean	AP
CNN+MAX(Sent.)	0.863	0.763	0.700	0.658	0.627	0.722	0.165
JointD+KATT(TEXT)	0.628	0.614	0.552	0.495	0.446	0.547	0.186
JointE+KATT(TEXT)	0.835	0.759	0.692	0.629	0.564	0.696	0.272
JointE+KATT(PATH)	**0.945**	0.911	0.881	0.842	0.796	0.875	0.432
JointE+KATT(TEXT+PATH)	0.941	**0.922**	**0.897**	**0.865**	**0.818**	**0.889**	**0.496**

Table 7.3: P@N and AP for different DS-RE models, where k=1000.

Case Study. Table 7.4 shows the comparison of the attention distribution between "JointE+KATT(TEXT)" (Base) and "JointE+KATT(TEXT+PATH)" (Proposed). The first and second columns represent the attention distribution (the highest and the lowest)

Base	Proposed	Sentences for (**Mitomycin_C (MCC)**, may_treat, **stomach/gastric_tumor**)
High	Low	The additive effect in the combination of TNF and Mitomycin_C was observed against two **Mitomycin_C** resistant **gastric_tumors**.
Low	High	One-quarter or one-half maximum tolerated doses (MTDs) of 5-FU or **MMC** resulted in a significant reduction of **stomach_tumor** growth, ...

Table 7.4: Comparison of attention between base model and proposed model, where High (or Low) represents the highest (or lowest) attention.

Attention	Paths for (**etoposide**, may_treat, **lung_tumor**)
Low	**etoposide** $\underleftarrow{has\_contraindicated\_drug}$ drug_allergy $\underrightarrow{has\_contraindicated\_drug}$ S-Liposomal Doxorubicin $\underrightarrow{may\_treat}$ **lung_tumor**
High	**etoposide** $\underrightarrow{may\_be\_treated\_by}$ Histiocytoses $\underleftarrow{may\_be\_treated\_by}$ Vinblastine $\underrightarrow{may\_treat}$ **lung_tumor**

Table 7.5: Some examples of attention distribution over reasoning paths from "JointE+KATT(TEXT+PATH)".

over input sentences. From the Table 7.4, we can see that the proposed model that incorporates reasoning paths is more capable of selecting informative sentences than the base model, because it "focuses" on the second sentence that explicitly describes the *may_treat* relation via the word "reduction", in contrast, the base model "ignores" such informative sentence. Table 7.5 shows the attention allocated by our proposed model for given reasoning paths. The first path generally means if two chemicals should not be used in the case of (or contraindicated with) drug_allergy, they will treat lung_tumor. In contrast, the second path generally means if two chemicals treat Histiocytoses (an excessive number of cells), they will also treat lung_tumor. Apparently the second one that our proposed model focused on is more plausible. This indicates that our proposed model has the capacity of identifying the plausible reasoning path.

7.5 Conclusion and Future Work

In this work, we tackle the task of DS-RE from biomedical datasets. However, the biomedical DS-RE could be negatively affected by the brevity of text. Specifically, authors always omit the BK that would be important for a machine to identify relationships between entities. To address this issue, in this work, we assume that the reasoning paths over a KG could be utilized as the BK to fill the "gaps" in text and thus facilitate DS-RE. Experimental results prove the effectiveness of the combination, because our proposed model achieves significant and consistent improvements as compared with a state-of-the-art DS-RE model. Although the reasoning paths over KG are useful for DS-RE, the sparsity of KG would hinder their effectiveness. Therefore, in the future, beside the reasoning paths over KG, we will also utilize the reasoning paths across multiple documents for our task. For instance, reasoning across Document1 and Document2, shown below, would facilitate the relation identification between "Aspirin" and "inflammation".

Document1: "***Aspirin*** *and other* ***nonsteroidal anti-inflammatory drugs*** *(NSAID) show ...*"

Document2: "***Nonsteroidal anti-inflammatory drugs*** *reduce* ***inflammation*** *by ...*"

Chapter 8

Reasoning across Multiple Documents for Distantly Supervised Biomedical Knowledge Acquisition

8.1 Introduction

In RE, one obstacle that is encountered when building a RE system is the generation of training instances. For coping with this difficulty, [48] proposes Distant Supervision (DS) to automatically generate training samples via leveraging the alignment between KGs and texts. They assume that if two entities are connected by a relation in a KG, then all sentences that contain those entity pairs will express the relation. For expanding the scope of DS-RE, [54] proposes Distant Supervision for Cross-sentence Relation EXtraction (DISCREX for short) to extract relations from adjacent sentences within single documents.

(36) ***Aspirin**$_{e_1}$ and other **nonsteroidal anti-inflammatory drugs**$_{e_2}$ (NSAID) show indisputable promise as cancer chemoprevention agents. (PMID [1]:21803981)*

(37) ***Nonsteroidal anti-inflammatory drugs**$_{e_2}$ reduce **inflammation**$_{e_3}$ by inhibiting the action of **Cyclooxygenase (COX) enzymes**$_{e_4}$, ... (PMID:24618207)*

(38) *Prostaglandins (PG) formed by **cyclooxygenase (COX) enzymes**$_{e_4}$ are important mediators of inflammation in **rheumatoid arthritis**$_{e_5}$. (PMID:10701683)*

Although DS-RE achieves significant progress, DS-RE has so far been limited to single documents, thus leaving the rich relations crossing the document boundary untapped. For instance, by reasoning over the two documents: Example 36 and Example 37, we could acquire the fact triplet ($aspirin$, may_treat, $inflammation$), because of the reasoning path over entities illustrated in Figure 8.1. The fact triplet, however, can not be directly extracted from each Example alone. Similarly, based on the three documents: Example 36, Example 37 and Example 38, we could infer that ($aspirin$, may_treat, $rheumatoid\_arthritis$), because of the reasoning path illustrated in Figure 8.2, but which is not originally conveyed by each Example alone.

To address the issue, in this chapter, we assume that cross-document reasoning paths that connecting those target entity pairs could be used for DS-RE. We define the cross-document reasoning paths as the multi-hop paths over a cross-document-level graph representation, as shown in Figure 8.1 and Figure 8.2, where each node is the entity of interest and each edge represents the middle context between an entity pair within a sentence.

[1] A PMID is the unique identifier number of each article in PubMed (`http://www.ncbi.nlm.nih.gov/pubmed`).

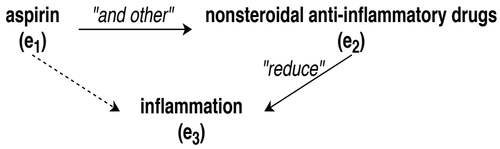


Figure 8.1: An example of reasoning path across 2 documents.

(39) *One hundred nineteen adults with active definite or classical* ***rheumatoid arthritis***$_{e_5}$ *were studied in a multicenter double-blind crossover study of naproxen (500 mg/day) and* ***aspirin***$_{e_1}$ *(3.6 Gm/day). (PMID:1092727)*

In addition, authors always omit the Background Knowledge (BK) that they assume is well known by reader, but would be essential for a machine to identify relationship. For instance, Example 39 omits the background connection between *rheumatoid arthritis* and *aspirin* and implicitly conveys that the latter *may_treat* the former. Human readers could easily make this inference based on their BK about the mechanism between them. However, for a machine, it would be extremely difficult to identify the relationship just from the given sentence without the important BK.

To address the issue of textual brevity, in this chapter, we also assume that the cross-document reasoning paths between an entity pair mentioned above could be applied as the BK to fill the "gaps" and thereby improve the performance of DS-RE. For instance, the reasoning paths in Figure 8.1 could be seen as the BK between *inflammation* and *aspirin*. To this end, we propose a DS-RE model that not only encodes the sentences containing target entity pairs, but also the cross-document reasoning paths between them.

