Joint Entity and Relation Extraction Based on Bidirectional Update and Long-Term Memory Gate Mechanism

Yili Qian * **Enlong Ren** **Haonan Xu**
School of Computer and Information Technology,
Shanxi University,Shanxi 030006
qyl@sxu.edu.cn

Abstract

Joint entity recognition and relation extraction are important tasks in natural language processing. While some previous work has recognized the importance of relation information in joint extraction, excessively focusing on relation information without utilizing entity information may lead to information loss and affect the identification of relation tuples. Additionally, ignoring the utilization of original information may result in the loss of hierarchical and semantic information, further reducing the richness of information.To address these issues, we propose a bidirectional information updating mechanism that integrates entity and relation information, iteratively fusing fine-grained information about entities and relations. We introduce a long-term memory gate mechanism to update and utilize original information using feature information, thereby enhancing the model's ability for entity recognition and relation extraction. We evaluated our approach on two Chinese datasets and achieved state-of-the-art results.

1 Introduction

Joint entity recognition and relation extraction aim to extract entities and their relationships from unstructured text. The two common methods of relation extraction are pipeline extraction and joint extraction. Pipeline extraction (Mintz et al., 2009; Riedel et al., 2010; Chan and Roth, 2011) was a commonly used method in the early stages of relation extraction research. It divides relation extraction into two steps: first identifying entities, and then determining if there is a relationship between the identified entities. Due to the lack of interaction between the two steps, error propagation can occur. Subsequently, scholars began to research on joint extraction models (Miwa and Sasaki, 2014; Zhong and Chen, 2020; Li et al., 2022b) to address this issue. Some researchers have approached the problem from the perspective of labeling, designing appropriate labels for each word in a sentence to achieve end-to-end joint extraction (Zheng et al., 2017). Models of this kind are referred to as entity and relation extraction models based on joint decoding. However, in real-world scenarios, relation triplets are not simply one-to-one correspondences, different relations may involve entity overlaps. Joint decoding methods, due to limitations in labeling, cannot handle such complex relation triplets.

To address the extraction challenges posed by complex relationships, Zeng et al. (2018) and Zeng et al. (2020) first extract relationships, and then sequentially extract head entities and tail entities based on the extracted relationships, effectively avoiding relationship omissions. Liu et al. (2022b) divide joint extraction into two linked subtasks: the first subtask extracts head entities, and The second subtask is to extract relationships and tail entities based on the extracted head entities. This is achieved by mapping the head entity domain to the relationship and tail entity domains to enhance the model's ability to extract complex relationships. Later, researchers realized the importance of relational information and strengthened the extraction capabilities of models by integrating and utilizing relational information in various ways. Zhao et al. (2021) introduces relationship information as prior knowledge and combines it with extracted head entities to extract relationships and tail entities. Gao et al. (2022) first identifies the

* The corresponding author.

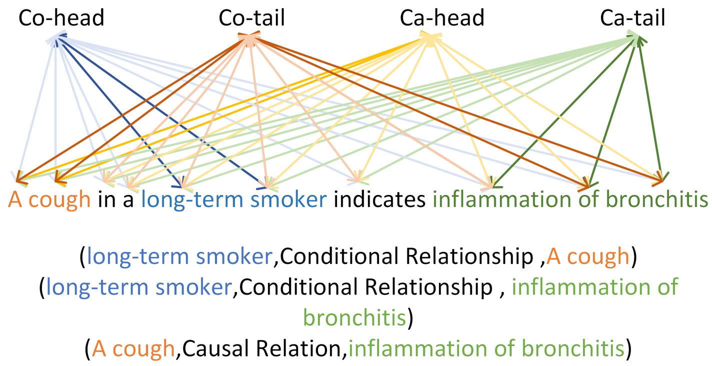

Figure 1: Example of Bidirectional Information Update. In the figure, the darker the arrow color, the greater the correlation between the word vectors and the dual-feature vectors. This shows that the richer the relation and entity information fused by the word vector. The dual-feature vectors also incorporate a wealth of word vector information, enhancing the representations of both.

relationships existing in the sentence, then randomly selects one relationship to merge into the sentence features to guide entity recognition. In order to reduce redundant information caused by complex networks, some methods (Liu et al., 2022a) use the same encoder to encode both relationship information and sentences, and achieve the fusion of the two parts of information through direct addition.

