Master's Thesis

A Study on Joint Extraction of Entities and Relations based on Constructing Differentiated Subtask-Specific Features and Enabling Fine-Grained Information Interaction

WANG, Yao

Supervisor NGUYEN, Minh Le

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)

Mar, 2024

Abstract

Extracting entities and relations from raw texts is a crucial and challenging task in the field of *Information Extraction*. Despite the successes achieved by the traditional approaches, fundamental research questions remain open. First, the subject and the object are assumed to have the same impact on their corresponding relation, ignoring the possibility of their differences when either the subject or the object is a complex entity, such as polysemous words, pronouns, and abbreviations. For instance, given a relational triple <Apple, isHeadquarteredIn, Cupertino>. The polysemous entity "Apple" may have a more important impact on the definition of the relation "isHeadquarteredIn" than the object "Cupertino". Because, if the "Apple" refers to the fruit, their relation will be of a "isProducedIn" type. Second, the information interaction mainly occurs between the subtasks of extracting the entity and relation, leaving the fine-grained interaction among the task-specific features of subjects, relations, and objects unexplored.

Motivated by the aforementioned limitations, we propose a novel model to jointly extract entities and relations. The main novelties are as follows: (1) During the encoding phase, we decouple *the whole task of jointly extracting entities and relations* into three subtasks, namely *named subject recognition*, *relation extraction* and *named object recognition*. Thanks to this, we are able to use fine-grained subtask-specific features. (2) We propose novel inter-aggregation and intra-aggregation strategies to enhance the information interaction and construct individual fine-grained subtask-specific features, respectively. (3) In the decoding phase, we combine subtask-specific features of the subject and the object to predict entities and incorporate them to enhance entity representation in the relation extraction subtask.

In order to well evaluate the effectiveness of the proposed method for jointly extracting entities and relations, we conducted a series of experiments based on seven benchmark datasets by comparing with many representative approaches. The experimental results demonstrate that: (1) when either the subject or the object is a complex entity, it has a greater impact on their corresponding relations than a normal entity. (2) Constructing fine-grained subtask-specific features for extracting the subject, the object, and their relation can improve the extraction ability. (3) Our model outperforms several previous state-of-the-art models. In specific, we increase the accuracy score by +2.7%, +0.1%, +0.6%, and +0.6% in the relation extraction task on ACE2004, ACE2005, ADE, and CoNLL04 datasets and +0.3%, +0.6%, +0.5%, +0.1%, and +0.1% in the entity extraction task on ACE2004, ADE, SciERC, NYT, and WebNLG datasets, respectively.

Contents

List of Figures

List of Tables

Acknowledgement

First, I would like to thank my advisor, Professor NGUYEN, Minh Le. He allowed me the freedom to choose my research direction and gave me a comfortable environment with an academic atmosphere that allowed me to focus on my research. He also gave me patience and guidance in paper writing, methodology, and research ideology. Moreover, his valuable comments and suggestions for corrections made my thesis more rigorous and better. I would also like to thank for his trust and support, which made me more determined to pursue my future research path. Without him, I could not finish my main research.

I would like to thank Associate Professor RACHARAK, Teeradaj, he recommended and explained the paper to me in detail and gave me a lot of inspiration for my research.

I would like to thank Professor UNOKI, Masashi for his patience and kind guidance for my minor research, and financial support. From him, I learned about how to build a model from a researcher's perspective, and the philosophy when writing a paper. Also, he started my transition from an engineer to a researcher.

Finally, I want to thank all my friends who helped me during my study.

Chapter 1

Introduction

1.1 Research Background

Sentence: Also the Pentagon is seeing lighter than expected resistance indicating at least that they may have really seriously degraded those Republican Guard divisions before the U.S. troops arrived, both in Karbala and also in Al Kut.		
NER	Normal	(Pentagon, ORG), (Republican Guard, ORG), (U.S, GPE), (troops, PER), (Karbala, GPE), (Al Kut, GPE)
	Polysemous	(**they**, ORG), (**divisions**, PER)
RE	Normal	(troops, PHYS, Karbala), (troops, PHYS, Al), (troops, ORG-AFF, U.S)
	Polysemous	(**divisions**, ORG-AFF, Republican)

Table 1.1: An example sentence with normal and complex entities. Detailed meanings of the abbreviations of entity and relation types are shown in Table 4.3.

Named Entity Recognition (NER) and Relation Extraction (RE), as two essential subtasks in information extraction, aim to extract entities and relations from semi-structured and unstructured texts. They are used in many downstream applications in different domains, such as knowledge graph construction (38; 39), Question-Answering (36; 37), and knowledge graph-based recommendation system (40; 41). Most traditional models and some methods used in specialized areas (43; 9; 46; 35) construct separate models for NER and RE to extract entities and relations in a pipelined manner. This type of method suffers from error propagation and unilateral information interaction. Thus, many works adopt joint extraction strategy (1; 25; 16; 28; 2; 3; 4; 5; 32; 42; 27) that constructs a unified model to jointly extract entities and relations in recent years, effectively alleviating the error propagation. However, information interaction in these methods mainly focuses on parameter sharing, feature sharing, or distinct interactive features between NER and RE, which leads to two problems.

First, they (16; 28; 4) default to the same impact of the subject and the object on their relations, ignoring the possibility of differences between them. This issue is crucial for determining relational types. For example, Table 1.1 provides a sentence containing normal and complex entities. Some normal entities' types are relatively simple, which plays little role in the judgment of relational types. However, some complex entities, such as polysemous words, pronouns, and abbreviations, may have multiple types, the judgment of their types will largely determine the relational types. The polysemous word "divisions" has multiple meanings and can refer to an institution (organization) or troopers (person). When determining the relation between "divisions" and "Republican Guard", if the entity "division" is determined as an "ORG" (organization) type, then the relation between them will be the "PART-WHOLE" (subsidiary) type. If the "divisions" entity is of type "PER" (person), their relation will type "ORG-AFF" (ownership or founder).

Second, since these researches default to the same impact of the subject and the object on their relations, lacking fine-grained feature construction and information interaction among them. This is crucial to determine the entity and relation types. In the example sentence, the single "divisions" entity is less likely to express its accurate type of semantic information (organization or persons), which makes it difficult to determine its involved relations. The information interaction with the object "Republican Guard" may help determine its type "person" and relation type "ORG-AFF".

1.2 Research Objectives

To address the above issues, we propose a novel joint model to construct fine-grained task-specific features for relational triples and enhance the information interaction among subjects, relations, and objects. Our main works are as follows:

First, in the encoding phase, to construct fine-grained semantic representations, we decouple the task into three subtasks: named subject recognition (NSR), named object recognition (NOR), and RE. Then, we design three task-specific cells that serve the functions of acquiring, storing, and interacting information for individual subtasks to construct the task-specific features for each subtask, respectively. Next, we design an aggregating method to perform and enhance fine-grained information interaction among NSR, NOR, and RE. It contains two parts: inter-aggregation and intra-aggregation. Inter-aggregation combines the features of different task-specific cells, such as the features of NSR-NOR, NOR-RE, and NSR-RE subtasks, which aims to realize the mutual information interaction among different subtasks. Intra-aggregation incorporates the mutually interacted features into each task-specific cell to enhance the context semantics and enable differentiated interaction information in three task-specific cells.

Second, in the decoding phase, the NSR and NOR task-specific features are combined to create the NER features. We continue to aggregate and incorporate them to enhance the entity semantics for the RE task.

Third, we conduct extensive experiments to validate whether constructing independent subtasks for subjects, relations, and objects is an excellent way to capture the semantic difference between task-specific features and whether an impact difference exists between the subject and the object for their relations.

Finally, we also analyze the limitations and shortcomings of our research through the experimental results. In addition, we also show our contributions to this research direction and future works.

1.3 Thesis Organization

We organize this study into five chapters. The first chapter introduces the remaining unexplored questions of the previous research. Then, we describe the objective of this study. The rest of the chapters are summarized as follows:

- **Chapter 2: Literature Review** We introduce and compare the representative or newly proposed pipeline and joint approaches in the field of entity and relation extractions. Then, we introduce the related background knowledge that is used in our proposed method.

- **Chapter 3: Methodologies** We define the research problem and describe our proposed method in detail. We also introduce the training strategy in our model.

- **Chapter 4: Experiments and Analysis** First, We introduce the experimental datasets and count the number of various types to show the individual characteristics of every dataset. Second, we describe the baseline models that exist for the unexplored questions we will solve. Third, we introduce the experimental settings and the training details. Finally, we conduct extensive experiments and deeply analyze the results to evaluate the effectiveness and limitations of our model in various dimensions.

- **Chapter 5: Conclusion** We summarize the contributions of this study and analyze some potential research questions in future works.

Chapter 2

Literature Review

2.1 Related Works

In this chapter, we introduce some related research in the field of entity and relation extractions. According to the extraction procedure, it can be divided into two classes: Pipeline extraction and joint extraction.

2.1.1 Pipeline Extraction Methods

Pipeline extraction method construct construct two separate models to encode NER and RE in a sequential manner. It is mainly used in many traditional models. Their information interaction is unilateral as it passes from the NER to the RE model. For example, (9) proposed a pipelined model that consists of a NER model and a RE model. The NER model first predicts the span and type of entities. Then, the RE model inserts extra marker tokens to highlight the subject and object and their types of all candidate entities output from the NER model. (46) proposed a pattern-first pipeline approach that contains three steps. It first uses a machine reading comprehension-based method to identify potential patterns to facilitate the construction of refined questions in the subsequent entity extraction stage. Then, a span-based method is used to extract all the entities. Finally, an error elimination strategy is applied to eliminate falsely extracted candidate entity-relation triples. Although these methods achieved high scores in NER and RE, they still suffer from the error propagation problem. The extracted wrong entities pass into the RE model, resulting in wrong relation triples. Thus, many researchers proposed a joint model to extract entities and relations to alleviate this problem.

2.1.2 Joint Extraction Methods

Another type is the joint extraction method, which extracts entities and relations simultaneously in a unified model. For example, (1) proposed a joint model, which incorporates entity information into the RE task through the copying mechanism. (25) designed a cascading sequential annotation model that extracts relational triples by mapping (subject entities, relations) to object entities. (20) proposed a joint extraction model based on a span schema. It first uses a span classifier to segment sentences. Then use a span filter to determine the entity. Finally, a relation classifier is used to predict relational triples. These models establish a unidirectional interaction between NER and RE tasks, where entity tasks cannot acquire features from relation tasks during encoding.

(19) proposed a task-specific bidirectional RNN model that emphasizes the significance of shared and task-specific parameters for relation extraction. (10) designed two separate encoders to generate task-specific features for entities and relations, enabling mutual interaction and enhancement between the two tasks. (29) introduced a recurrent interaction network for NER and RE, extending the encoding structure to a graph structure that facilitates interaction between the two tasks through a shared network. Building upon Sun's model (29), (30) added a cross-attention interaction network to enhance the information interaction of entity and relation types. (4) proposed a translating schema-based model that infers object entities by constructing a self-attention mechanism between the features of subjects-relations and object entities. However, the information interaction in this model is limited to parameter sharing and does not fully leverage the interconnections between NER and RE tasks. (32) proposed a joint model that decomposes the entity relational triple extraction into three subtasks: relation judgment, entity extraction, and subject-object alignment. These tree subtasks serve the prediction of relational types, relation-involved potential entities, and relational triples. (5) proposed a joint encoding model highlighting the importance of shared features between NER and RE tasks. (42) proposed a joint extraction method using a sampling and interaction method. It divides negative samples into sentences based on whether they overlap with positive samples to enhance the accuracy of the NER task. Then, it introduces a GNN model to enhance the interaction between NER and RE modules. (27) proposed a joint model that adopts a boundary regression mechanism to enhance the extraction of possible entities. However, the information interaction in encoding is still a sequential order. (33) proposed to encode semantic representation with different granularities for NER and RE tasks and perform information interaction between them by a cross-attention approach.