We conduct evaluation on biomedical datasets in which KG is collected from UMLS and textual data is extracted from Medline corpus. The experimental results prove the effectiveness of the cross-document reasoning paths for improving DS-RE from

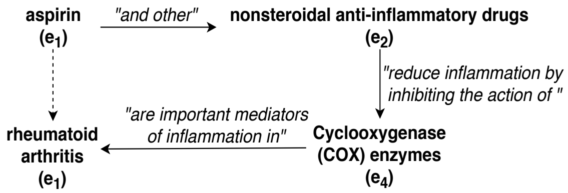

aspirin
(e1)
"and other"
nonsteroidal anti-inflammatory drugs
(e2)
"reduce inflammation by inhibiting the action of "
"are important mediators of inflammation in"
rheumatoid arthritis
(e1)
Cyclooxygenase (COX) enzymes
(e4)

Figure 8.2: An example of reasoning path across 3 documents.

biomedical datasets.

8.2 Related Work

RE is a fundamental task in the NLP community. In recent years, Neural Network (NN)-based models have been the dominant approaches for non-scientific RE, which include Convolutional Neural Network (CNN)-based frameworks [79, 77, 61] Recurrent Neural Network (RNN)-based frameworks [81, 49, 84]. NN-based approaches are also used in scientific RE. For instance, [22] utilizes a CNN-based model for identifying *chemical-disease* relations from Medline corpus. [25] proposes an LSTM-based model for identifying *causal precedence* relationship between two event mentions in biomedical papers. [1] applies [49]'s model for scientific RE.

Although remarkably good performances are achieved by the models mentioned above, they still train and extract relations on sentence-level and thus need a large amount of annotation data, which is expensive and time-consuming. To address this issue, distant supervision is proposed by [48]. To alleviate the noisy data from the distant supervision, many studies model DS-RE as a Multiple Instance Learning (MIL) problem [55, 28, 80], in which all sentences containing a target entity pair (e.g.,*ketorolac_tromethamine* and *pain*) are seen as a bag to be classified. To make full use of all the sentences in the bag, rather than just the most informative one in the bag, researchers apply attention mechanism in deep NN-based models for DS-RE. [41] proposes a relation vector based attention mechanism to extract feature from the entire bag and outperforms the prior approaches. [17] proposes multi-level structured self-attention mechanism. [26] proposes a joint model that adopts a KG-based attention mechanism and achieves significant improvement than [41] on DS-RE.

The attention mechanism in deep NN-based models has achieved significant progress on DS-RE. However, the brevity of input sentences could still negatively affect the performance. To address this issue, we assume that the cross-document reasoning paths between target entity pairs could be applied as BK to fill the “gaps” of input sentences and thus promote the efficiency of DS-RE. [29] applies entity descriptions generated form Freebase and Wikipedia as BK, [40] utilizes multilingual text as BK and [69] uses relation alias information (e.g., $founded$ and $co\text{-}founded$ are aliases for the relation $founderOfCompany$) as BK for DS-RE. However, none of these existing approaches mentioned above comprehensively consider the the sentences containing entity pairs the reasoning paths for DS-RE, especially in the biomedical domain. For expanding the scope of DS-RE, [54] proposes Distant Supervision for Cross-sentence Relation EXtraction (DISCREX for short) to extract relations from adjacent sentences within single documents, but net from multiple documents. [13] applies reasoning chains over KG and textual corpus to infer missing relation in KG, but does not use the reasoning chains as BK for DS-RE.

8.3 Proposed Model

8.3.1 Reasoning Paths Generation

Entity Recognition. In this step, we use the UMLS entity recognizer [2] (called QuickUMLS) proposed by [64] to identify UMLS concepts in Medline corpus. The model annotates the detected entities by their corresponding UMLS Concept Unique Identifier (CUI) as shown in Example 40 the first row.

(40) *One reason that the association*$_{C0004083}$ *between myocardial_infarction*$_{C0027051}$ *and pneumonia*$_{C0032285}$ *was not previously recognized is that aspirin*$_{C0004057}$ *was widely used*$_{C1273517}$ *in the past when people had acute febrile*$_{C0015967}$ *conditions*$_{C0012634}$ ...

Paths Generation. Let (e_1, e_2) be an entity pair of interest. The set of reasoning paths P_r is obtained by computing all shortest paths starting from e_1 till e_2 in the cross-document graph representation as shown in Figure 8.3. For simulating the situation of cross-document reasoning, we requires each triplet in a path, such as $(aspirin, like, nonsteroidalanti - inflammatorydrugs)$, should be extracted from different documents. Since the most informative part of text to classify the relation type generally exists between and including target entity pair [37, 78]. Additionally, Open Information Extraction (OIE) systems perform significantly worse on scientific text than encyclopedic text [21]. We simply extract the middle context between entity pairs as their relation representation. If there are multiple relation expressions between an entity pair as shown in Table 8.1, we randomly select the one with smallest text span

[2] It is available at `https://github.com/Georgetown-IR-Lab/QuickUMLS`

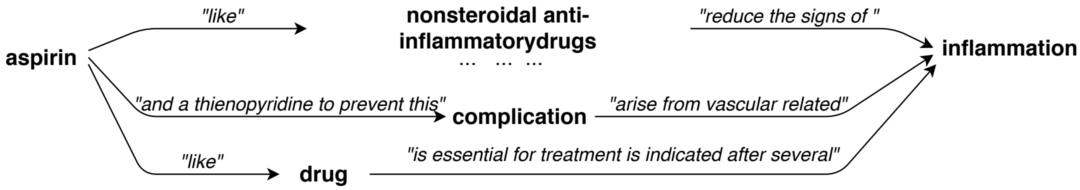

"like"
nonsteroidal anti-
inflammatorydrugs
... ...
"reduce the signs of "
aspirin
inflammation
"and a thienopyridine to prevent this"
complication
"arise from vascular related"
"like"
drug
"is essential for treatment is indicated after several"

Figure 8.3: Multiple reasoning paths between *aspirin* and *inflammation*.

e_1	r	e_2
aspirin	*"and several types of"* ***"like"*** *"and commonly used"* *"to enhance synthesis suggests that"* ...	nonsteroidal anti-inflammatory drugs

Table 8.1: An example of multiple relation expressions.

as its relation [3].

8.3.2 Architecture

The proposed model consists of three parts: KG Encoding Part, Sentence Encoding Part and Path Encoding Part, as shown in Figure 7.2 in Chapter 7.3.1. The KG Encoding Part and Sentence Encoding Part are identical to the base model introduced in Chapter 6, except that the final input to the relation classification layer. The Path Encoding Part takes as input a set of cross-document reasoning paths, $P_r = \{p_1, ..., p_m\}$, between two entities of interest (e_1, e_2), and encodes them into the final representation of cross-document reasoning paths, $\mathbf{cp}_{final}$. Specifically, let $p = \{e_1, r_1, e_{r_1}, r_2, e_{r_2}, ..., r_i, e_{r_i} ..., e_2\}$ denote a path between (e_1, e_2). To express the semantic meaning of a relation in a path, we represent r_i by its component words, rather than treat it as a unit. Therefore,

[3] In the future, we will implement more effective method to select the representative relation expression.

a path will be represented as $p = \{e_1, w_1^{r_1}, w_2^{r_1}, ..., e_{r_1},, w_1^{r_2}, w_2^{r_2}, ..., e_{r_2}, ..., e_2\}$, where $w_2^{r_1}$ denotes the second word of r_1 (e.g., "*inhibit*" in "*to inhibit fever*" relation).