Despite the good results achieved by the aforementioned work, the following issues still exist:

- Lack of utilization of entity information. Xu et al. (2021) and Yuan et al. (2021) argue that relationships play a guiding role in triplet extraction and use attention mechanisms to update the contribution of each word in the sentence to the relationship. However, the lack of entity information leads to imbalance in word information, which is detrimental to the recognition of head/tail entities, thereby affecting the extraction of relationship tuples.
- Ignoring the original information leads to information loss. With the training of deep learning neural networks, although vector information is updated and new feature information is integrated, the original information is also altered as a result. Lack of original context and semantic information can reduce the accuracy of joint extraction.

To address the aforementioned issues, we propose a joint entity and relation extraction model based on bidirectional information updating and long-term memory gating mechanism. Firstly, we introduce entity information and relationship information as prior knowledge, represented together in a multi-dimensional vector, which we refer to as a dual-feature vector. Through an information updating mechanism, we sequentially update word vectors and dual-feature vectors. Multiple rounds of interactive updates allow for deep fusion of word feature vectors and dual-feature vectors. As shown in the Figure 1, we integrate specific relationships with head entity and tail entity information. "Co-head" represents the head entity vector of the "Conditional Relationship", while "Co-tail" represents the tail entity vector of the "Conditional Relationship". Similarly, "Ca-head" represents the head entity vector of the "Causal Relation", and "Ca-tail" represents the tail entity vector of the "Causal Relation", During the iterative fusion process with word vectors, the word vectors are enriched with rich relationship and entity information. The separate representation of head and tail entities also provides more precise entity information to the word vectors. Word vectors related to predefined information obtain enhanced representations, which are beneficial for further entity recognition and relationship extraction. Additionally, the pre-embedded dual-feature vectors also receive enhanced representations during the updating process. Subsequently, a relationship filtering module is employed to select the dual-feature vectors with the highest relevance, combining them with word feature vectors to guide the update of original word feature vectors. Finally, the combined feature vectors containing dual-feature information and original context information serve as the final word feature vectors for relation extraction.The main contributions of this work are as follows:

- We combine entity information with relationship information as prior knowledge and use an iterative information updating network to update word feature vectors, enriching them with rich entity and relationship information.
- We propose a feature-guided long-term memory gate mechanism to update the original word feature vectors containing contextual information, thereby preventing information loss.
- Our method achieves state-of-the-art results on two datasets.

2 Related Work

The traditional pipeline extraction approach (Cai et al., 2016; Zeng et al., 2014) sequentially extracts entities and relationships. Although flexible and simple, it faces two main issues. First, there is error propagation due to the independent nature of entity recognition and relationship extraction stages. Errors in entity recognition cannot be corrected during the relationship extraction stage, leading to irreversible impacts on the model's ability to accurately extract relationships. Second, there is a lack of information interaction between entity recognition and relationship extraction tasks, overlooking the inherent connection between entities and relationships. To address these issues, some scholars have proposed the concept of joint extraction to enhance the accuracy of extraction by strengthening the correlation between entity recognition and relationship extraction. Miwa and Bansal (2016) proposed an end-to-end joint extraction model where the authors used BiLSTM to extract contextual semantic information and utilized dependency syntax trees to extract features between entities. This approach achieved parameter sharing between the two subtasks, effectively alleviating the issue of error accumulation and improving extraction efficiency. However, it heavily relies on features generated by external tools and does not fully integrate the two subtasks.

Stanovsky et al. (2018) and Jia et al. (2022) utilize a sequence labeling approach to design labels, where relationship categories and entity information are designed as different labels. Each word is assigned an appropriate label and decoded based on these labels. However, since a word can only be assigned one label, the extraction efficiency of these models decreases as the complexity of relationships (such as entity overlap) increases. Takanobu et al. (2019) builds upon sequence labeling by treating entity information as parameters of the relationship. Through reinforcement learning, it enhances the interaction between entities and relationships, addressing the extraction of overlapping relationships. Liu et al. (2023a) uses a pointer network to label sentences, viewing the relationship extraction process as a mapping between two entities. It first extracts the subject in the sentence and then extracts the corresponding relationship and object based on the subject. This labeling approach solves the problem of weak expression capability in sequence labeling but may encounter label imbalance issues, requiring parameter tuning to address. Some researchers (Li and Fu, 2022; Nayak and Ng, 2020; Zeng et al., 2019) drew inspiration from machine translation, treating relation extraction as a sequence-to-sequence generation problem. They treat triplets as sequences to be translated, using a copying mechanism to find entities from the source sentence. Zhang et al. (2020) proposes the Seq2UMTree model, which addresses the label bias issue in sequence-to-sequence methods by shortening the decoding length and using tree-based decoding.