However, these models do not consider the different impacts of the subject and object on their relations when either the subject or the object is a complex entity. Moreover, they mainly focus on the information interaction between NER and RE tasks, overriding fine-grained feature construction and interaction among subjects, relations, and objects.

2.2 Background Knowledge

2.2.1 Long Short-Term Memory Model

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN), which is proposed to overcome the vanishing gradient problem in traditional sequence-to-sequence models. The LSTM model processes information based on LSTM cells. Every LSTM cell is composed of a cell state, a hidden state, and three gates. The following are the equations of each component:

$$i_t = \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \tag{2.1}$$

$$f_t = \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \tag{2.2}$$

$$\tilde{c}_t = \tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \tag{2.3}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{2.4}$$

$$o_t = \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \tag{2.5}$$

$$h_t = o_t \odot \tanh(c_t) \tag{2.6}$$

where x_t, h_t, and c_t are input, hidden state, and the cell state at time step t. i_t, f_t, and o_t denote the input gate, forget gate and output gate. $\tilde{c}_t$ represents the current candidate cell state that is used to build the current cell state. σ, tanh, and $\odot$ denote the sigmoid activation function, the hyperbolic tangent activation function, and the element-wise multiplication, respectively.

The cell state c_t, as the long-term memory of the network, allows the model to capture the long-range dependencies. The hidden state, serving as the short-term memory of the network, captures the related information from the current input. Then, the forget gate decides what information from the cell state should be thrown away or kept. Next, the input gate determines what new information can be stored in the cell state. Then, the information update module $\tilde{c}_t$ computes the new candidate values for the cell state by combining the current input x_t and the previous hidden state h_{t-1}. The output gate o_t computes the current hidden state based on the current cell state and decides what to output. Finally, the current hidden state h_t is computed and passed into the next LSTM cell to transfer the information.

Through these mechanism, LSTM model is used to encode features in various tasks, such as Relation Extraction (3), Entity Alignment (47), and Question Answering (48).

2.2.2 BERT Model: Bidirectional Encoder Representations from Transformers

The BERT model (11) is designed to pretrain deep bidirectional representations from unlabeled text. It can be fine-tuned with only an additional output layer for a wide range of tasks, such as Question Answering (49) and Nature Language Inference (50), without substantial task-specific architecture modifications. Additionally, it also has the following characteristics:

- High efficiency: BERT is built on the Transformer encoder. Since it is based on self-attention mechanisms, it allows for parallel processing of input sequences, making them highly efficient for training and inference.
- Large-Scale Training: BERT is pre-trained on an enormous 3.3 billion word dataset, including Wikipedia and Google's BooksCorpus.
- Birectional encoding: It considers the entire context of a word by looking at it from both directions, which significantly improves the understanding of context and semantics.
- BERT provides contextualized word representations. The representation of a word depends not only on its context but also on the surrounding words in a sentence. This allows BERT to capture nuances and polysemy in language.

Additionally it also contains two main unsupervised pretraining objectives: Masked Language Model (MLM) and Next Sentence Prediction (NSP).

MLM: Masked Language Model

The MLM aims to randomly make some of the words in a sentence and train the model to predict the masked words based on the context provided by the surrounding words. In training sentences, about 15% of the words (tokens) are randomly selected and replaced with a special "[MASK]" token. The model is then tasked with predicting the original identity of the masked words. Since there is no Token like "[MASK]" in the input sequence during fine-tuning, this will lead to a mismatch between pre-training and fine-tuning.

To address this problem, 15% tokens are selected, then among them, 80% of them are replaced with "[MASK]", 10% of them are replaced with other tokens, and the resting tokens are unchanged. Finally, the corresponding output of these tokens is used for classification.

NSP: Next Sentence Prediction

Some downstream tasks need to analyze the relations between two sentences, such as Question Answering and Semantic Similarity. In order to enable the model to have this ability, This strategy is proposed to predict whether the second sentence is a consecutive sentence following the first or a randomly chosen sentence unrelated to the first.

Both the MLM and the NSP contribute to the creation of a powerful contextualized language model. Pretraining BERT with these objectives allows it to capture rich contextual information, making it highly effective for a wide range of downstream natural language processing tasks.

Chapter 3

Methodologies

3.1 Problem Statement

Training the model involves two tasks: NER and RE. Let $\mathcal{E}$ and $\mathcal{R}$ represent the predicted entities and relations sets, respectively. Let $\mathcal{K}$ and $\mathcal{L}$ denote the pre-defined entity types and relation types with total numbers of u and v. Given a sentence $s = \{w_1, ..., w_t\}$ consisting of t words. The NER task focuses on extracting entities $\mathrm{e}_{ij}^{k} = \{(w_i, w_j, k) \mid e \varepsilon \mathcal{E}, 1 \leq i, j \leq t, k \varepsilon \mathcal{K}\}$, where i and j denote the head and tail positions of an entity in a sentence, while k denotes its type. The RE task aims to identify relation types between subjects and objects. Formally, $\mathrm{r}_{im}^{l} = \{(w_i, w_m, l) \mid r \varepsilon \mathcal{R}, 1 \leq m, n \leq t, l \varepsilon \mathcal{L}\}$, where i and m represent the head position of the subject and object, and l represents their relation type. In joint extraction, the final set of predicted relational triples is denoted as $\langle \mathrm{e}_{ij}^{k1}, \mathrm{r}_{im}^{l}, \mathrm{e}_{mn}^{k2} \rangle$, where $\mathrm{e}_{ij}^{k1}, \mathrm{e}_{mn}^{k2} \varepsilon \mathcal{E}$; $\mathrm{r}_{im}^{l} \varepsilon \mathcal{R}; k1, k2 \varepsilon \mathcal{K}$. For any sentence that does not contain entities or relations, their labels will be empty.

3.2 Proposed Model

Figure 3.1 illustrates the overall structure of our model called DArtER, which stands for **D**ecoupling and **A**ggregating Network for Joint ext**r**ac**t**ion of **E**ntities and **R**elations. It consists of three main components: an encoder and two decoders for NER and RE. The encoder comprises several **D**ecoupling and **A**ggregation **M**odules known as DAM. Every DAM contains three task-specific cells for NSR, NOR, and RE, which serve the functions of information interaction and fine-grained feature construction. Each DAM module generates three outputs: task-specific features, hidden, and cell states. The decoder uses the task-specific features for prediction, while

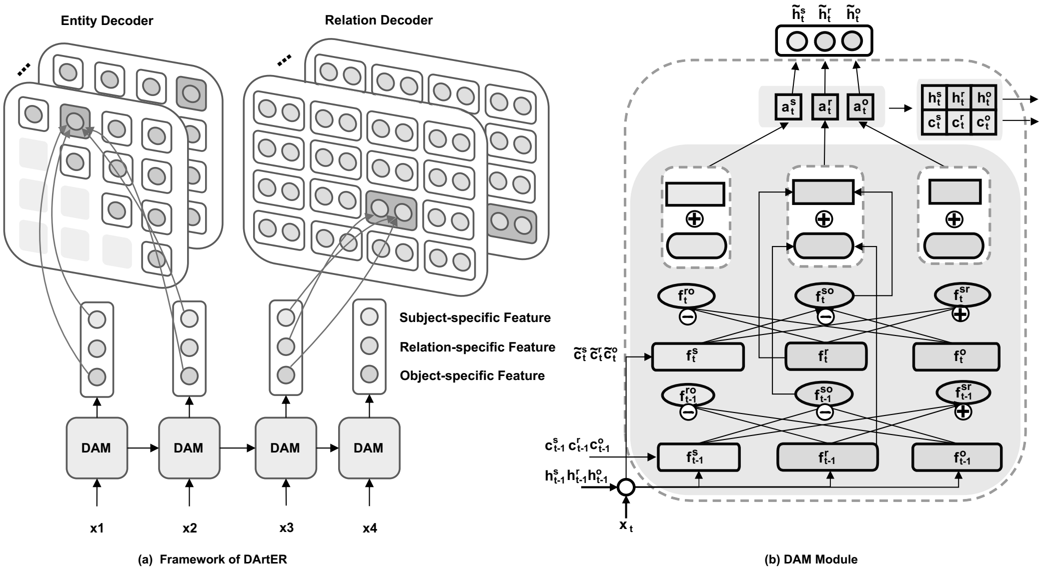

Figure 3.1: The overall framework of the DArtER model.

the others are passed on to the next DAM for feature construction.

3.2.1 Encoder

Let $X = \{x_1, \cdots, x_t\}, X \in \mathbb{R}^{d_t \times p}$ denote the feature matrix of a sentence extracted by a pre-trained language model. The transformation is implemented by feeding X into three linear layers. The process is formalized as:

$$Z^s = XW_{z\_s} + b_{z\_s};\ Z^r = XW_{z\_r} + b_{z\_r};\ Z^o = XW_{z\_o} + b_{z\_o} \tag{3.1}$$

where $W_{\{\cdot\}}$ and $b_{\{\cdot\}}$ are learnable parameters. The outputs $Z^s, Z^r, Z^o \in \mathbb{R}^{d_t \times h}$ are three representations that are used in each task-specific cells of subjects, relations, and objects in DAM, respectively. In every task-specific cell of each DAM, we perform the linear transformation of the hidden features $\mathrm{h}_{t-1}^p \in \mathbb{R}^{d_h}$ from the previous DAM, then combine its output with the current token embeddings z_t^p to generate the current individual task-specific features $f_t^p \in \mathbb{R}^{d_h}$ and the candidate cell state $\tilde{c}_t^p \in \mathbb{R}^{d_h}$.

$$\begin{aligned} \mathrm{f_t^p} &= z_t^p + (\mathrm{h}_{t-1}^p W_{f\_p} + b_{f\_p}) \\ \mathrm{\tilde{c}_t^p} &= Tanh(z_t^p + (\mathrm{h}_{t-1}^p W_{c\_p} + b_{c\_p})) \end{aligned} \tag{3.2}$$

where $p \in \{s, r, o\}$, denoting **s**ubjects, **r**elations, and **o**bjects. Then, we use an inter-aggregating method to enable mutual information interaction among task-specific cells as follows.

$$\begin{aligned} f_t^{ro} &= f_t^o - f_t^r \\ f_t^{so} &= f_t^o - f_t^s \\ f_t^{sr} &= f_t^s + f_t^r \end{aligned} \tag{3.3}$$

f_t^{ro}, f_t^{so} and $f_t^{sr} \in \mathbb{R}^{d_h}$ are the inter-aggregated features of RE-NOR, NSR-NOR, and NSR-RE at the current time. To enhance semantic context and enable differentiated interaction information in three task-specific cells, we perform an intra-aggregating approach within every task-specific cell by incorporating the inter-aggregated features into the individual original features from both the previous and current time steps. This results in three enhanced task-specific features: a_t^s, a_t^r, and $a_t^o \in \mathbb{R}^{d_h}$.

$$\begin{aligned} a_t^s &= (f_{t-1}^s + f_{t-1}^{ro}) \odot c_{t-1}^s + (f_t^s + f_t^{ro}) \odot \tilde{c}_t^s \\ a_t^r &= (f_{t-1}^r + f_{t-1}^{so}) \odot c_{t-1}^r + (f_t^r + f_t^{so}) \odot \tilde{c}_t^r \\ a_t^o &= (f_{t-1}^o + f_{t-1}^{sr}) \odot c_{t-1}^o + (f_t^o + f_t^{sr}) \odot \tilde{c}_t^o \end{aligned} \tag{3.4}$$

The symbol $\odot$ denotes element-wise multiplication. $\tilde{c}_t^s$, $\tilde{c}_t^r$, and $\tilde{c}_t^o$ are generated in Equation 3.2. c_{t-1}^s, c_{t-1}^r, and c_{t-1}^o are come from the previous DAM. Finally, the aggregated features are utilized to create the final task-specific features $\tilde{\mathrm{h}}_t^p \in \mathbb{R}^{d_h}$, hidden states $\mathrm{h}_t^p \in \mathbb{R}^{d_h}$, and cell states $\mathrm{c}_t^p \in \mathbb{R}^{d_h}$.