Since a path is represented as a sequence of words, or a special sentence, we apply the similar CNN model used in the Sentence Encoding Part to encode the path into vector representation $\mathbf{p}_i$. The Path Encoding Part and Sentence Encoding Part share the word embedding projection matrix $\mathbf{W}_{emb}^{w}$, and word position projection matrix $\mathbf{W}_{emb}^{wp}$ in Equation 6.5 except the convolutional kernal $\mathbf{W}$ and its corresponding bias vector $\mathbf{b}$ in Equation 6.6. To utilize evidence from all the paths between target entity pair, we also adopt the KG-based attention mechanism applied in Sentence Encoding Part to calculate the final representation of paths $\mathbf{cp}_{final}$. We calculate $\mathbf{cp}_{final}$ via Equation 8.1, where $\mathbf{W}_s$ is the weight matrix, $\mathbf{b}_s$ is the bias vector, a'_i is the weight for $\mathbf{p}_i$, which is the distributed representation for the i-th path in P_r.

$$\mathbf{cp}_{final} = \sum_{i=1}^{m} a'_i \mathbf{p}_i, \tag{8.1}$$

$$a'_i = \frac{\exp(\langle \mathbf{r}_{ht}, \mathbf{x'}_i \rangle)}{\sum_{k=1}^{m} \exp(\langle \mathbf{r}_{ht}, \mathbf{x'}_k \rangle)},$$

$$\mathbf{x'}_i = \tanh(\mathbf{W}_s \mathbf{p}_i + \mathbf{b}_s)$$

Finally, we concatenate the resulting representation $\mathbf{s}_{final}$ and $\mathbf{cp}_{final}$ for S_r (the set of input sentences) and P_r (the set of reasoning paths) respectively as the input to the relation classification layer. The conditional probability $P(r|S_r, P_r, \theta_S, \theta_P)$ is formulated via Equation 8.2 and Equation 8.3, where, θ_P is the parameters in Path Encoding Part, $\mathbf{M}$ is the representation matrix of relations, $\mathbf{d}$ is a bias vector, $\mathbf{o}$ is the output vector containing the prediction probabilities of all target relations for both input sentences set S_r and input paths set P_r. n_r is the total number of relations.

$$P(r|S_r, P_r, \theta_S, \theta_P) = \frac{\exp(\mathbf{o}_r)}{\sum_{c=1}^{n_r} \exp(\mathbf{o}_c)} \tag{8.2}$$

$$\mathbf{o} = \mathbf{M}[\mathbf{s}_{final}; \mathbf{cp}_{final}] + \mathbf{d} \tag{8.3}$$

Similar to the base model, we define the optimization function as the log-likelihood of the objective function in Equation 8.4.

$$P(G, D|\theta) = P(G|\theta_{\mathcal{E}}, \theta_{\mathcal{R}}) + P(D|\theta_S, \theta_P) \tag{8.4}$$

8.4 Experiments

Our experiments aim to demonstrate the effectiveness of the proposed model, which is discussed in Section 8.3, for DS-RE from biomedical datasets.

8.4.1 Data and Parameter Settings

We use the identical dataset that is introduced in Chapter 7 for evaluation. We base our work on [26] and its implementation available at `https://github.com/thunlp/JointNRE`, and thus adopt identical optimization process. We use the default settings of parameters [4] provided by the base model. Since we address the DS-RE in biomedical domain, we use the Medline corpus to train the domain specific word embedding projection matrix $\mathbf{W}^{w}_{emb}$.

[4] As a preliminary study, we only adopt the default hyperparameters, but we will tune them for our task in the furture.

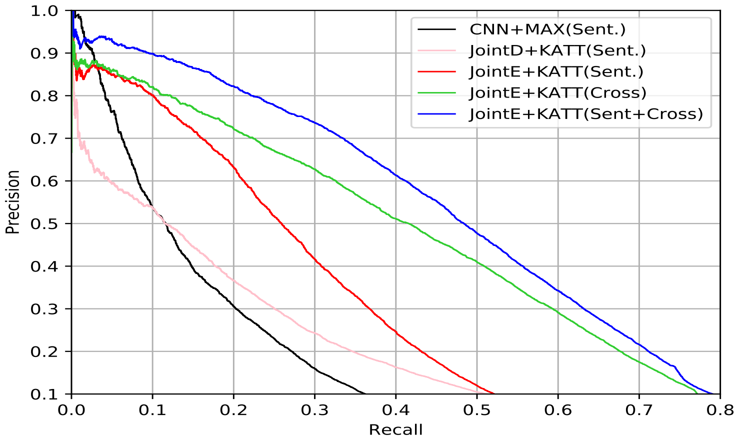


Figure 8.4: Precision-Recall curves for different DS-RE models.

8.4.2 Result and Discussion

We investigate the effectiveness of our proposed model with respect to enhancing the DS-RE from biomedical datasets. We follow [48, 74, 41, 26] and conduct the held-out evaluation, in which the model for DS-RE is evaluated by comparing the fact triplets identified from textual data (i.e. the set of sentences containing the target entity pairs) with those in KG. Following the evaluation of previous works, we draw Precision-Recall curves and report the micro average precision (AP) score, which is a measure of the area under the Precision-Recall curve (higher is better), as well as Precision@N (P@N) metrics, which gives the percentage of correct triplets among top N ranked candidates.

Precision-Recall Curves. The Precision-Recall (PR) curves are shown in Figure 8.4, where “(JointE)” (or “(JointD)”) represents that the DS-RE model applies Prob-TransE (or Prob-TransD) as its KG Encoding Part for attention calculation. “Sent.” indicates that the model only takes the textual data as input (i.e., the set of sentences containing target entity pairs), which is the base model. “Cross” indicates the DS-RE model takes the reasoning paths across different documents as its input. “Sent.+Cross” indicates the DS-RE model takes both the textual data and reasoning paths across multiple documents as its input, which is encoded by the proposed model illustrated in Figure 7.2. The results show that:

(1) DS-RE from cross-document reasoning paths (i.e., “Cross”) achieves better performance than the base model (i.e., “Sent.”), proving that it is feasible to extract knowledge from the reasoning paths across multiple documents, especially when a target entity pair does not co-occur in a single sentence. (2) DS-RE based on the combination of “Sent.” (i.e., the sentences containing the target entity pairs) and “Cross.” (i.e., multi-hop reasoning paths across multiple documents) significantly outperforms the base model (i.e., “Sent.”), proving that reasoning paths are useful BK for biomedical DS-RE. (3) “Sent.+Cross” achieves better performance than “Cross”, demonstrating the mutual complementary relationship between the sentences containing entity pairs and the reasoning paths across multiple documents. Specifically, on the one hand, as discussed in Section 8.1, reasoning paths could provide BK for interpreting the implicitly expressed relation between entity pairs in “Sent.”. On the other hand, the “Sent.” could positively affect the “Cross” for DS-RE.

AP and P@N Evaluation. The results in terms of P@1K, P@2K, P@3K, P@4K, P@5K, the mean of them and AP are shown in Table 8.2. From the table, we have similar observation to the PR curves: (1) The proposed model (i.e., “Sent.+Cross”) significantly outperforms the base model (i.e., “Sent.”) for all measures. (2) “Sent.” and “Cross” could compensate each other for biomedical DS-RE.