Treating relationship extraction as a table-filling problem (Wang and Lu, 2020; Kong and Xia, 2023) is also a commonly used method in joint extraction. Zhang et al. (2017), Ren et al. (2021) and Ning et al. (2023) generate a table for each relationship, labeling the start and end characters of the corresponding entities in the relationship table, and extract triplets using joint decoding. Li et al. (2022a) builds upon the table-filling approach by incorporating relationship information, strengthening global connections between same-position words in different relationship tables to achieve joint extraction.

3 Task Definition and Labeling Scheme

Given a sentence $X = [x_1, x_2, \ldots, x_m]$, where x_i is the i-th word in the sentence, the objective is to extract triples $Y = \{y_1, y_2, \ldots, y_n\}$ that match predefined relation types within the sentence. Each $y = (h, r, t)$ represents a triple, where h represents the head entity, t represents the tail entity, and r represents the relation type.In this paper, we adopt a labeling scheme similar to (Wang et al., 2020) and

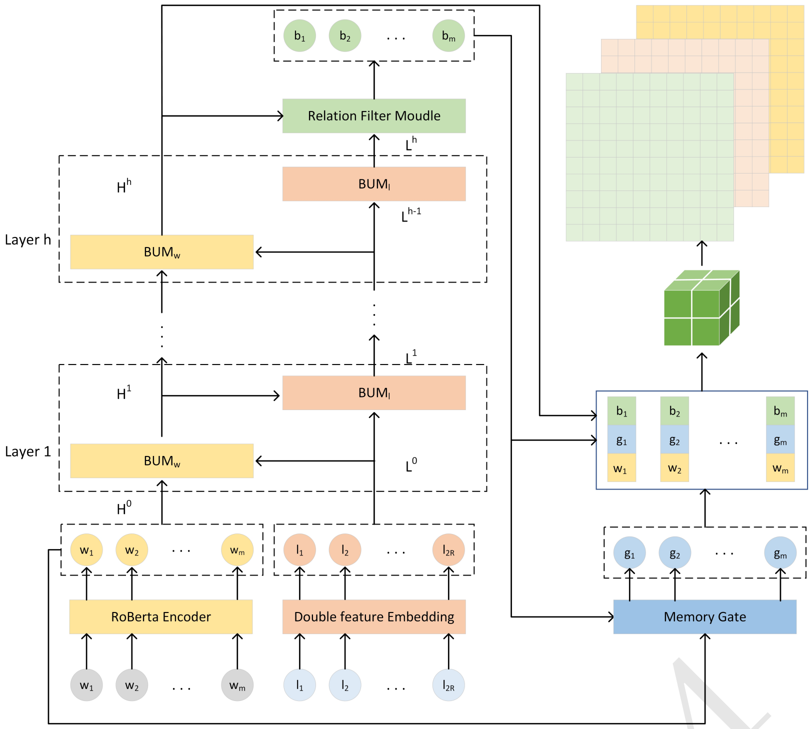

Figure 2: The overall framework of our proposed model.

design $2R + 2$ labels based on the specific characteristics of the dataset. “N” represents that the text span (w_i, w_j) does not form an entity, while “HT” represents that the text span (w_i, w_j) forms an entity. For each type of relation, two labels are generated to annotate the head entity and tail entity, “HrH” represents that the start of the head entity of the r-th relation tuple is at position w_i, and the start of the tail entity is at position w_j. “TrT” represents that the end of the head entity of the r-th relation tuple is at position w_i, and the end of the tail entity is at position w_j. Both labels together form the head and tail entities of that relation. This labeling scheme for table filling generates $2R + 1$ tables in total, with each relation assigned two tables labeled “HrH” and “TrT” respectively. A separate table is used solely to label which fields form entities, without distinguishing the type of entity.

4 Methodology

In this section, we will provide a detailed introduction to the joint entity and relation extraction model based on bidirectional updating and long-term memory gate mechanism. The framework we propose is illustrated in the Figure 2 and mainly consists of four parts:

(1) Encoding Module: Given a sentence, we encode the words in the sentence into vectors using a pre-trained model. Each relationship is combined with head/tail entities and represented using multi-dimensional vectors.

(2) Deep Fusion of Feature Information: We utilize multi-layer bidirectional updating networks to achieve deep fusion of word vectors and dual-feature vectors and ensure timely updates.

(3) Long-term Memory Gate Mechanism Guided by Most Relevant Dual-Feature Vectors: We first select the most relevant dual-feature vectors for each word in the sentence. Through a gate mechanism, we update the original word vectors to extract the original information of words, thereby avoiding the loss of semantic information.

(4) Relationship Extraction: We use the final word vectors for joint entity and relationship extraction.