The following equation shows the formalization where $p \in \{s, r, o\}$.

$$\begin{aligned} \tilde{\mathrm{h}}_t^p &= Tanh(a_p^t) \\ \mathrm{c}_t^p &= a_p^t \; W_{a\_p} + b_{a\_p} \\ \mathrm{h}_t^p &= Tanh(\mathrm{c}_t^p) \end{aligned} \tag{3.5}$$

3.2.2 Decoder

In the NER decoder, we combine the NSR and NOR features to form the NER features $\tilde{\mathrm{h}}_t^e \in \mathbb{R}^{d_h}$. We apply a linear transformation to all the possible entity span features $[\tilde{\mathrm{h}}_i^e; \tilde{\mathrm{h}}_j^e]$ and then normalize them, enabling the integration of features among different words. ; denotes the vector concatenation. The resulting features are output through the ELU activation function, which aids the model's quick convergence.

$$\begin{aligned} \tilde{\mathrm{h}}_t^e &= \tilde{\mathrm{h}}_t^s + \tilde{\mathrm{h}}_t^o \\ \mathrm{h}_{ij}^e &= ELU(Norm([\tilde{\mathrm{h}}_i^e; \tilde{\mathrm{h}}_j^e] W_{h\_e} + b_{h\_e})) \\ \tilde{\mathrm{e}}_{ij}^k &= Sigmoid(\mathrm{h}_{ij}^e W_e + b_e) \end{aligned} \tag{3.6}$$

Finally, the probabilities of the entities $\tilde{e}_{ij}^{k} \in \mathbb{R}^{d_{t \times t \times u}}$ ($\langle i, k, j \rangle$) with the start word i, end word j, and type k are predicted by feeding the features into a fully connected layer with a sigmoid activation function.

For the RE decoder, we inter-aggregate the features of subjects and objects by computing the element-wise subtraction between them. In Equation 3.7, constants $\alpha, \beta \in \{-1, 0.5, 1\}$ are aggregating parameters obtained through grid search on the validation set. Then, we incorporate the aggregated features into the RE decoder. Finally, we predict the probabilities of the relations $\tilde{r}_{im}^{l} \in \mathbb{R}^{d_{t \times t \times v}}$ ($\langle i, l, m \rangle$) with the type l, as well as the start word i and m for both the subjects and objects, respectively.

$$
\begin{aligned}
\tilde{h}_t^r &= \tilde{h}_t^r + (\alpha \tilde{h}_t^o - \beta \tilde{h}_t^s) \\
h_{im}^r &= ELU(Norm([\tilde{h}_i^r; \tilde{h}_m^r] W_{h\_r} + b_{h\_r})) \\
\tilde{r}_{im}^l &= Sigmoid(h_{im}^r W_r + b_r)
\end{aligned}
\tag{3.7}
$$

3.2.3 Bi-Encoder

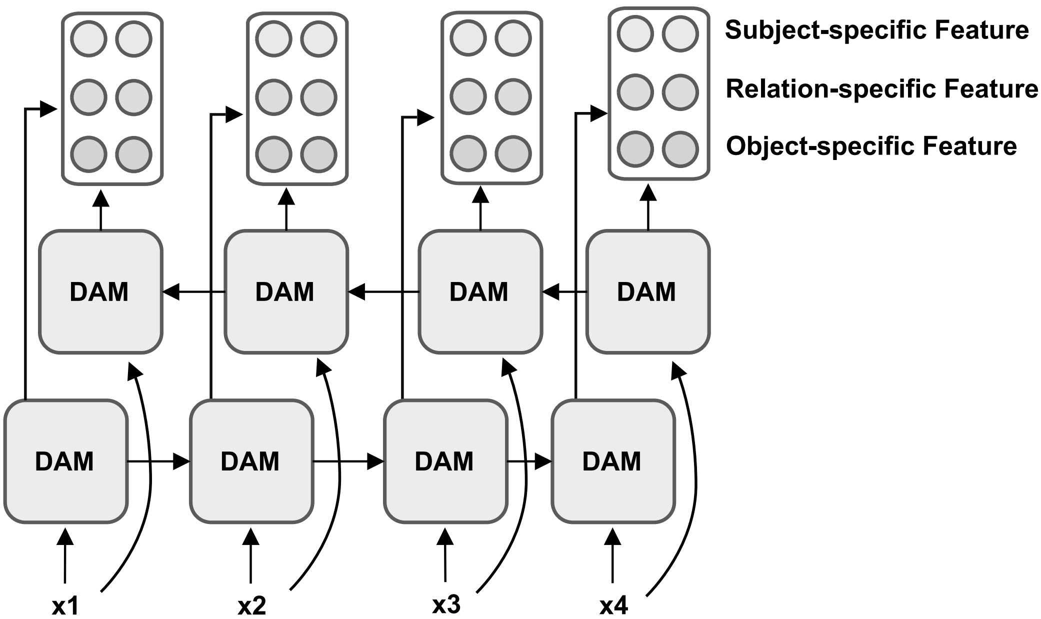

Figure 3.2: The overall framework of the BiDArtER model.

We also design an extension model called *BiDArtER* to extract features bi-directionally. As shown in Figure 3.2, the first-layer encoder captures features from left to right, while the second-layer from right to left within a sentence. The obtained features from the individual tasks of both encoders are combined and simultaneously fed into the decoders. Consequently, the decoding formulas for NER and RE are adjusted as follows: For NER:

$$
\begin{aligned}
&\overrightarrow{\tilde{\mathrm{h}}}^e_t = \overrightarrow{\tilde{\mathrm{h}}}^s_t + \overrightarrow{\tilde{\mathrm{h}}}^o_t \\
&\overleftarrow{\mathrm{h}}^e_t = \overleftarrow{\mathrm{h}}^s_t + \overleftarrow{\mathrm{h}}^o_t \\
&\mathrm{h}^e_{ij} = ELU(Norm([\overrightarrow{\tilde{\mathrm{h}}}^e_i; \overrightarrow{\tilde{\mathrm{h}}}^e_j; \overleftarrow{\mathrm{h}}^e_i; \overleftarrow{\mathrm{h}}^e_j]W_{h\_e} + b_{h\_e})) \\
&\tilde{\mathrm{e}}^k_{ij} = Sigmoid(\mathrm{h}^e_{ij} W_e + b_e)
\end{aligned}
\tag{3.8}
$$

For RE:

$$
\begin{aligned}
&\overrightarrow{\tilde{\mathrm{h}}}^r_t = \overrightarrow{\tilde{\mathrm{h}}}^r_t + (\alpha\overrightarrow{\tilde{\mathrm{h}}}^o_t - \beta\overrightarrow{\tilde{\mathrm{h}}}^s_t) \\
&\overleftarrow{\mathrm{h}'}^r_t = \overleftarrow{\mathrm{h}}^r_t + (\alpha\overleftarrow{\mathrm{h}}^o_t - \beta\overleftarrow{\mathrm{h}}^s_t) \\
&\mathrm{h}^r_{im} = ELU(Norm([\overrightarrow{\tilde{\mathrm{h}}}^r_i; \overrightarrow{\tilde{\mathrm{h}}}^r_m; \overleftarrow{\mathrm{h}}^r_i; \overleftarrow{\mathrm{h}}^r_m]W_{h\_r} + b_{h\_r})) \\
&\tilde{\mathrm{r}}^l_{im} = Sigmoid(\mathrm{h}^r_{im} W_r + b_r)
\end{aligned}
\tag{3.9}
$$

where $\rightarrow$ and $\leftarrow$ represent the left-to-right and right-to-left encodings, respectively.

3.2.4 Training

We threshold at 0.5 for the NER and RE tasks: $\mathrm{e}^k_{ij} := (\tilde{\mathrm{e}}^k_{ij} > 0.5)$ and $\mathrm{r}^l_{im} := (\tilde{\mathrm{r}}^l_{im} > 0.5)$. Here, e^k_{ij} and r^l_{im} represent the predicted entities and relations, respectively. The model is trained using the binary cross-entropy loss function. The total loss L_{total} is composed of L_{ner} and L_{re} as follows, where $\hat{\mathrm{e}}^k_{ij}$ and $\hat{\mathrm{r}}^l_{im}$ represent the gold labels of the entities and relations, respectively. $\mathcal{E}$ and $\mathcal{R}$ denote the entity and relation sets. We determine the constants γ and δ through grid search on the validation set, testing different values such as 0.75, 0.85, and 1.0 to find the best weights for each task.

$$
\begin{aligned}
&L_{total} = \gamma L_{ner} + \delta L_{re} \\
\text{where } &L_{ner} = -\sum_{\hat{\mathrm{e}}^k_{ij} \in \mathcal{E}} \hat{\mathrm{e}}^k_{ij} log(\mathrm{e}^k_{ij}) + (1 - \hat{\mathrm{e}}^k_{ij}) log(1 - \mathrm{e}^k_{ij}) \\
&L_{re} = -\sum_{\hat{\mathrm{r}}^l_{im} \in \mathcal{R}} \hat{\mathrm{r}}^l_{im} log(\mathrm{r}^l_{im}) + (1 - \hat{\mathrm{r}}^l_{im}) log(1 - \mathrm{r}^l_{im})
\end{aligned}
\tag{3.10}
$$

Chapter 4

Experiments and Analysis

4.1 Datasets

Dataset	**Train**	**Dev**	**Test**	**Type$_e$**	**Type$_r$**
CONLL04	922	231	288	4	5
ADE	4,272	10-fold cross-validation		2	1
SciERC	1,861	275	551	6	7
ACE2004	8,683	5-fold cross-validation		7	6
ACE2005	10,051	2,424	2,050	7	6
NYT	56,196	5,000	5,000	1	24
WebNLG	5,019	500	703	1	170

Table 4.1: Statistics of all datasets.

We conducted experiments on seven benchmark datasets: the CoNLL04 dataset (7), the ADE dataset (8), the SciERC dataset (6), the ACE2004 dataset (24), and the ACE2005 dataset (23), the NYT dataset (21), the WebNLG dataset (22). The NYT and WebNLG datasets are partially annotated, where only the tail positions of entities are annotated. This means that the datasets provide information about where entities are mentioned but do not include annotations for their specific roles or relations. Other datasets are fully annotated, meaning an entity's head and tail positions are labeled. The statistics of the number of entities, relations, entity types (Type$_e$), and relation types (Type$_r$) are shown in Table 4.1. BERT (11), SciBERT (13), and ALBERT (12) are the pre-trained language models used for different datasets in our work. The details of each dataset are described as follows.

- The **SciERC** dataset is extensively used for relation extraction and named entity recognition tasks in scientific papers. It contains six entity types (e.g., task, method, and material) and eight relation types, including ordinary and co-referential relations. Our experiments focus on general relations, which consist of seven relation types.

- The **CoNLL04** dataset is extracted from news articles and contains five relation types and four entity types.

- The **ADE** dataset is related to the biomedical domain and focuses on extracting the adverse effects of drugs and diseases. It consists of two entity types (drug, Adverse-Effect) and one relation type (Adverse-Effect).

- The **ACE2004** dataset is a benchmark dataset developed by the Linguistic Data Consortium (LDC) for evaluating NLP systems in 2004. It contains 8683 sentences with seven entity types and six relation types. We use the English version of this dataset for training, validation, and testing.