Case Study. Table 8.3 shows the attention allocated by our proposed model for given reasoning paths. The first path does not clearly interpret the relationship between

Model	P@1k	P@2k	P@3k	P@4k	P@5k	Mean	AP
CNN+MAX(Sent.)	0.863	0.763	0.700	0.658	0.627	0.722	0.165
JointD+KATT(Sent.)	0.628	0.614	0.552	0.495	0.446	0.547	0.186
JointE+KATT(Sent.)	0.835	0.759	0.692	0.629	0.564	0.696	0.272
JointE+KATT(Cross)	0.841	0.794	0.746	0.700	0.659	0.748	0.409
JointE+KATT(Sent.+Cross)	**0.912**	**0.874**	**0.823**	**0.779**	**0.740**	**0.825**	**0.470**

Table 8.2: P@N and AP for different DS-RE models, where k=1000.

Attention	Paths for (**naproxen_sodium**, may_treat, **headache**)
0.0002	**naproxen_sodium** $\underrightarrow{in\_two\_randomized\_open}$ label $\underrightarrow{except\_for\_those\_experiencing}$ headache
0.8790	**naproxen_sodium** $\underleftarrow{following\_treatment\_with}$ disease_regression $\underrightarrow{of}$ **headache**

Table 8.3: Some examples of attention distribution over reasoning paths from “JointE+KATT(Sent.+Cross)”.

“naproxen_sodium” and “headache”. In contrast, the second path generally means that “disease_regression” could happen along with the treatment with “naproxen_sodium” and “disease_regression” could also happen in the case of “headache”. Apparently the second one that our proposed model focused on is more plausible the first one. This indicates that our proposed model has the capacity of identifying plausible reasoning path.

8.5 Conclusion

In this work, we tackle the task of DS-RE from biomedical datasets. Existing approaches only focus on extracting knowledge within the scope of single document. This leaves the rich relations crossing the document boundary untapped. Therefore, we hypothesize that reasoning paths across multiple documents could be used as the source of DS-RE. In addition, we also assume that the cross-document reasoning paths could address the issue of brevity. Specifically, We assume that cross-document reasoning paths could be utilized as the BK that authors always omit in the sentences containing the target entity pairs. Experimental results prove the effectiveness of the cross-document reasoning paths for biomedical DS-RE, because our proposed model achieves significant and consistent improvements as compared with a state-of-the-art DS-RE model.

Chapter 9

Combination of Knowledge Graph based Inference and Cross-document Inference for Distantly Supervised Relation Extraction

9.1 Introduction

Chapter 7 and Chapter 8 respectively discuss the effectiveness of Knowledge Graph based Inference (KGI) and Cross-document Inference (CDI) for Distantly Supervised Relation Extraction (DS-RE). In this chapter, we study the efficacy of combing KGI and CDI for scientific DS-RE. We hypothesize that KGI and CDI could compensate each other and their combination could outperform each of them. For instance, the KGI shown in Figure 9.1 is unable to interpret the relationship between entity pair *Re-*

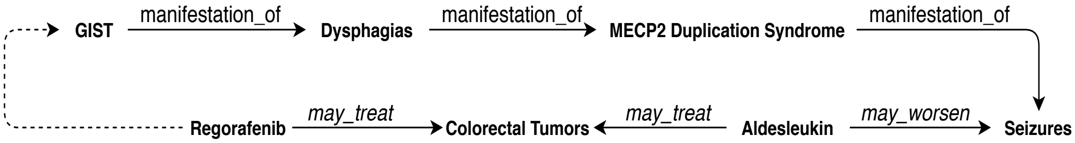


Figure 9.1: An example of KGI.

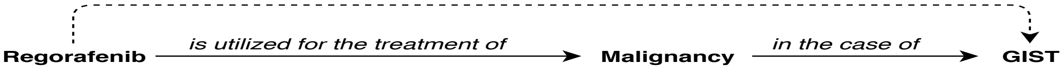


Figure 9.2: An example of CDI.

gorafenib and *GIST* (gastrointestinal stromal tumor). However, the CDI shown in Figure 9.2 is more informative to clarify their relationship as ($Regorafenib$, may_treat, $GIST$). Additionally, in order to investigate the effectiveness of our proposed model, we also compare our model with the one proposed by [13] from the perspective of inference representation, surrounding context and attention mechanism.

9.2 Combination of KGI and CDI

We combine KGI and CDI via tha Equation 9.1, where $\mathbf{kp}_{final}$ means the final representation of KGI discussed in Chapter 7.3.1 and $\mathbf{cp}_{final}$ represents the final representation of CDI discussed in Chapter 8.3.2. **M** is the representation matrix of relations, **d** is a bias vector, **o** is the output vector containing the prediction probabilities of all target relations.

$$\mathbf{o} = \mathbf{M}[\mathbf{s}_{final}; \mathbf{kp}_{final}; \mathbf{cp}_{final}] + \mathbf{d} \tag{9.1}$$

9.3 Evaluation and Result

9.3.1 Evaluation on Scientific Dataset

In order to evaluate the effectiveness of the combination of KGI and CDI, We use the biomedical dataset introduced in Chapter 7 and Chapter 8.

Precision-Recall Curves. The Precision-Recall (PR) curves are shown in Figure 9.3, where “(JointE)” (or “(JointD)”) represents that the DS-RE model applies Prob-TransE (or Prob-TransD) as its KG Encoding Part for attention calculation. “Sent.” indicates that the model only takes the textual data as input (i.e., the set of sentences containing target entity pairs), which is the base model. “Cross” indicates the DS-RE model takes CDI its input. “Path” indicates the DS-RE model takes KGI its input.“Sent.+Path+Cross” indicates the DS-RE model takes the textual data, CDI and KGI as its input. The results show that: incorporating both KGI and CDI (i.e., “Sent.+Path+Cross”) significantly outperforms the base model (i.e., “Sent.”), proving that the combination of KGI and CDI is useful for scientific DS-RE.

AP and P@N Evaluation. The results in terms of P@N, the mean of them and AP are shown in Table 9.1. From the table, we have similar observation as the PR curves: The proposed model (i.e., “Sent.+Path+Cross”) significantly outperforms the base model (i.e., “Sent.”) for all measures.

Model	P@1k	P@2k	P@3k	P@4k	P@5k	Mean	AP
JointD+KATT(Sent.)	0.628	0.614	0.552	0.495	0.446	0.547	0.186
JointE+KATT(Sent.)	0.835	0.759	0.692	0.629	0.564	0.696	0.272
JointE+KATT(Sent.+Cross)	0.912	0.874	0.823	0.779	0.740	0.825	0.470
JointE+KATT(Sent.+Path)	0.941	0.922	0.897	0.865	0.818	0.889	0.496
JointE+KATT(Sent.+Path+Cross)	**0.956**	**0.935**	**0.908**	**0.877**	**0.847**	**0.905**	**0.569**

Table 9.1: P@N and AP on scientific dataset, where k=1000.

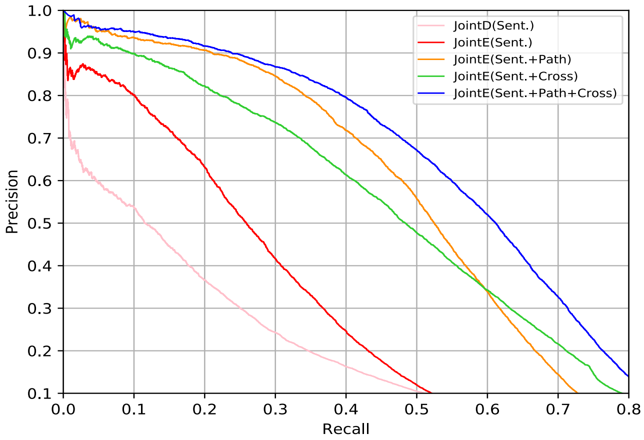


Figure 9.3: Precision-Recall curves on scientific dataset.