4.1 Encoding Module

The pre-trained models trained on large-scale unlabeled text data have shown good performance in capturing context and semantics of sentences, and have been validated across various downstream tasks. For

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 1099-1111, Taiyuan, China, July 25 - 28, 2024.
Volume 1: Main Conference Papers

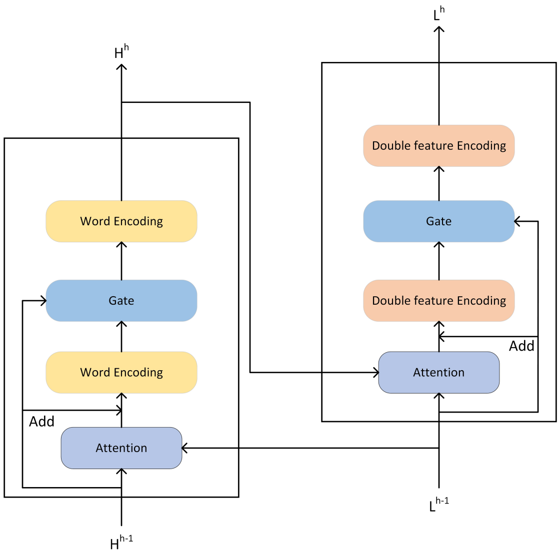

Figure 3: A complete layer of BUM.

a given sentence $X = [w_1, w_2, \ldots, w_m]$, we obtain the contextual representation of the sentence by encoding it using Chinese-Roberta-Wwm-Ext-Large (Cui et al., 2021):

$$H = \text{Roberta}(X) \tag{1}$$

where $\mathrm{H} \in \mathrm{R}^{\mathrm{m}^* \mathrm{d}}$, m as the number of characters in the sentence, and d as the dimension of each word vector obtained after encoding. We initialize the predefined dual features as high-dimensional vectors and then obtain the vector representation of each feature through a linear layer mapping:

$$L = W_1 E + b_1 \tag{2}$$

where $E \in R^{2R^*d}$ is the dual-feature vector, $L \in R^{2R^*d}$, R is the number of predefined relations, and d is the dimensionality of each dual feature vector.

4.2 Deep Fusion of Feature Information

4.2.1 Vector Updating Mechanism

We have designed a Bidirectional Updating Mechanism (BUM) to acquire and update the semantic information between words and dual features. "Bidirectional" refers to the updates of dual feature vectors to word vectors and word vectors to feature vectors. The updating process for both follows the same mechanism, and we will only introduce one of them here. As shown in the Figure 3, for a given word vector $\{h_i\}_{i=1}^{m}$ and dual feature vector $\{l_j\}_{j=1}^{2R}$, we use an attention mechanism to perform semantic fusion and update between the two. We first use w_i as the query, l_j as the key and value to update the information of w_i, and then use residual connections to prevent gradient vanishing during training:

$$\begin{gathered} a_{ij} = W_q w_i M W_k l_j^T \\ \mu_{ij} = \frac{\exp\left(a_{ij}\right)}{\sum_{s \in \mathrm{N}_i} \exp\left(a_{is}\right)} \\ w_i' = w_i + \sum_{j=1}^{2R} \mu_{ij} W_v l_j \end{gathered} \tag{3}$$

where $W_q, W_k, W_v, M \in R^{d^*d}$ are trainable weights, and μ_{ij} is the weights between $w_i \in R^d$ and $l_j \in R^d$.

We use a gate mechanism similar to Zhao et al. (2021) to control the information flow and maintain

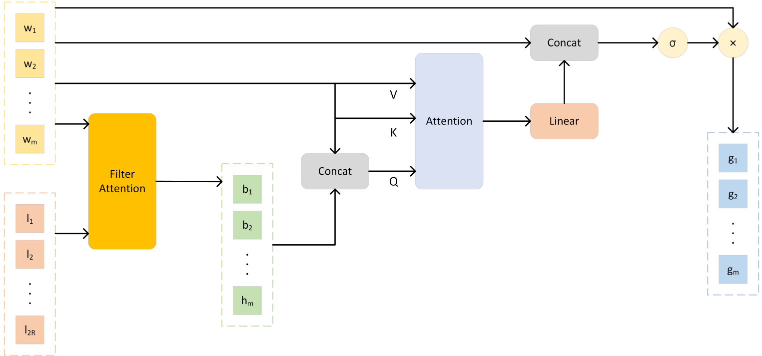

Figure 4: The detailed structure of the long-term memory gate mechanism.

nonlinear capabilities:

$$\begin{aligned}\theta_i &= \text{sigmoid}\left(W_z\left[w_i; w_i'\right]\right)\\ \tilde{w_i} &= \theta_i \odot w_i' + (1-\theta_i) \odot w_i\end{aligned} \tag{4}$$

where $W_z \in R^{d^*d}$ is trainable weights, θ_i is the retention ratio of information, $\odot$ denotes element-wise multiplication, and $\tilde{w_i}$ is the vector of words updated through the bidirectional updating mechanism.