- The **ACE2005** dataset is an extended version of ACE2004. It contains 10,051 training, 2,424 development, and 2,050 test sentences. ACE2005 is a larger dataset compared to ACE2004 regarding the number of sentences.

- The **NYT** dataset is obtained from the New York Times Corpus using distantly supervised methods and is aligned with Freebase.

- The **WebNLG** dataset was initially extracted using natural language generation and has become a commonly used dataset for relation extraction tasks.

Table 4.2 provides the statistics of different types of entities and relations on the train, dev, and test datasets. For example, in the CoNLL04 dataset, the entity type "Peop" contains 318 entities, and the relation type "kill" includes 47 relational triples in the test dataset. As for the NYT dataset, we categorized it into two classes: one containing relational types starting with "people/*" and the other containing the remaining types ("Others" type). For the WebNLG dataset, we counted only the total number of entities and relational triples. For the other datasets, we counted the number of all entity and relational types. In addition, Table 4.3 explains the abbreviation of entity

Dataset	Entity type	Train	Dev	Test	Relation Type	Train	Dev	Test
WebNLG	None	15854	2187	1536	Total	11687	1581	1112
NYT	None	120776	10777	10794	Total	88253	8110	7976
	-	-	-	-	/people/*	17713	1582	1528
	-	-	-	-	Others	70540	6528	6448
CONLL04	Total	3315	875	1059	Total	1254	331	402
	Peop	1066	278	318	Kill	179	42	47
	Org	602	168	195	OrgBased_In	260	71	93
	Other	453	116	132	Work_For	249	69	76
	Loc	1194	313	414	Live_In	322	84	97
	-	-	-	-	Located_In	244	65	89
ADE	Total	7891	1400	1032	Total	4867	875	636
	Drug	3650	640	477	Adverse-Effect	4867	875	636
	Adverse-Effect	4241	760	555	-	-	-	-
SciERC	Total	4877	678	1445	Total	3196	453	970
	OtherScientificTerm	1245	166	413	Used-for	1678	212	529
	Generic	835	116	209	Feature-of	173	32	59
	Task	806	112	239	Evaluate-for	309	50	91
	Method	1289	189	377	Conjunction	400	59	123
	Metric	213	36	67	Part-of	177	27	63
	Material	489	59	140	Hyponym-of	294	44	67
	-	-	-	-	Compare	165	29	38
ACE2004	Total	14732	2567	4351	Total	2778	480	815
	ORG	2811	514	846	OTHER-AFF	97	20	28
	GPE	2765	460	818	EMP-ORG	1108	194	325
	VEH	140	22	41	GPE-AFF	355	60	105
	FAC	465	80	135	PER-SOC	246	40	73
	LOC	416	65	121	PHYS	823	143	242
	PER	8059	1416	2366	ART	149	23	42
	WEA	76	10	24	-	-	-	-
ACE2005	Total	25165	6049	4492	Total	4766	1123	795
	ORG	3647	954	1014	ORG-AFF	1469	365	26
	GPE	4980	1207	847	PART-WHOLE	772	160	319
	VEH	659	123	47	GEN-AFF	509	123	84
	FAC	896	243	110	PER-SOC	432	102	81
	LOC	809	153	94	PHYS	1095	277	247
	PER	13526	3247	2337	ART	489	96	38
	WEA	648	122	43	-	-	-	-

Table 4.2: Statistics of all datasets, where the type "Total" is the sum of all types.

and relation types on ACE2004 and ACE2005 datasets. Every relation type represents multiple sub-types. RE task is to predict the coarse type instead of the sub-types.

Entity		Relation	
Type	Meaning	Type	Sub-Type
ORG	organization	PHYS	Located, Near
GPE	geopolitical	PART-WHOLE	Geographical, Subsidiary, Artifact
VEH	vehicle	PER-SOC	Lasting-Personal, Business, Family
FAC	facility	ORG-AFF	Employment, Ownership, Founder, Student-Alum, Sports-Affiliation, Investor-Shareholder, Membership
LOC	location	ART	User-Owner-Inventor-Manufacturer
PER	person	GEN-AFF	Citizen-Resident-Religion-Ethnicity, Org-Location
WEA	weapons	OTHER-AFF	Ethnic, Ideology, Other
-	-	EMP-ORG	Employ-Exec, Employ-Staff, Employ-Undetermined, Member-of-Group, Subsidiary, Partner*, Other*
-	-	GPE-AFF	Citizen-Resident, Based-In, Other

Table 4.3: Detailed entity and relation types on ACE2004 and ACE2005 datasets.

4.2 Baseline Models

To demonstrate the different semantic granularities between the subject and the object and the impact difference of them on their relations, we compare our results with the following related models.

- TpLinker (26) proposed a one-stage joint extraction model with a novel handshaking tagging scheme.
- PURE (9) proposed a simple pipelined approach that uses the NER model to construct the input for the RE model.
- TDEER (4) proposed a novel translating decoding schema for joint extraction of entities and relations.

- RIFRE (31) proposed a representation iterative fusion based on heterogeneous graph neural networks for relation extraction.
- PRGC (32) proposed a joint relational triple extraction framework based on Potential Relation and Global Correspondence, which constructs three subtasks to enhance extraction: relation judgment, entity extraction, and subject-object alignment from a novel perspective.
- BR (27) proposed a boundary regression model for joint NER and RE with a boundary regression mechanism to learn the offset of possible entities to enhance the RE task.
- Table-Sequence (10) proposed a joint extraction model with two different encoders designed to interact with each other.
- PFN (5) proposed a partition filter network to properly model two-way interaction between NER and RE tasks.
- UNIRE (17) proposed a unified classifier to predict each cell's label, which unifies and enhances the learning of two subtasks.
- TablERT (15) proposed a method to extract entities and relations based on table representation, which enables information interaction between NER and RE by a beam search approach.
- IEER (34) proposed a joint entity and relation extraction method based on information enhancement. It uses a special marking strategy to mark and integrate NER and RE features and enhance their mutual interaction to address the discriminative problem of entity and relation in the triple overlapping problem.
- (33) proposed a novel joint extraction model with two independent token embedding modules for encoding features about entities and relations, respectively. It enables the encoding of semantic representation with different granularities for NER and RE and uses a cross-attention approach to capture the interaction between them.

Additionally, we also add some previously proposed State-of-the-Art models in this field, such as CAMFF (18), TAGPRIME (44), and CRFIE (45).

4.3 Experimental Settings

We use the exact match principle to predict entities' head and tail positions for fully annotated datasets such as SciERC, ACE05, ACE04, ADE, and CoNLL04. The evaluation metric for CoNLL04, SciERC, ACE2004, and ACE2005 is the Micro-F1 score, and for ADE is the Macro-F1 score. In order to prevent model overfitting and make the trained models more accurate and credible, we perform five-fold cross-validation on the ACE2004 dataset and ten-fold cross-validation on the ADE dataset. The ALBERT (12) pre-trained language model is used for the ACE2004, ACE2005, and CoNLL04 datasets, and SciBERT (13) for the SciERC dataset. For the ADE dataset, we use both the BERT (11) and ALBERT (12) pretrained language models during training. For half-annotated datasets, we evaluate our model on two datasets, NYT and WebNLG, using the partial matching principle, where the entity task only predicts the tail position of an entity. The evaluation metric is the Micro-F1 score. The pre-trained language model is the BERT (11). All datasets are trained in the single-sentence setting in our model.

4.4 Implementation Details

Table 4.4 provides the implementation details as depicted in the figure. The parameters for DArtER are slightly different from the BiDArtER model. We set the batch size to 20 and the learning rate to 2e-5 for the NYT, WebNLG, CoNLL04, and ADE datasets. For the SciERC and ACE2005 datasets, the batch size is set to 4, and the learning rate is set to 1e-5. Regarding the ACE2004 dataset, which contains 5 folds, we customize the batch size and learning rate for each fold. The selection of hyperparameter settings is based on the test results on the validation set. To ensure a fair comparison with the baseline model, we use the BERT (11) pre-trained language model for prediction on the NYT and WebNLG datasets, employ ALBERT (12) as the pre-trained language model for CoNLL04, ACE2004, and ACE2005, and utilize the SciBERT (13) model for the SciERC dataset. The maximum input length for words is set to 128. Our model is trained on an NVIDIA Tesla P100, an NVIDIA A40, and an NVIDIA A100 for each dataset, regardless of the graphics card model.

Datasets	Model	Type	Batch Size	LR	α	β	γ	δ
NYT	DArtER	Total	20	2e-5	1	1	1	1
	BiDArtER	Total	20	2e-5	-1	1	1	1
WebNLG	DArtER	Total	20	2e-5	1	1	1	1
	BiDArtER	Total	20	2e-5	-1	1	1	1
CONLL04	DArtER	Total	20	2e-5	1	1	1	1
	BiDArtER	Total	20	2e-5	-1	1	1	1
ADE	DArtER	Total	20	2e-5	1	1	1	1
	BiDArtER	Total	20	2e-5	-1	1	1	1
SciERC	DArtER	Total	4	1e-5	1	1	1	1
	BiDArtER	Total	4	1e-5	-1	1	1	1
ACE2004	DArtER	0	20	2e-5	1	1	0.75	1
		1	4	1e-5	1	1	1	1
		2	4	1e-5	1	1	0.75	1
		3	4	1e-5	1	1	1	1
		4	4	1e-5	1	1	1	1
	BiDArtER	0	4	1e-5	1	1	0.75	1
		1	8	1.5e-5	-1	1	1	1
		2	4	1e-5	-1	1	1	1
		3	4	1e-5	-1	1	0.75	1
		4	4	1e-5	-1	1	1	1
ACE2005	DArtER	total	4	1e-5	1	1	0.85	1
	BiDArtER	total	4	1e-5	-1	1	0.75	1

Table 4.4: Parameter settings in our experiments. α and β are the fusing parameters between subjects and objects in the decoder. γ and δ are the loss weights for the NER and RE tasks in the training process.

4.5 Results and Analysis

Table 4.5, 4.6, 4.7, 4.8, and 4.9 present comparisons of our proposed model with previous related approaches on five fully-annotated public datasets. On the ACE2004 dataset, our model outperforms the best results by +0.1%/+2.7% in NER and RE tasks. On the ACE2005 dataset, our model performs slightly weaker than the BR model in the NER task (-1.0%) but achieves a higher score of +0.1% in the RE task. On the ADE dataset, when using BERT (11) as the pre-trained language model, our model shows a slight decrease of

Method	**PLM**	**NER**	**RE**
Table-Sequence (10)	ALBERT	88.6	59.6
PURE (9)	ALBERT	88.8	60.2
UNIRE (17)	ALBERT	89.5	63.0
PFN (5)	ALBERT	89.3	62.5
TAGPRIME (44)	ALBERT	89.0	62.3
BR(27)	ALBERT	88.7	62.3
DArtER	ALBERT	**89.6**	64.6
BiDArtER	ALBERT	89.3	**65.7**

Table 4.5: Comparison of the proposed model with the prior works on the ACE2004 dataset.

Method	**PLM**	**NER**	**RE**
Table-Sequence (10)	ALBERT	89.5	64.3
PURE (9)	ALBERT	89.7	65.6
UNIRE (17)	ALBERT	90.2	66.0
PFN (5)	ALBERT	89.0	66.8
TablERT (15)	ALBERT	89.8	65.2
TAGPRIME (44)	ALBERT	89.6	68.1
CRFIE (45)	ALBERT	90.1	68.3
BR(27)	ALBERT	**90.8**	66.0
DArtER	ALBERT	89.5	68.3
BiDArtER	ALBERT	89.8	**68.4**

Table 4.6: Comparison of the proposed model with the prior works on the ACE2005 dataset.