9.3.2 The Effect of Unified Graph Representation

To enlarge the space of reasoning path searching, [13] represent Knowledge Base (or Knowledge Graph) and textual documents into a unified graph, as shown in Fig 9.4. To evaluate the effect of the unified graph on our task, we also collect reasoning paths from the unified graph. The results are shown in Table 9.2 and Fig 9.5, where "Unified" means the reasoning paths obtained from the unified graph. Note that 'Unified" alone slightly outperforms "Cross" but the combination of "Path" and "Unified" achieves the best performance. It indicates that, compared with a single method, such as "Path" or "Unified", diversifying the method of reasoning path searching, such as "Path+Cross" or "Path+Unified", could more effectively improve the performance of scientific DS-RE.

Model	**P@1k**	**P@2k**	**P@3k**	**P@4k**	**P@5k**	**Mean**	**AP**
JointE+KATT(Sent.+Path+Cross)	0.956	0.935	0.908	0.877	0.847	0.905	0.569
JointE+KATT(Sent.+Path+Unified)	**0.972**	**0.955**	**0.928**	**0.894**	**0.859**	**0.921**	**0.596**

Table 9.2: P@N and AP on the unified graph representation, where k=1000.

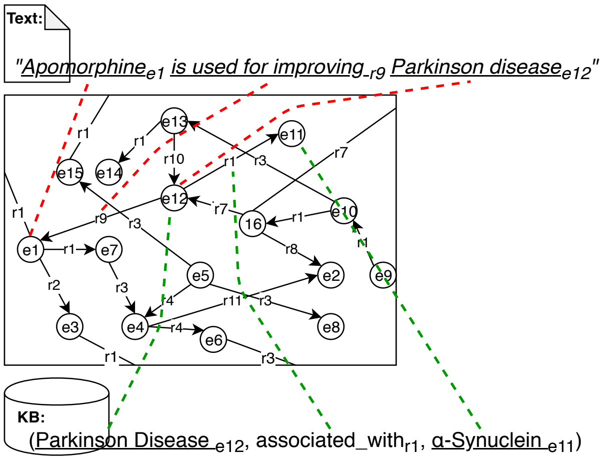

Text:
"Apomorphine e1 is used for improving r9 Parkinson disease e12"
e1
e2
e3
e4
e5
e6
e7
e8
e9
e10
e11
e12
e13
e14
e15
16
r1
r2
r3
r4
r7
r8
r9
r10
r11
KB:
(Parkinson Disease e12, associated_with r1, α-Synuclein e11)

Figure 9.4: Unified graph representation.

9.3.3 The Effect of Textual Representation of Inference

[13] propose a multi-hop inference model that represent the relation between an entity pair as a unit relation rather than a sequence of words. For instance, in Fig 9.2, their model represents the relation between e_2 and e_4 as $\{is\_utilized\_treatment\_of\}$[1], rather than $\{is, utilized, for, the, treatment, of\}$. This model, thus, cannot utilize the semantic feature of each word, such as the word embedding of "*reduce*", and is unable to effectively represent the meaning of the relation and the corresponding inference (or reasoning path). In contrast, our proposed textual representation of inference could more effectively capture the semantic meaning of the inference. Based on this consideration, we compare the performance of the unit representation (e.g., $\{e_2, is\_utilized\_treatment\_of, e_4, ...\}$) proposed by [13] with our textual representation (e.g., $\{e_2, is, utilized, for, the, treatment, of, e_4, ...\}$), which is introduced in Chapter 7 and Chapter 8. In addition, while dealing with the one-hop inference, which is the target sentence containing the target entity pair e_1 and e_2, the model pro-

[1] The following two words after the first entity and two words before the second entity.

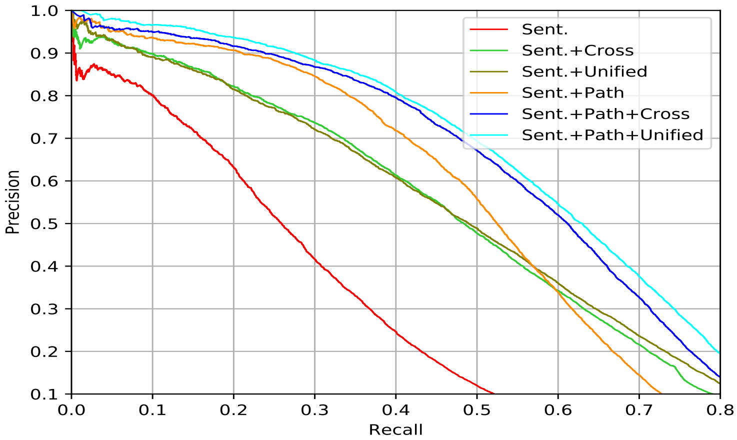
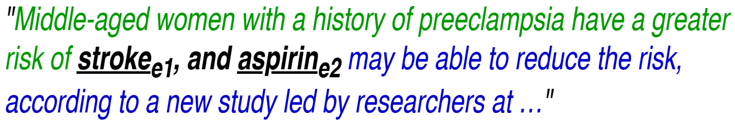


Figure 9.5: Performance of the unified graph representation.



"Middle-aged women with a history of preeclampsia have a greater risk of **stroke**$_{e1}$**, and** **aspirin**$_{e2}$ may be able to reduce the risk, according to a new study led by researchers at ..."

Figure 9.6: An example of middle context and surrounding context.

posed by [13] only considers the middle context and ignores the surrounding context as shown in Fig 9.6, where the former is denoted in black and the latter in green and blue. It can be observed that, only depending on the middle context would lose the useful information described in the surrounding context for relation identification. For instance, in Fig 9.6, middle context presents the relationship between the target entity pair $stroke$ and $aspirin$ with an ambiguous syntactic pattern "*..., and ...*". In contrast, the surrounding context clearly describes that the target entity $aspirin$ could inhibit the "*risk*" of another target entity $stroke$.

The performance is represented in Table 9.3 and Fig. 9.7, where "Unit" (or "Textual") means unit (or textual) representation, "(Mid.)" (or "(Sur.)") means the middle context (or surrounding context). It can be observed that our proposed textual representation

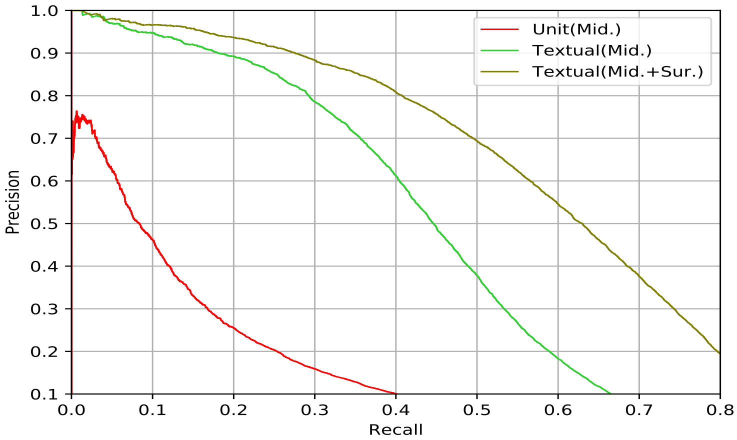


Figure 9.7: Performance of unit representation and textual representation.

significantly outperforms the unit presentation. This suggests that considering semantic meaning of each word (including entity) in a reasoning path (multi-hop inference including one-hop inference) is effective for improving performance. Additionally, combination middle context and surrounding context achieves the best performance, this indicates that it is useful to incorporate contextual information (e.g., surrounding context) for improving the performance of scientific RE.

Model	P@1k	P@2k	P@3k	P@4k	P@5k	Mean	AP
Unit(Mid.)	0.625	0.513	0.436	0.374	0.308	0.451	0.147
Textual(Mid.)	0.953	0.920	0.886	0.832	0.774	0.873	0.438
Textual(Mid.+Sur.)	**0.972**	**0.955**	**0.928**	**0.894**	**0.859**	**0.921**	**0.596**

Table 9.3: P@N and AP on unit representation and textual representation, where k=1000.

9.4 The Effect of KG-based Attention Mechanism

[13] propose a soft attention mechanism to reason over multiple reasoning paths between a target entity pair. The soft attention mechanism is calculated via Equation 9.2, 9.3 and 9.4, where, $\mathbf{s}_i$ is the vector representation of a multi-hop inference (including one-hop

inference, that is, the target sentence containing the target entity pair e_1 and e_2) between a target entity pair, $\mathbf{r}_i$ represents the vector representation of a candidate relation r_x, s_i indicates the score of a multi-hop inference (or a target sentence) regarding a candidate relation r_x.

$$P(r_x|e_1, e_2) = sigmoid(LSE(s_1, s_2, ..., s_N)) \tag{9.2}$$

$$LSE(s_1, s_2, ..., s_N) = \log(\sum_i \exp(s_i)) \tag{9.3}$$

$$s_i = \langle \mathbf{s}_i, \mathbf{r}_x \rangle \tag{9.4}$$

In this section, we compare the soft attention mechanism with the KG-based attention mechanism, which is introduced in Chapter 7 and Chapter 8. The performance is represented in Table 9.4 and Fig. 9.8, where "Soft Attention" and "KG Attention" represent the attention mechanism proposed by [13] and the KG-based attention mechanism respectively. It can be observed that KG-based attention mechanism significantly outperforms the attention mechanism proposed by [13]. This suggests that applying KG-based attention mechanism is useful for improving performance on scientific RE and further prove the effectiveness of our proposed DS-RE model.

Model	**P@1k**	**P@2k**	**P@3k**	**P@4k**	**P@5k**	**Mean**	**AP**
Soft Attention	0.957	0.942	0.910	0.877	0.835	0.904	0.566
KG Attention	**0.972**	**0.955**	**0.928**	**0.894**	**0.859**	**0.921**	**0.596**

Table 9.4: P@N and AP on the attention mechanism proposed by [13] and KG-based attention mechanism, where k=1000.

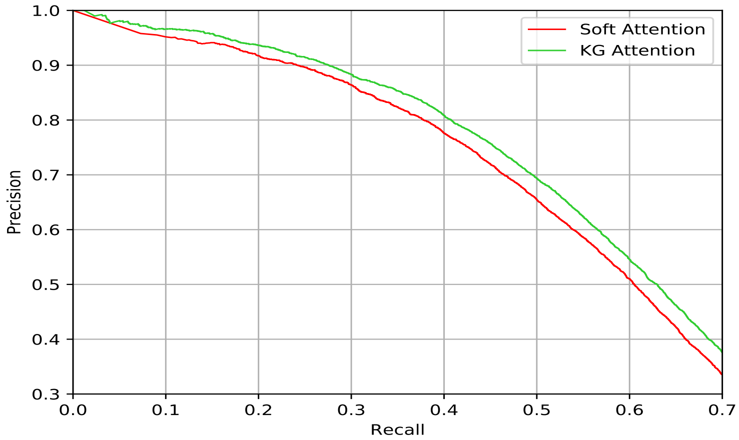


Figure 9.8: Performance of the attention mechanism proposed by [13] and KG-based attention mechanism.

9.5 Conclusion

In this chapter, we tackle the task of scientific DS-RE via the combination of the textual representation of KGI and CDI. We hypothesize that the combination of the textual representation of KGI and CDI could improve the performance of scientific DS-RE. Experimental results not only prove the effectiveness of the combination, but also suggest the importance of textual representation of reasoning path for scientific DS-RE, because our proposed model achieves significant and consistent improvements as compared with corresponding baselines. In this section, we also investigate the effect of surrounding context on the overall performance. Empirical results prove the importance of the contextual information for scientific DS-RE. In addition, we also compare the attention mechanism proposed by [13] and the KG-based attention mechanism, which is introduced in Chapter 7 and Chapter 8. Empirical results show that KG-based attention mechanism outperforms the attention mechanism proposed by [13]. This not only indicates the importance of KG-based attention mechanism for scientific DS-RE, but also prove the effectiveness of our proposed DS-RE model.

Chapter 10

Conclusion and Future Work

Background Knowledge (BK), such as coreference and entity type, has been proved to be important for Relation Extraction (RE) task [16, 15, 36]. In this thesis, we propose three approaches to extract BK from unannotated scientific papers for scientific RE. In the first method, we propose a new semantic category called Task Specific Supersense (TSS). Different from the existing fixed semantic categories, such as the hypernym in WordNet, TSS is dynamically defined based on the property of a given RE task. Evaluation on three types of scientific dataset proves the effectiveness of TSS on scientific RE. In the second one, we design a novel neural network model that not only collects feature from a given target sentence, but also extracts BK from unannotated scientific papers. We proposed two unsupervised methods: Term Sentence (TS) and Semantically Related Word (SRW). Experimental results on the RANIS corpus demonstrated that unannotated scientific papers could be used as a source of BK for scientific RE. In addition, the proposed unsupervised methods (i.e., TS and SRW) are also proven to be effective for acquiring BK from unannotated scientific papers. In the third one, we assume that the entity embedding learned by a Knowledge Graph

Completion (KGC) model could be utilized as the BK to improve the performance of scientific RE. Based on the assumption, we propose a pipeline architecture to utilize the learned entity embedding from a selected KGC model to extend a state-of-the-arts RE model. Experimental results show that incorporating the entity embedding from the KGC model could enhance the performance of scientific RE.

For Distantly Supervised Relation Extraction (DS-RE) from biomedical dataset, we propose the use of textual representation of inference to tackle the brevity of text. Specifically, scientific authors always omit the Background Knowledge (BK) that would be important for a machine to identify relationships between entities from scientific papers. To address this issue, in this thesis, we assume that the textual representation of inference (or reasoning path) over a scientific knowledge base and multiple scientific documents could be applied as the BK to fill the "gaps" in text and thus enhance the performance of scientific DS-RE. Experimental results prove the effectiveness of the inferences especially the combination of these two types of inference, because our proposed model achieves significant and consistent improvements as compared with a state-of-the-art DS-RE model.

Since manual annotation is expensive and time-consuming, our immediate future work is in the area of scientific DS-RE. Serious problems remain which limit the application of our current model. Firstly, our proposed model is not general enough to encode text, KG and inferences (i.e., inference over KG and cross-document inference) into a continuous vector space. This might negatively affect the computational cost and the flexibility of our proposed model. For instance, our proposed model is incapable for the task of entity prediction (i.e., (h, r, ?) or (?, r, t)). Secondly, the proposed model applies the strategy of random walk to search inferences for a given entity pair. Although this strategy could collect plausible inferences as discussed in previous chapters, there are still lots of noisy inferences, which could hinder the performance of our proposed model.