Method	**PLM**	**NER**	**RE**
Table-Sequence (10)	ALBERT	89.7	80.1
PFN (5)	BERT	89.6	80.0
PFN (5)	ALBERT	91.3	83.2
IEER (34)	BERT	90.1	82.5
(33)	ALBERT	91.6	83.7
BR (27)	BERT	**91.0**	**82.9**
BR (27)	ALBERT	91.7	84.8
DArtER	BERT	90.3	82.0
BiDArtER	BERT	90.6	82.5
DArtER	ALBERT	**92.3**	**85.4**
BiDArtER	ALBERT	92.2	**85.4**

Table 4.7: Comparison of the proposed model with the prior works on the ADE dataset.

Method	**PLM**	**NER**	**RE**
PURE (9)	SciBERT	66.6	35.6
UNIRE (17)	SciBERT	68.4	36.9
PFN (5)	SciBERT	66.8	38.4
UIE (14)	T5-v1.1-large	-	36.53
CAMFF (18)	SciBERT	68.9	-
IEER (34)	BERT	68.4	**40.0**
DArtER	SciBERT	69.1	39.1
BiDArtER	SciBERT	**69.4**	39.9

Table 4.8: Comparison of the proposed model with the prior works on the SciERC dataset.

Method	**PLM**	**NER**	**RE**
Table-Sequence (10)	ALBERT	90.1	73.6
PFN (5)	ALBERT	89.6	75.0
TablERT (15)	ALBERT	89.7	73.7
(33)	ALBERT	90.2	74.4
BR(27)	ALBERT	**90.3**	74.9
DArtER	ALBERT	89.6	75.3
BiDArtER	ALBERT	89.7	**75.6**

Table 4.9: Comparison of the proposed model with the prior works on the CoNLL04 dataset.

-0.4% compared to the previous state-of-the-art model BR in both NER and RE tasks. However, when using ALBERT (12) as the pre-trained language model, our model surpasses the previous highest score by +0.6% in both NER and RE tasks, respectively. Additionally, on the SciERC dataset, our model demonstrates good accuracy in the NER task with an improvement of +0.5% but decreases slightly in the RE task by -0.1%. On the CoNLL04 dataset, our model performs slightly weaker than the BR model in the NER task by -0.6% but achieves a higher score of +0.2% in the RE task.

By analyzing the experimental results, we can draw the following two conclusions. (1). Compared with the models that are mainly based on constructing distinct subtasks of NER and RE, e.g., (34) and (27), our approach proposes to consider the different impact of the subject and the object on their relations. Our model significantly improves the results on these five datasets. In particular, many subjects and objects in the ACE2004, AcE2005, ADE, and SciERC datasets are complex entities. Treating both the complex and the normal entities as the same entities in feature representations weakens the performance of the RE task. Our model facilitates the differentiation of semantic differences between subjects and objects by encoding fine-grained relational triple information, thus better-defining relation types.

(2). Regarding information interaction, previous models, either parameter sharing (27), or the shared features (10; 5), or the mutual information interaction of NER and RE (33; 34), do not consider the information interaction among subjects, relations, and objects. Our model builds three subtasks to construct differentiated task-specific features of subjects, objects, and relations and enhance their mutual interaction. The experimental results demonstrate that fine-grained information interaction can improve task recognition.

Method	PLM	NYT		WebNLG	
		NER	**RE**	**NER**	**RE**
TpLinker (26)	BERT	-	91.9	-	91.9
TDEER (4)	BERT	-	92.5	-	93.1
PFN (5)	BERT	95.8	92.4	98.0	93.6
RIFRE (31)	BERT	-	92.0	-	92.6
PRGC (32)	BERT	-	92.6	-	93.0
IEER (34)	BERT	-	-	98.1	**94.1**
(33)	BERT	-	**93.0**	-	91.2
DArtER	BERT	95.8	92.4	**98.2**	93.7
BiDArtER	BERT	**95.9**	92.6	98.1	**94.1**

Table 4.10: Comparison of the proposed model with the prior works on the NYT and WebNLG datasets.

To evaluate our model's effectiveness on datasets containing more overlapping relations, we conduct experiments on two half-annotated datasets: NYT and WebNLG. Overall, compared with the five fully-annotated datasets above, the performance improvement of our model is relatively small. However, compared with the models (5) that focus on constructing the sharing features, we demonstrate the effectiveness of building differentiated task-specific features for subjects, objects, and relations. Compared with the model (33) that is built with differentiated features of NER and RE tasks, our model is slightly weaker in the RE task on the NYT dataset by -0.4%, but outperforms it by +2.9% in the WebNLG dataset. This may be because the NYT dataset differs from the five fully annotated datasets above in that it contains many normal-type entities with relatively few complex entities. So, there is little reliance on constructing differentiated task-specific features for subjects and objects.

4.6 Ablation Study

We conducted an ablation study (see Table 4.11) to assess the contribution of each component in our model on the SciERC dataset. For this purpose, we performed experiments on a subset of the data using the following options:

Ablation	Settings	NER	RE
Number of Layers	N=1	69.1	39.1
	N=2	**69.4**	**39.9**
	N=3	68.9	38.4
	N=4	69.2	39.0
Information Interaction	F=Y	**69.1**	**39.1**
	F=N	68.1	38.1
Encoder Strategy	DAM	**69.1**	**39.1**
	LSTM	68.7	38.8
	PFN (5)	66.8	38.4
Decoding Strategy	RE+NER	**69.1**	**39.1**
	RE	68.0	37.0

Table 4.11: Ablation study results.

4.6.1 Number of the DAM Encoder Layers

We conducted experiments on different DAM encoder layers in the ablation study. The first layer represents the left-to-right encoder, and when the second layer is added, the direction becomes right-to-left. For three layers and four layers, we followed the same rules as for one layer and two layers. We tested up to four layers on the SciERC dataset. Results shown in Table 4.11 indicate that the two-layer model performs better than the one-layer model. However, the three-layer and four-layer models perform worse than the lower-layer models. This may be attributed to the increase in dimensionality of the relation features as the number of layers increases. Therefore, important information may be lost when using sigmoid to compress high-dimensional features.

4.6.2 Bidirection VS Unidirection

To determine whether the bidirectional model outperforms the unidirectional model, we conducted testing using a two-layer network. The unidirectional network encodes sentences from left to right in both layers. In contrast, the bidirectional network considers information from both directions during encoding. As shown in Table 4.11, the extraction accuracy of the BiDArtER model is generally higher than that of the DArtER model. This indicates that the bidirectional encoder can capture more semantic information, thereby facilitating the extraction of more accurate entities and relations.

4.6.3 Information Interaction VS No Information Interaction

To evaluate the importance of the encoder modules for semantic information aggregation of different task-specific cells, we conducted an experiment where we removed the inter-aggregated features in the encoder. In terms of the relation decoder, we removed the aggregated entity features. After making these modifications, we performed experiments and obtained the subsequent results. The results indicate that the model with the aggregating strategy (F=Y) performs better than the one without the aggregating schema. It demonstrates that information interaction among different subtasks can help build differentiated task-specific features and information transfer.

4.6.4 Encoder Strategy

To evaluate the effectiveness of the DAM module and the decoupling strategy used to construct fine-grained task-specific features in the encoding phase, we employed three LSTM models to replace the three subtasks to construct task-specific features of the subjects, relations, and objects, respectively. The LSTM-based model lacks information aggregation in the encoding phase, while the decoder part remains unchanged. The experimental results show that our model performs better in entities (+0.4%) and relations (+0.3%) than the LSTM-based model. This highlights the crucial role of information interaction and aggregation in the DAM module. Additionally, compared to the baseline model (5) that builds sharing features among NER and RE, the LSTM-based model achieves a higher F1-score (+1.9% and +0.4%) for NER and RE tasks, respectively. We can draw two conclusions. (1). The impact of the subject and the object on their relations may have difference. (2). Building fine-grained task-specific features for subjects, relations, and objects can effectively improve the task prediction.

4.6.5 Decoder Strategy

We removed the entity features in the RE decoder to examine the necessity of incorporating entity features into the RE task. The experimental results show that introducing entity features improves the NER and RE tasks by +1.1% and +2.1%, respectively. This indicates that the entity features in our model help enhance the context information to improve the RE task. Since our model is a joint training model, enhancing the RE task can also contribute to the NER task.

Dataset	Type	Train	Dev	Test	Ratio(%)
SciERC	OOT	495	88	154	28.0
	IT	1366	187	397	72.0
ACE2004	OOT	4276	727	1250	72.0
	IT	1668	276	486	28.0
ACE2005	OOT	7408	1793	1453	70.9
	IT	2643	631	597	29.1

Table 4.12: Statistics of OOT and IT sentences.

Datasets	Model	PLM	OOT			IT		
			P	**R**	**F1**	**P**	**R**	**F1**
SciERC	**PFN** (5)	SciBERT	53.9	65.7	59.2	66.9	69.5	68.2
	DArtER	SciBERT	52.7	66.1	58.7	**70.2**	**71.8**	**71.0**
	BiDArtER	SciBERT	**58.3**	**69.5**	**63.4**	69.6	71.0	70.3
ACE2004	**PFN** (5)	ALBERT	**87.4**	89.0	**88.2**	90.3	90.0	90.1
	DArtER	ALBERT	87.1	**89.4**	**88.2**	**91.3**	**91.0**	**91.2**
	BiDArtER	ALBERT	86.9	88.85	87.9	90.3	90.5	**91.2**
ACE2005	**PFN** (5)	ALBERT	**85.8**	86.1	85.9	91.5	90.4	91.0
	DArtER	ALBERT	**85.8**	86.9	86.3	**91.6**	91.3	**91.5**
	BiDArtER	ALBERT	85.2	**87.7**	**86.4**	90.5	**92.1**	91.3

Table 4.13: NER Results on In-triple (IT) and Out-of-triple (OOT) sentences.

4.6.6 NER Performance on Different Sentence Types

For the SciERC, ACE2004, and ACE2005 datasets, which include both Out-of-triple (OOT) and In-triple (IT) sentences, as discussed in (5), we conducted the same experiment to test the model's performance on different types of sentences. OOT sentences refer to sentences that do not contain triples, while IT sentences represent sentences with triples. The statistics of sentence count in the train, dev, and test sets for SciERC, ACE2004, and ACE2005 datasets are shown in Table 4.12. The sentence counts we obtained differ slightly from those reported in the PFN model paper, so the scores reported for the PFN model (5) are the ones we retested.

The results are presented in Table 4.13. In the case of OOT sentences, our model achieves a higher F1 score on SciERC and ACE 2005 datasets while performing comparably to the baseline model on the ACE2004 dataset. For IT sentences, our model outperforms the baseline on all datasets. However, the DArtER model performs slightly lower for OOT sentences, with a decrease of -0.5% on the SciERC dataset. The BiDArtER model performs slightly lower than the baseline model by -0.3% on the ACE2004 dataset. We speculate that because the original dataset contains a small portion of OOT sentences, it may not be conducive to our model's training and parameter updating based on triple interactions. Moreover, it may be ineffective for constructing fine-grained features for subjects and objects since OOT sentences do not contain relations. That is why our model outperforms the baseline model on all datasets in the case of IT sentences. We can draw two conclusions compared with the baseline model. (1). Regarding the results on IT sentences, we can demonstrate that building fine-grained task-specific features of subjects, relations, and objects and enabling task interaction are conducive to the NER task. (2) With more fine-grained task interactions, the RE task is more helpful for the NER task.

4.7 Error Analysis

To investigate the factors that influence the extraction of entity types and relation types in our model, we analyze the performance of jointly predicting different elements of the entity and triple ($\langle E, E_t \rangle$, $\langle S, R, O \rangle$) on ACE2005 dataset. $\langle E, E_t \rangle$ represents the entity E with its type E_t, $\langle S, R, O \rangle$ represents the relational triple with the subject S, the relation R, and the object O. Each type of error is shown in Table 4.14.