Our next step includes several folds of research: 1) We will redesign the architecture of the proposed model so that it is capable of encoding text (i.e., target sentences), KG inferences, cross-document inferences and KG completion into a continuous vector space. 2) In order to prevent the noisy inferences, we will apply more sophisticated strategy such as reinforcement learning [76, 39, 14] to search inferences from graph representation. Moreover, we will apply the proposed methods to further knowledge discovery. An example of this is the discovery of potential treatment: It would be interesting to use our learned inferences between drugs and diseases to discover new treatment. For instance, according to Wikipedia, **Alzheimer's disease** is a brain disease that slowly destroys brain cells. It is the sixth leading cause of death in the United States causing about 83,500 deaths a year. Unfortunately, it is incurable so far. However, based on the inference, "**Apomorphine** $\underrightarrow{may\_treat}$ **Parkinson disease** $\underleftarrow{associated\_with}$ α**-synuclein** $\underrightarrow{is\_deeply\_associated\_with}$ **Alzheimer's disease**", we could generate a hypothesis of treatment that "**Apomorphine** $\underrightarrow{may\_treat}$ **Alzheimer's disease**", because the inference indicates that **Apomorphine** might work on the protein called α**-synuclein**, and the protein deeply associates with **Alzheimer's disease**. In our future work, we focus on such knowledge discovery task and look forward to discovering such potential treatment to make a better world.

Bibliography

[1] Waleed Ammar et al. “The AI2 system at SemEval-2017 Task 10 (ScienceIE): semi-supervised end-to-end entity and relation extraction”. In: *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. 2017, pp. 592–596.

[2] Gabor Angeli et al. “Combining Distant and Partial Supervision for Relation Extraction.” In: *EMNLP*. 2014, pp. 1556–1567.

[3] Isabelle Augenstein et al. “SemEval 2017 Task 10: ScienceIE-Extracting Keyphrases and Relations from Scientific Publications”. In: *arXiv preprint arXiv:1704.02853* (2017).

[4] James Bergstra et al. “Theano: A CPU and GPU math compiler in Python”. In: *Proc. 9th Python in Science Conf*. 2010, pp. 1–7.

[5] Steven Bird et al. “The ACL Anthology Reference Corpus: A Reference Dataset for Bibliographic Research in Computational Linguistics.” In: *LREC*. 2008.

[6] Piotr Bojanowski et al. “Enriching word vectors with subword information”. In: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 135–146.

[7] Kurt Bollacker et al. “Freebase: a collaboratively created graph database for structuring human knowledge”. In: *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. AcM. 2008, pp. 1247–1250.

[8] Antoine Bordes et al. “Translating embeddings for modeling multi-relational data”. In: *Advances in neural information processing systems*. 2013, pp. 2787–2795.

[9] Elizabeth Boschee, Ralph Weischedel, and Alex Zamanian. “Automatic information extraction”. In: *Proceedings of the International Conference on Intelligence Analysis*. Vol. 71. Citeseer. 2005.

[10] Yee Seng Chan and Dan Roth. “Exploiting background knowledge for relation extraction”. In: *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics. 2010, pp. 152–160.

[11] Kai-Wei Chang et al. “Typed tensor decomposition of knowledge bases for relation extraction”. In: (2014).

[12] Kevin Bretonnel Cohen et al. “BioNLP 2017”. In: *BioNLP 2017* (2017).

[13] Rajarshi Das et al. “Chains of Reasoning over Entities, Relations, and Text using Recurrent Neural Networks”. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. 2017, pp. 132–141.

[14] Rajarshi Das et al. “Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning”. In: *arXiv preprint arXiv:1711.05851* (2017).

[15] Quang Do, Wei Lu, and Dan Roth. “Joint Inference for Event Timeline Construction”. In: *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Jeju Island, Korea: Association for Computational Linguistics, July 2012, pp. 677–687. URL: `https://www.aclweb.org/anthology/D12-1062`.

[16] Quang Do and Dan Roth. “Constraints Based Taxonomic Relation Classification”. In: *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Cambridge, MA: Association for Computational Linguistics, Oct. 2010, pp. 1099–1109. URL: `https://www.aclweb.org/anthology/D10-1107`.

[17] Jinhua Du et al. “Multi-Level Structured Self-Attentions for Distantly Supervised Relation Extraction”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2018, pp. 2216–2225.

[18] Christian Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[19] Lucie Flekova and Iryna Gurevych. “Supersense embeddings: A unified model for supersense interpretation, prediction, and utilization”. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vol. 1. 2016, pp. 2029–2041.

[20] Kata Gábor et al. “Semeval-2018 Task 7: Semantic relation extraction and classification in scientific papers”. In: *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018, pp. 679–688.

[21] Paul Groth et al. “Open Information Extraction on Scientific Text: An Evaluation”. In: *Proceedings of the 27th International Conference on Computational Linguistics*. 2018, pp. 3414–3423.

[22] Jinghang Gu et al. “Chemical-induced disease relation extraction via convolutional neural network”. In: *Database* 2017 (2017).

[23] Shu Guo et al. “Semantically smooth knowledge graph embedding”. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Vol. 1. 2015, pp. 84–94.

[24] Zhou GuoDong et al. “Exploring various knowledge in relation extraction”. In: *Proceedings of the 43rd annual meeting on association for computational linguistics*. Association for Computational Linguistics. 2005, pp. 427–434.

[25] Gus Hahn-Powell et al. “This before That: Causal Precedence in the Biomedical Domain”. In: *arXiv preprint arXiv:1606.08089* (2016).

[26] Xu Han, Zhiyuan Liu, and Maosong Sun. “Neural knowledge acquisition via mutual attention between knowledge graph and text”. In: *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

[27] Lena Hettinger et al. “ClaiRE at SemEval-2018 Task 7: Classification of Relations using Embeddings”. In: *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018, pp. 836–841.

[28] Raphael Hoffmann et al. “Knowledge-based weak supervision for information extraction of overlapping relations”. In: *Proceedings of the 49th Annual Meeting of*

the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics. 2011, pp. 541–550.

[29] Guoliang Ji et al. “Distant supervision for relation extraction with sentence-level attention and entity descriptions”. In: *Thirty-First AAAI Conference on Artificial Intelligence*. 2017.

[30] Guoliang Ji et al. “Knowledge graph embedding via dynamic mapping matrix”. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Vol. 1. 2015, pp. 687–696.

[31] Di Jin et al. “MIT-MEDG at SemEval-2018 Task 7: Semantic Relation Classification via Convolution Neural Network”. In: *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018, pp. 798–804.

[32] Seyed Mehran Kazemi and David Poole. “SimplE embedding for link prediction in knowledge graphs”. In: *Advances in Neural Information Processing Systems*. 2018, pp. 4289–4300.

[33] Jin-Dong Kim, Tomoko Ohta, and Jun’ichi Tsujii. “Corpus annotation for mining biomedical events from literature”. In: *BMC bioinformatics* 9.1 (2008), p. 10.

[34] Patrick Klein, Simone Paolo Ponzetto, and Goran Glavaš. “Improving neural knowledge base completion with cross-lingual projections”. In: Association for Computational Linguistics. 2017.

[35] Shantanu Kumar. “A Survey of Deep Learning Methods for Relation Extraction”. In: *arXiv preprint arXiv:1705.03645* (2017).

[36] Ni Lao et al. "Reading the web with learned syntactic-semantic inference rules". In: *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics. 2012, pp. 1017–1026.

[37] Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. "MIT at SemEval-2017 task 10: relation extraction with convolutional neural networks". In: *arXiv preprint arXiv:1704.01523* (2017).

[38] Jens Lehmann et al. "DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia". In: *Semantic Web* 6.2 (2015), pp. 167–195.

[39] Xi Victoria Lin, Richard Socher, and Caiming Xiong. "Multi-Hop Knowledge Graph Reasoning with Reward Shaping". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 3243–3253. DOI: 10.18653/v1/D18-1362. URL: https://www.aclweb.org/anthology/D18-1362.

[40] Yankai Lin, Zhiyuan Liu, and Maosong Sun. "Neural relation extraction with multi-lingual attention". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2017, pp. 34–43.

[41] Yankai Lin et al. "Neural relation extraction with selective attention over instances". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vol. 1. 2016, pp. 2124–2133.

[42] Hanxiao Liu, Yuexin Wu, and Yiming Yang. "Analogical inference for multi-relational embeddings". In: *arXiv preprint arXiv:1705.02426* (2017).

[43] Yi Luan, Mari Ostendorf, and Hannaneh Hajishirzi. "The UWNLP system at SemEval-2018 Task 7: Neural Relation Extraction Model with Selectively Incorporated Concept Embeddings". In: *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018, pp. 788–792.

[44] Okumura Manabu and Honda Takeo. "Word sense disambiguation and text segmentation based on lexical cohesion". In: *Proceedings of the 15th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics. 1994, pp. 755–761.

[45] Laura Mascarell. "Lexical Chains meet Word Embeddings in Document-level Statistical Machine Translation". In: *Proceedings of the Third Workshop on Discourse in Machine Translation*. 2017, pp. 99–109.

[46] Tomas Mikolov et al. "Distributed representations of words and phrases and their compositionality". In: *Advances in neural information processing systems*. 2013, pp. 3111–3119.

[47] Bonan Min et al. "Distant supervision for relation extraction with an incomplete knowledge base". In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2013, pp. 777–782.

[48] Mike Mintz et al. "Distant supervision for relation extraction without labeled data". In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing*

of the AFNLP: Volume 2-Volume 2. Association for Computational Linguistics. 2009, pp. 1003–1011.

[49] Makoto Miwa and Mohit Bansal. “End-to-end relation extraction using lstms on sequences and tree structures”. In: *arXiv preprint arXiv:1601.00770* (2016).

[50] Randall Munroe. “The rise of open access”. In: *Science* 342.6154 (2013), pp. 58–59.

[51] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. “A Three-Way Model for Collective Learning on Multi-Relational Data.” In: *ICML*. Vol. 11. 2011, pp. 809–816.

[52] Jeffrey Pennington, Richard Socher, and Christopher Manning. “Glove: Global vectors for word representation”. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, pp. 1532–1543.

[53] Bhanu Pratap et al. “Talla at SemEval-2018 Task 7: Hybrid Loss Optimization for Relation Classification using Convolutional Neural Networks”. In: *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018, pp. 863–867.

[54] Chris Quirk and Hoifung Poon. “Distant Supervision for Relation Extraction beyond the Sentence Boundary”. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. 2017, pp. 1171–1182.

[55] Sebastian Riedel, Limin Yao, and Andrew McCallum. “Modeling relations and their mentions without labeled text”. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2010, pp. 148–163.

[56] Sebastian Riedel et al. “Relation extraction with matrix factorization and universal schemas”. In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2013, pp. 74–84.

[57] Tim Rocktäschel, Sameer Singh, and Sebastian Riedel. “Injecting logical background knowledge into embeddings for relation extraction”. In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2015, pp. 1119–1129.

[58] RA Roller et al. “Improving distant supervision using inference learning”. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*. Association for Computational Linguistics. 2015.

[59] Barbara Rosario and Marti A Hearst. “Multi-way relation classification: application to protein-protein interactions”. In: *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. 2005, pp. 732–739.

[60] Jonathan Rotsztejn, Nora Hollenstein, and Ce Zhang. “ETH-DS3Lab at SemEval-2018 Task 7: Effectively Combining Recurrent and Convolutional Neural Networks for Relation Classification and Extraction”. In: *arXiv preprint arXiv:1804.02042* (2018).

[61] Cicero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. “Classifying relations by ranking with convolutional neural networks”. In: *arXiv preprint arXiv:1504.06580* (2015).

[62] Michael Schuhmacher and Simone Paolo Ponzetto. “Knowledge-based graph document modeling”. In: *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM. 2014, pp. 543–552.

[63] Stephen Soderland. “Learning information extraction rules for semi-structured and free text”. In: *Machine learning* 34.1-3 (1999), pp. 233–272.

[64] Luca Soldaini and Nazli Goharian. “Quickumls: a fast, unsupervised approach for medical concept extraction”. In: *MedIR workshop, sigir*. 2016.

[65] Fabian M Suchanek, Georgiana Ifrim, and Gerhard Weikum. “Combining linguistic and statistical analysis to extract relations from web documents”. In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2006, pp. 712–717.

[66] Yuka Tateisi et al. “Annotation of Computer Science Papers for Semantic Relation Extraction.” In: *LREC*. 2014, pp. 1423–1429.

[67] Simone Teufel et al. “Argumentative zoning: Information extraction from scientific text”. PhD thesis. University of Edinburgh, 2000.

[68] Théo Trouillon et al. “Complex embeddings for simple link prediction”. In: *International Conference on Machine Learning*. 2016, pp. 2071–2080.

[69] Shikhar Vashishth et al. “RESIDE: Improving Distantly-Supervised Neural Relation Extraction using Side Information”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2018, pp. 1257–1266.

[70] Quan Wang, Bin Wang, Li Guo, et al. “Knowledge Base Completion Using Embeddings and Rules.” In: *IJCAI*. 2015, pp. 1859–1866.

[71] Quan Wang et al. “Knowledge graph embedding: A survey of approaches and applications”. In: *IEEE Transactions on Knowledge and Data Engineering* 29.12 (2017), pp. 2724–2743.

[72] Ting Wang et al. “Automatic extraction of hierarchical relations from text”. In: *European Semantic Web Conference*. Springer. 2006, pp. 215–229.

[73] Zhen Wang et al. “Knowledge Graph Embedding by Translating on Hyperplanes.” In: *AAAI*. Vol. 14. 2014, pp. 1112–1119.

[74] Jason Weston et al. “Connecting language and knowledge bases with embedding models for relation extraction”. In: *arXiv preprint arXiv:1307.7973* (2013).

[75] Ruobing Xie, Zhiyuan Liu, and Maosong Sun. “Representation Learning of Knowledge Graphs with Hierarchical Types.” In: *IJCAI*. 2016, pp. 2965–2971.

[76] Wenhan Xiong, Thien Hoang, and William Yang Wang. “DeepPath: A Reinforcement Learning Method for Knowledge Graph Reasoning”. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sept. 2017, pp. 564–573. DOI: `10.18653/v1/D17-1060`. URL: `https://www.aclweb.org/anthology/D17-1060`.

[77] Kun Xu et al. “Semantic relation classification via convolutional neural networks with simple negative sampling”. In: *arXiv preprint arXiv:1506.07650* (2015).

[78] Zhongbo Yin et al. “IRCMS at SemEval-2018 Task 7: Evaluating a basic CNN Method and Traditional Pipeline Method for Relation Classification”. In: *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018, pp. 811–815.

[79] Daojian Zeng et al. “Relation Classification via Convolutional Deep Neural Network.” In: *COLING*. 2014, pp. 2335–2344.

[80] Daojian Zeng et al. “Distant supervision for relation extraction via piecewise convolutional neural networks”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 2015, pp. 1753–1762.

[81] Dongxu Zhang and Dong Wang. “Relation classification via recurrent neural network”. In: *arXiv preprint arXiv:1508.01006* (2015).

[82] Yu Zhao, Zhiyuan Liu, and Maosong Sun. “Representation Learning for Measuring Entity Relatedness with Rich Information.” In: *IJCAI*. 2015, pp. 1412–1418.

[83] Deyu Zhou, Dayou Zhong, and Yulan He. “Biomedical relation extraction: from binary to complex”. In: *Computational and mathematical methods in medicine* 2014 (2014).

[84] Peng Zhou et al. “Attention-based bidirectional long short-term memory networks for relation classification”. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Vol. 2. 2016, pp. 207–212.