For NER, we divided it into three types: ET indicates that the entity span and type are predicted correctly. EN means that the entity span is correctly predicted, but the entity type is incorrectly predicted. ET_NP means that the entity is presented in the gold label but not predicted.

For RE, there are three types: SOR indicates that the head positions of the subject and object entities and the relation type are predicted correctly. SON means that the head positions of the subject and object entities are correct, but their relation type is incorrectly predicted. SOR_NP indicates that the relational triples existing in the gold label are not predicted.

For joint prediction, there are two cases: ETSOR indicates that both the span and type of the subject and object entities are predicted correctly, and the relation triples are also predicted correctly. ETSON indicates that the entity span and type are predicted correctly, and the head position of the subject entity and the object entity in the relational triple are predicted correctly, but their relation type is mispredicted.

Type	**E**	$\mathbf{E_t}$	$\langle$**S, O**$\rangle$	**R**	**Model**	**Predicted Numbers**
ET	✔	✔	-	-	**PFN** (5)	4443
					Our Model	4510
EN	✔	✗	-	-	**PFN** (5)	289
					Our Model	256
ET_NP	✗	✗	-	-	**PFN** (5)	438
					Our Model	404
SOR	-	-	✔	✔	**PFN** (5)	708
					Our Model	727
SON	-	-	✔	✗	**PFN** (5)	46
					Our Model	37
SOR_NP	✗	✗	✗	✗	**PFN** (5)	393
					Our Model	383
ETSOR	✔	✔	✔	✔	**PFN** (5)	676
					Our Model	691
ETSON	✔	✔	✔	✗	**PFN** (5)	43
					Our Model	33

Table 4.14: The classification of Error study. The predicted numbers reported for the PFN model are re-implemented.

Table 4.14 displays the predicted numbers for different NER, RE, and joint prediction settings. Our model outperforms the baseline PFN model regarding entity type (ET) and relation prediction (SOR). We also exhibit fewer errors in predicting wrong entity types (EN) and relationship types (SON). When comparing the scores of ET_NP and SOR_NP, we observe that our model has lower scores, indicating a higher ability to predict entity spans and relational triples. In joint prediction, our model has ten fewer errors than the PFN model for the ETSON cases. This indicates that our model is less likely to predict the wrong relation type when the entity span and type are predicted correctly. In addition, the experimental results can also demonstrate that building fine-grained task-specific features of subjects, relations, and objects is more effective in predicting entities and relational triples.

4.8 Case Study

We conducted a case study experiment as shown in Table 4.15 and 4.16 to investigate the effectiveness of constructing fine-grained task-specific features and whether there exist the impact difference of the subject and the object on their relations. We compared our results with the baseline model PFN (5) that constructs sharing features between NER and RE tasks and does not differentiate semantic differences between subjects and objects. Thus, we chose some sentences containing complex entities for comparison.

Sentence 1 shows that our model effectively identifies the term "divisions" as referring to the type of people (PER) rather than branches of organizations. In the RE task, through information interactions of the task-specific features of the subjects and objects, our model prefers the "ORG-AFF" relation type over the "PART-WHOLE" relation type. This indicates that the subject and the object have different impacts on their relations. The type of subject "divisions" has a more significant impact on the RE task than the object "Republican".

Sentence 2 reveals that our model and the baseline model correctly determine the type of "people" as "PER". However, when extracting relations, our model can build fine-grained task-specific features for the subject "people" and the object "United States", then aggregate them into the RE task to enhance the prediction of their type of "GEN-AFF" (e.g., citizen, resident) instead of "ORG-AFF". Another relational triple (it, ART (e.g., owner, manufacturer), nukes) is also predicted in our results. In this triple, all the entities and their types are extracted correctly in both models, but the baseline model does not predict their relations. Thus, constructing task-specific features of subjects and objects separately can help determine their relation types. Furthermore, sentences 3, 4, and 5 show the ability of our model to extract entities' span, determine their types, and predict their relations.

Overall, we can draw three conclusions as follows: (1). when the entity is predicted correctly, our model is more likely to predict their relations. (2). Due to the presence of complex entities, which have different effects on relations than normal types of entity words, we need to encode them at a fine-grained level to ensure semantic differences. (3). Information interaction among task-specific features of subjects, relations, and objects contributes to the RE task.

Sentence 1: As shown in Table 1.1
(PFN) Entities: (Pentagon, ORG), (they, ORG), (divisions, **ORG**), (Republican Guard, ORG), (U.S, GPE), (troops, PER), (Karbala, GPE), (Al Kut, GPE) **Relations**: (troops, PHYS, Karbala), (troops, PHYS, Al), (divisions, **PART-WHOLE**, Republican), (troops, ORG-AFF, U.S)
(Our) Entities: (Pentagon, ORG), (they, ORG), (divisions, **PER**), (Republican Guard, ORG), (U.S, GPE), (troops, PER), (Karbala, GPE), (Al Kut, GPE) **Relations**: (troops, PHYS, Karbala), (troops, PHYS, Al), (divisions, **ORG-AFF**, Republican), (troops, ORG-AFF, U.S)
Sentence 2: North Korea has told important people of the United States that it has developed nukes and reprocessed spent fuel rods. **Entities**: (United States, GPE), (North Korea, GPE), (it, GPE), (people, PER), (nukes, WEA) **Relations**: (it, ART, nukes), (people, GEN-AFF, United States)
(PFN) Entities: (United States, GPE), (North Korea, GPE), (it, GPE), (people, PER), (nukes, WEA) **Relations**: (people, **ORG-AFF**, United States)
(Our) Entities: (United States, GPE), (North Korea, GPE), (it, GPE), (people, PER), (nukes, WEA) **Relations**: (**it, ART, nukes**), (people, **GEN-AFF**, United States)
Sentence 3: Nic, we're getting information in bits and pieces about the incursion by coalition land forces, about air flights over the city. **Entities**: (coalition, GPE), (Nic, PER), (we, ORG), (forces, PER), (city, GPE) **Relations**: (forces, coalition, ORG-AFF)
(PFN) Entities: (coalition, GPE), (Nic, PER), (we, **PER**), (forces, PER), (city, GPE) **Relations**: (forces, coalition, ORG-AFF)
(Our) Entities: (coalition, GPE), (Nic, PER), (we, **ORG**), (forces, PER), (city, GPE) **Relations**: (forces, coalition, ORG-AFF)

Table 4.15: Case Study of our NER and RE results. Table 4.3 shows the detailed meanings of the entity and relation types.

Sentence 4: Soldiers are here to tear down the regime and all it stands for. **Entities**: (here, GPE), (Soldiers, PER), (regime, ORG), (it, ORG) **Relations**: (Soldiers, here, PHYS)
(PFN) Entities: (Soldiers, PER), (regime, ORG), (it, ORG) **Relations**: none
(Our) Entities: (**here, GPE**), (Soldiers, PER), (regime, ORG), (it, ORG) **Relations**: (**Soldiers, here, PHYS**)
Sentence 5: a man on a motorcycle was killed while being chased about i police, the violence broke out. **Entities**: (man, PER), (motorcycle, VEH), (police, PER) **Relations**: (man, motorcycle, ART)
(PFN) Entities: (man, PER), (police, PER) **Relations**: none
(Our) Entities: (man, PER), (**motorcycle, VEH**), (police, PER) **Relations**: (**man, motorcycle, ART**)

Table 4.16: Case Study of our NER and RE results. Table 4.3 shows the detailed meanings of the entity and relation types.

4.9 Analysis on Different Relation Types

In addition, our model exhibits weaker performance improvements on the NYT dataset than other datasets. This section aims to explore the possible reasons behind this observation. We speculate that our model may heavily rely on the semantic interactions to construct NER and RE features. Consequently, semantically rich words in the pre-trained language model, such as city names and country names, may contain a more informative semantic context, leading to relatively more accurate predictions of the corresponding relations. On the other hand, long-tail words with limited semantic information, such as common person names, may not be as well-predicted. Thus, we conduct an experiment to test the RE performance of our model in different relational types. We tested on three datasets. For the ACE2004 and ACE2005 datasets, we calculated the F1 scores for relation extraction in each relation type. For the NYT dataset, we divided the sentences into two subsets: one subset contained relation types with the start word "/people/*", while the other subset did not. The relation types with a start word of "/people/*" contained more entities of people's names, and we believe that

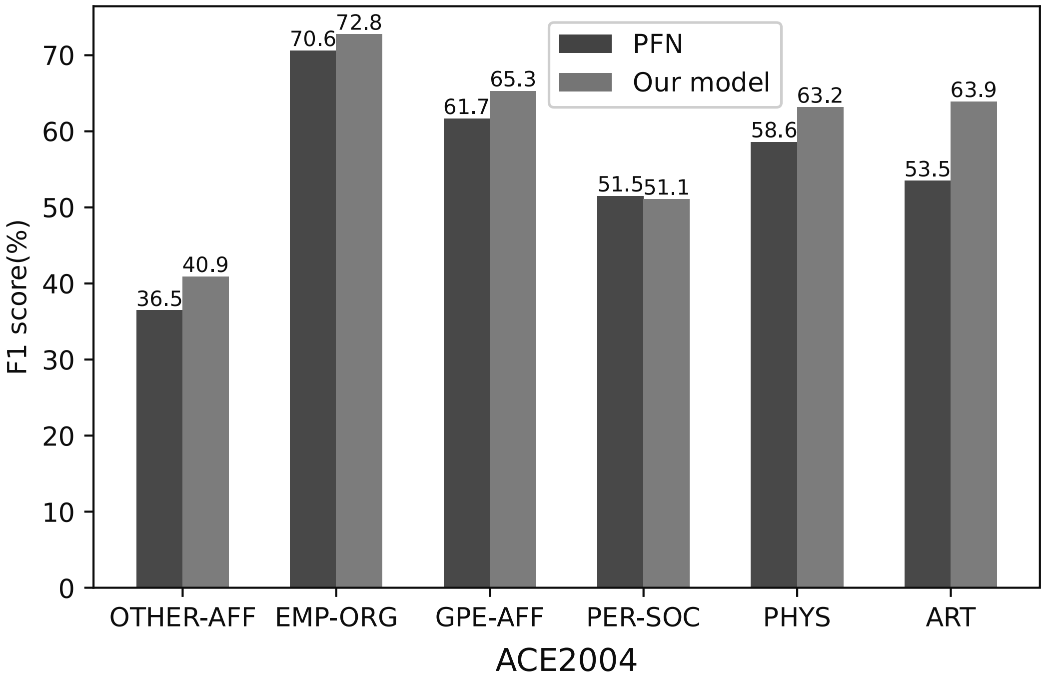

Figure 4.1: Comparison of different relational types on ACE2004 dataset. The scores reported for the PFN model (5) are re-implemented.

extracting their relation types is relatively more challenging. Table 4.2 shows the detailed statistics of the different entity and relation types.

Figure 4.1, 4.2 and 4.3 compares the experimental results between the baseline and our models. On the ACE2004 dataset, our model is -0.4% lower than the baseline model for the type "PER-SOC" (business, family, other). On the ACE2005 dataset, our model scores lower in the case of the "OTHER-AFF" (ethnic, ideology, other) type and has the same score in the "PER-SOC" type. The NYT dataset also exhibits relatively low scores for the "PEOPLE/" type. These results indicate that our model achieves higher accuracy in extracting relation types that are richer in semantic information. However, the extraction accuracy is relatively lower for relation triples involving long-tailed entities. The reason may be that our model interacts among NSR, RE, and NOR when building their task-specific features; the entities with rich semantic entities may override the long-tailed entities with fewer semantics during information interaction, leading to problems in feature construction. To effectively improve the extraction accuracy of these types, exploring other methods to enhance their semantic information is necessary in the future.

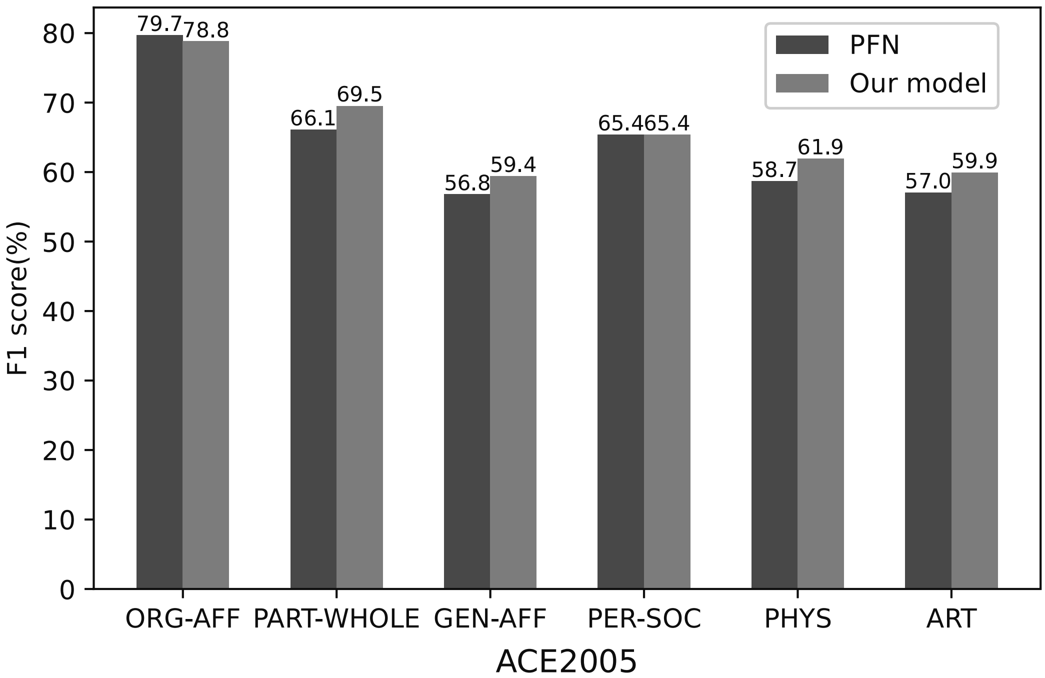

Figure 4.2: Comparison of different relational types on ACE2005 dataset. The scores reported for the PFN model (5) are re-implemented.

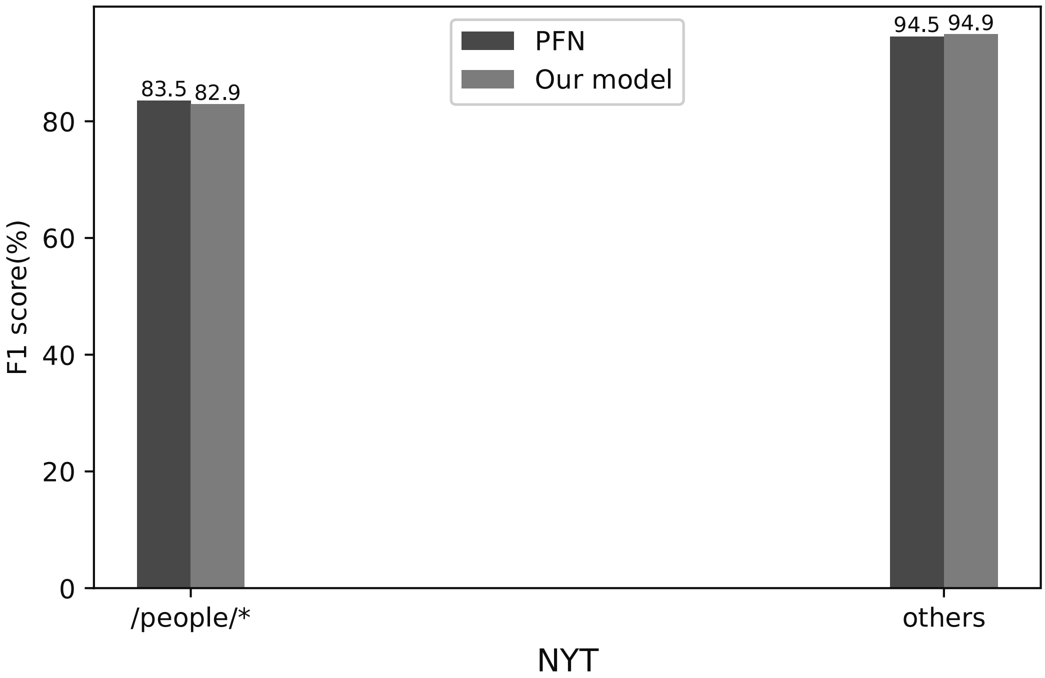

Figure 4.3: Comparison of different relational types on NYT dataset. The scores reported for the PFN model (5) are re-implemented.

4.10 Detailed Results on Overlapping Pattern and Different Triple Number

We also conducted experiments to address the overlapping problem and tested sentences containing different numbers of triples. The overlapping problem is categorized into three classes: normal sentences, SEO sentences, and EPO sentences. Specific examples and introductions can be found in (25). As shown in Table 4.17, Our model outperforms the baseline models by +1.1% and +0.5% for the normal and SEO classes on the NYT dataset. However, our model performs slightly worse by -0.6% on EPO sentences. On the WebNLG dataset, our model increases by +0.6% and +0.1% for the SEO and EPO classes, respectively. However, our model's performance is slightly lower by -0.1% for the normal class.

Our model performs better than the baseline models in several categories for sentences with different numbers of triples (see Table 4.18). On the NYT dataset, our model achieves higher scores by +0.3% and +0.5% on the N=2 and N=3 types of sentences, respectively. However, our model's performance is slightly lower by -0.8% and -0.5% on the N=1 and N≥5 types of sentences. On the WebNLG dataset, our model outperforms the baseline models by +0.1%, +0.9%, and +1.3% on the N=1, N=3, and N≥5 classes, respectively. However, our model's performance is -0.1% and -0.8% lower on the N=2 and N=4 classes.

Even though our model is not specifically designed to address overlapping problems, our model has relatively higher scores on several types compared with the baseline models. However, the improvement of our model is not significant. Thus, we plan to design other methods to address these issues further in future work.

Method	NYT			WebNLG		
	Normal	SEO	EPO	Normal	SEO	EPO
TDEER (4)	90.8	94.1	94.5	90.7	93.5	95.4
RIFRE (31)	90.7	93.2	93.5	90.1	93.1	94.7
PFN (5)	90.2	94.1	**95.3**	**91.6**	94.0	94.7
PRGC (32)	91.0	94.0	94.5	90.4	93.6	95.9
Our Mdoel	**92.1**	**94.6**	94.7	91.5	**94.6**	**96.0**
Gap with baseline models	**+1.1**	**+0.5**	-0.6	-0.1	**+0.6**	**+0.1**

Table 4.17: F1 score of predicting relational triples on overlapping problems

Method	NYT					WebNLG				
	N=1	N=2	N=3	N=4	N≥5	N=1	N=2	N=3	N=4	N≥5
TDEER (4)	90.8	92.8	94.1	95.9	92.8	90.5	**93.2**	94.6	93.8	92.3
RIFRE (31)	90.7	92.8	93.4	94.8	89.6	90.2	92.0	94.8	93.0	92.0
PFN (5)	90.5	92.9	93.7	**96.3**	92.6	91.3	92.4	95.6	94.7	93.3
PRGC (32)	**91.1**	93.0	93.5	95.5	**93.0**	89.9	91.6	95.0	**94.8**	92.8
Our Mdoel	90.3	**93.3**	**94.6**	**96.3**	92.5	**91.4**	93.1	**96.5**	94.0	**94.6**
Gap with baseline models	-0.8	**+0.3**	**+0.5**	0	-0.5	**+0.1**	-0.1	**+0.9**	-0.8	**+1.3**

Table 4.18: F1 score of predicting relational triples on sentences with different number of triples

Chapter 5

Conclusion

5.1 Summary

This work proposes a new joint model that enables efficient interaction among triple subtasks. Our approach recognizes the difference in the impact of the subject and the object on their relations when either the subject or the object is a complex entity. By decoupling the extraction task and aggregating information, we construct the fine-grained task-specific features and enhance the information interaction among each sub-task. Experimental results validate the effectiveness of our approach. Our contributions are summarized as follows:

- We propose a novel joint extraction model of entities and relations. Our model leverages three subtasks of extracting the subject, the object, and their corresponding relation to build their differentiated features and propose an aggregating strategy to enable fine-grained information interaction among each subtask-specific feature, addressing the previous limitations of (1) ignoring the possibility of the impact difference between the subject and the object on their relation when either of them is a complex entity; (2) information interaction mainly based on the subtasks of extracting entities and relations. Additionally, we also design a *BiDArtER* model that can capture richer context semantics of each word in a bi-directional way.

- Our model outperforms several state-of-the-art models. In specific, we increase the accuracy score by +2.7%, +0.1%, +0.6%, and +0.6% in the relation extraction task on ACE2004, ACE2005, ADE, and CoNLL04 datasets and +0.3%, +0.6%, +0.5%, +0.1%, and +0.1% in the entity extraction task on ACE2004, ADE, SciERC, NYT, and WebNLG datasets, respectively.

- Compared with the models that are mainly based on constructing subtasks of extracting entities and relations without differentiated representation of the subject and the object, our approach proposes to consider the different impacts of the subject and the object on their relations. The experimental results demonstrate that our approach can effectively improve the task recognition ability.

- Regarding information interaction, previous models, either based on parameter and feature sharing, or on mutual interaction between subtasks of extracting entities and relations, do not consider the information interaction among the subtask features of *extracting the subject*, *the object*, and *their relations*. Our model enhances their information interaction. The experimental results demonstrate that fine-grained information interaction can improve task recognition.

- Compared with the baseline models on the case of the OOT sentences, we also verify that building differentiated featurs for the subject, the object, and their relation can improve the NER task. Moreover, with fine-grained information interaction, the RE task is more helpful for the NER task.

5.2 Future Works

There are several promising improvements and extensions to the current method for future work.

- Concerning the encoding method, since our model is a type of RNN architecture, there may be some similar limitations when dealing with long sentences, such as sequential encoding or vanishing Gradients. Thus, future works will based on the parallel encoding of a sentence, which may improve the efficiency and deal with the limitations of the RNN-based model.

- As to the entity and relation types, it is necessary to delve into more complex scenarios. For example, (1). determining the relational type when both the subject and object types are complex entities; (2). for some specialized domain datasets, where the concepts of entities and relationships are quite abstract, how to conduct effective information interaction and subtask-specific feature construction is also a worthwhile research question.

- Furthermore, there is a need to explore ways to enhance the semantics of long-tail entities, such as the names of ordinary people. Our model performs poorly on long-tail entities relative to semantically rich regular entities. We speculate that this is mainly due to the problem of insufficient semantic features. Thus, how to effectively enhance the semantics of long-tail keywords is also an important issue.

- Finally, in specific domain datasets, such as SciERC and ADE, there is still much room for improvement in the existing methods that need to be addressed.

Bibliography

[1] Zeng, X., Zeng, D., He, S., Liu, K. & Zhao, J. Extracting relational facts by an end-to-end neural model with copy mechanism. *Proceedings Of The 56th Annual Meeting Of The Association For Computational Linguistics (Volume 1: Long Papers)*. pp. 506-514 (2018)

[2] Zeng, D., Zhang, H. & Liu, Q. Copymtl: Copy mechanism for joint extraction of entities and relations with multi-task learning. *Proceedings Of The AAAI Conference On Artificial Intelligence*. **34**, 9507-9514 (2020)

[3] Chen, Y., Zhang, Y., Hu, C. & Huang, Y. Jointly extracting explicit and implicit relational triples with reasoning pattern enhanced binary pointer network. *Proceedings Of The 2021 Conference Of The North American Chapter Of The Association For Computational Linguistics: Human Language Technologies*. pp. 5694-5703 (2021)

[4] Li, X., Luo, X., Dong, C., Yang, D., Luan, B. & He, Z. TDEER: An efficient translating decoding schema for joint extraction of entities and relations. *Proceedings Of The 2021 Conference On Empirical Methods In Natural Language Processing*. pp. 8055-8064 (2021)

[5] Yan, Z., Zhang, C., Fu, J., Zhang, Q. & Wei, Z. A Partition Filter Network for Joint Entity and Relation Extraction. *Proceedings Of The 2021 Conference On Empirical Methods In Natural Language Processing*. pp. 185-197 (2021)

[6] Luan, Y., He, L., Ostendorf, M. & Hajishirzi, H. Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction. *Proceedings Of The 2018 Conference On Empirical Methods In Natural Language Processing*. pp. 3219-3232 (2018)

[7] Roth, D. & Yih, W. A Linear Programming Formulation for Global Inference in Natural Language Tasks. *Proceedings Of The Eighth Conference On Computational Natural Language Learning (CoNLL-2004)*

At HLT-NAACL 2004. pp. 1-8 (2004,5), https://aclanthology.org/W04-2401

[8] Gurulingappa, H., Rajput, A., Roberts, A., Fluck, J., Hofmann-Apitius, M. & Toldo, L. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal Of Biomedical Informatics*. **45**, 885-892 (2012)

[9] Zhong, Z. & Chen, D. A Frustratingly Easy Approach for Entity and Relation Extraction. *2021 Conference Of The North American Chapter Of The Association For Computational Linguistics: Human Language Technologies, NAACL-HLT 2021*. pp. 50-61 (2021)

[10] Wang, J. & Lu, W. Two are Better than One: Joint Entity and Relation Extraction with Table-Sequence Encoders. *Proceedings Of The 2020 Conference On Empirical Methods In Natural Language Processing (EMNLP)*. pp. 1706-1721 (2020)

[11] Devlin, J., Chang, M., Lee, K. & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings Of The 2019 Conference Of The North American Chapter Of The Association For Computational Linguistics: Human Language Technologies, Volume 1 (Long And Short Papers)*. pp. 4171-4186 (2019,6), https://aclanthology.org/N19-1423

[12] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P. & Soricut, R. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *International Conference On Learning Representations*. (2019)

[13] Beltagy, I., Lo, K. & Cohan, A. SciBERT: A Pretrained Language Model for Scientific Text. *Proceedings Of The 2019 Conference On Empirical Methods In Natural Language Processing And The 9th International Joint Conference On Natural Language Processing (EMNLP-IJCNLP)*. pp. 3615-3620 (2019)

[14] Lu, Y., Liu, Q., Dai, D., Xiao, X., Lin, H., Han, X., Sun, L. & Wu, H. Unified Structure Generation for Universal Information Extraction. *Proceedings Of The 60th Annual Meeting Of The Association For Computational Linguistics (Volume 1: Long Papers)*. pp. 5755-5772 (2022)

[15] Ma, Y., Hiraoka, T. & Okazaki, N. Named entity recognition and relation extraction using enhanced table filling by contextualized representations. *Journal Of Natural Language Processing*. **29**, 187-223 (2022)

[16] Fu, T., Li, P. & Ma, W. Graphrel: Modeling text as relational graphs for joint entity and relation extraction. *Proceedings Of The 57th Annual Meeting Of The Association For Computational Linguistics*. pp. 1409-1418 (2019)

[17] Wang, Y., Sun, C., Wu, Y., Zhou, H., Li, L. & Yan, J. UniRE: A Unified Label Space for Entity Relation Extraction. *Proceedings Of The 59th Annual Meeting Of The Association For Computational Linguistics And The 11th International Joint Conference On Natural Language Processing (Volume 1: Long Papers)*. pp. 220-231 (2021)

[18] Yang, Z., Ma, J., Chen, H., Zhang, J. & Chang, Y. Context-aware attentive multilevel feature fusion for named entity recognition. *IEEE Transactions On Neural Networks And Learning Systems*. (2022)

[19] Crone, P. Deeper task-specificity improves joint entity and relation extraction. *ArXiv Preprint ArXiv:2002.06424*. (2020)

[20] Eberts, M. & Ulges, A. Span-based joint entity and relation extraction with transformer pre-training. *ArXiv Preprint ArXiv:1909.07755*. (2019)

[21] Riedel, S., Yao, L. & McCallum, A. Modeling relations and their mentions without labeled text. *Machine Learning And Knowledge Discovery In Databases: European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part III 21*. pp. 148-163 (2010)

[22] Gardent, C., Shimorina, A., Narayan, S. & Perez-Beltrachini, L. Creating training corpora for nlg micro-planning. *55th Annual Meeting Of The Association For Computational Linguistics (ACL)*. (2017)

[23] Walker, C., Strassel, S., Medero, J. & Maeda, K. ACE 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*. **57** pp. 45 (2006)

[24] Mitchell, A., Strassel, S., Huang, S. & Zakhary, R. Ace 2004 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*. **1** pp. 1-1 (2005)

[25] Wei, Z., Su, J., Wang, Y., Tian, Y. & Chang, Y. A Novel Cascade Binary Tagging Framework for Relational Triple Extraction. *Proceedings Of The 58th Annual Meeting Of The Association For Computational Linguistics*. pp. 1476-1488 (2020)

[26] Wang, Y., Yu, B., Zhang, Y., Liu, T., Zhu, H. & Sun, L. TPLinker: Single-stage joint extraction of entities and relations through token pair linking. *ArXiv Preprint ArXiv:2010.13415*. (2020)

[27] Tang, R., Chen, Y., Qin, Y., Huang, R. & Zheng, Q. Boundary regression model for joint entity and relation extraction. *Expert Systems With Applications*. **229** pp. 120441 (2023)

[28] Dixit, K. & Al-Onaizan, Y. Span-Level Model for Relation Extraction. *Proceedings Of The 57th Annual Meeting Of The Association For Computational Linguistics*. pp. 5308-5314 (2019)

[29] Sun, K., Zhang, R., Mensah, S., Mao, Y. & Liu, X. Recurrent Interaction Network for Jointly Extracting Entities and Classifying Relations. *Proceedings Of The 2020 Conference On Empirical Methods In Natural Language Processing (EMNLP)*. pp. 3722-3732 (2020)

[30] Wu, H. & Shi, X. Synchronous dual network with cross-type attention for joint entity and relation extraction. *Proceedings Of The 2021 Conference On Empirical Methods In Natural Language Processing*. pp. 2769-2779 (2021)

[31] Zhao, K., Xu, H., Cheng, Y., Li, X. & Gao, K. Representation iterative fusion based on heterogeneous graph neural network for joint entity and relation extraction. *Knowledge-Based Systems*. **219** pp. 106888 (2021)

[32] Zheng, H., Wen, R., Chen, X., Yang, Y., Zhang, Y., Zhang, Z., Zhang, N., Qin, B., Ming, X. & Zheng, Y. PRGC: Potential Relation and Global Correspondence Based Joint Relational Triple Extraction. *Proceedings Of The 59th Annual Meeting Of The Association For Computational Linguistics And The 11th International Joint Conference On Natural Language Processing (Volume 1: Long Papers)*. pp. 6225-6235 (2021,8), https://aclanthology.org/2021.acl-long.486

[33] Tang, R., Chen, Y., Huang, R. & Qin, Y. Enhancing interaction representation for joint entity and relation extraction. *Cognitive Systems Research*. **82** pp. 101153 (2023)

[34] Guo, W., Li, S., Liu, Y., Fan, X. & Hu, M. Information Enhancement for Joint Extraction of Entity and Relation. *Proceedings Of The 2023 7th International Conference On Innovation In Artificial Intelligence*. pp. 172-177 (2023)

[35] Machi, K., Akiyama, S., Nagata, Y. & Yoshioka, M. OSPAR: A Corpus for Extraction of Organic Synthesis Procedures with Argument Roles. *Journal Of Chemical Information And Modeling*. **63**, 6619-6628 (2023)

[36] Shamsabadi, A., Ramezani, R., Farsani, H. & Nematbakhsh, M. Direct relation detection for knowledge-based question answering. *Expert Systems With Applications*. **211** pp. 118678 (2023)

[37] Hu, Z. & Ma, X. A novel neural network model fusion approach for improving medical named entity recognition in online health expert question-answering services. *Expert Systems With Applications*. **223** pp. 119880 (2023)

[38] Gan, L., Ye, B., Huang, Z., Xu, Y., Chen, Q. & Shu, Y. Knowledge graph construction based on ship collision accident reports to improve maritime traffic safety. *Ocean Coastal Management*. **240** pp. 106660 (2023)

[39] Wu, X., Duan, J., Pan, Y. & Li, M. Medical knowledge graph: Data sources, construction, reasoning, and applications. *Big Data Mining And Analytics*. **6**, 201-217 (2023)

[40] Liu, X., Wu, K., Liu, B. & Qian, R. HNERec: Scientific collaborator recommendation model based on heterogeneous network embedding. *Information Processing Management*. **60**, 103253 (2023)

[41] Xia, L., Liang, Y., Leng, J. & Zheng, P. Maintenance planning recommendation of complex industrial equipment based on knowledge graph and graph neural network. *Reliability Engineering System Safety*. **232** pp. 109068 (2023)

[42] Li, Q., Yao, N., Zhou, N., Zhao, J. & Zhang, Y. A Joint Entity and Relation Extraction Model based on Efficient Sampling and Explicit Interaction. *ACM Transactions On Intelligent Systems And Technology*. (2023)

[43] Gormley, M., Mo, Y. & Dredze, M. Improved relation extraction with Feature-rich Compositional embedding models. *Conference On Empirical Methods In Natural Language Processing, EMNLP 2015*. pp. 1774-1784 (2015)

[44] Hsu, I., Huang, K., Zhang, S., Cheng, W., Natarajan, P., Chang, K. & Peng, N. TAGPRIME: A unified framework for relational structure extraction. *Proceedings Of The 61st Annual Meeting Of The Association*

For Computational Linguistics (Volume 1: Long Papers). pp. 12917-12932 (2023)

[45] Jia, Z., Yan, Z., Han, W., Zheng, Z. & Tu, K. Modeling Instance Interactions for Joint Information Extraction with Neural High-Order Conditional Random Field. *Proceedings Of The 61st Annual Meeting Of The Association For Computational Linguistics (Volume 1: Long Papers)*. pp. 13695-13710 (2023)

[46] Chen, Z. & Guo, C. A pattern-first pipeline approach for entity and relation extraction. *Neurocomputing*. **494** pp. 182-191 (2022)

[47] Shen, L., He, R. & Huang, S. Entity alignment with adaptive margin learning knowledge graph embedding. *Data Knowledge Engineering*. **139** pp. 101987 (2022)

[48] Yu, Y., Li, H., Shi, H., Li, L. & Xiao, J. Question-guided feature pyramid network for medical visual question answering. *Expert Systems With Applications*. **214** pp. 119148 (2023)

[49] Yang, J., Yang, X., Li, R., Luo, M., Jiang, S., Zhang, Y. & Wang, D. BERT and hierarchical cross attention-based question answering over bridge inspection knowledge graph. *Expert Systems With Applications*. pp. 120896 (2023)

[50] Zhou, K., Qiao, Q., Li, Y. & Li, Q. Improving distantly supervised relation extraction by natural language inference. *Proceedings Of The AAAI Conference On Artificial Intelligence*. **37**, 14047-14055 (2023)