4.2.2 Vector Iterative Fusion

We can summarize the aforementioned bidirectional update mechanism as follows:

$$\begin{aligned}\tilde{w_i} &= BUM_w\left(w_i, \{l_j\}_{j=1}^{2R}\right)\\ \tilde{l_j} &= BUM_l\left(l_j, \{w_i\}_{i=1}^{m}\right)\end{aligned} \tag{5}$$

where BUM_w and BUM_l are responsible for updating the bidirectional feature vectors to word vectors and word vectors to bidirectional feature vectors, respectively. Together, they form a complete unit called BUM. w_i is the word vectors before the update, l_j is the bidirectional feature vectors before the update, $\tilde{w_i}$ is the word vectors after the update, and $\tilde{l_j}$ is the bidirectional feature vectors after the update.

We achieve the iterative fusion of word vectors and dual feature vectors through multiple layers of BUM units.Given the sentence context representation $H = [w_1, w_2, \ldots, w_m]$ and the dual feature vector representation $L = [l_1, l_2, \ldots, l_{2R}]$, we first update the word vectors and then use the updated word vectors to update the dual feature vectors. To prevent gradient vanishing, we introduce residual connections. Therefore, we represent the update process of the h-th layer as follows:

$$\begin{aligned}w_i^h &= w_i^{h-1} + BUM_w\left(w_i^{h-1}, \left\{l_j^{h-1}\right\}_{j=1}^{2R}\right)\\ l_j^h &= l_j^{h-1} + BUM_l\left(l_j^{h-1}, \left\{w_i^{h-1}\right\}_{i=1}^{m}\right)\end{aligned} \tag{6}$$

where w_i^h is the update of the previous layer's output word vector by the h-th layer of BUM_w, and l_j^h is the update of the previous layer's output double feature vector by the h-th layer of BUM_l.

4.3 Long-Term Memory Gate Mechanism Guided by Most Relevant Double Feature Vectors

In order to reduce the interference of redundant feature vectors, we designed a feature filtering module with the aim of finding the double feature vector that has the highest relevance with each word vector. Additionally, we developed a long-term memory gate mechanism guided by the most relevant double feature vectors to update and utilize the original word vectors, preventing the loss of original contextual information and achieving long-term memory effects. The detailed structure is illustrated in the Figure 4.

4.3.1 Double Feature Encoding and Filtering Module

We believe that not all double feature vectors will contribute positively to entity recognition and relation extraction. Indiscriminate use of all double feature vectors may introduce redundant information and consequently affect the accuracy of identification. Therefore, we employ an attention mechanism to determine the most relevant double feature vectors for each word and utilize them accordingly, The specific calculation process is as follows:

$$\begin{aligned} & A_{i,j} = \operatorname{soft}\max\left(W_q w_i^h M W_k l_j'^T\right) \\ & A_{i,j}' = \begin{cases} 1, \text{ if } j = \operatorname{argmax}\left(A_i\right) \\ 0, \text{ otherwise} \end{cases} \\ & b_i = \sum A_{i,j}' l_j' \end{aligned} \tag{7}$$

where $W_q, W_k, W_v, M \in R^{d^*d}$ are the trainable weights, $A_{i,j}$ is the weights for the word vector and double feature vector, $A_{i,j}'$ is setting the position of the most relevant double feature vector for each word vector to 1 and others to 0, for subsequent computations. $b_i \in R^{1^*d}$ is the most relevant double feature vector representation for each word vector.

4.3.2 Long-Term Memory Gate Mechanism

After iterative fusion, the original context information is lost. We designed a gate mechanism based on feature encoding guidance to combine the filtered dual-feature vectors with the original word vectors, update the original word vectors, and then further filter the necessary information through the gate mechanism to reduce the loss of original information and achieve long-term memory. The specific implementation of this module is as follows:

$$\begin{aligned} & w_i' = W_c\left[W_a w_i; W_b b_i\right] \\ & w_i'' = \operatorname{soft}\max\left(W_q w_i' W_k l_j^T\right) W_v l_j \\ & w_i^{\text{gate}} = \sigma\left(W_s\left[w_i'; w_i''\right]\right) w_i' \end{aligned} \tag{8}$$

where $W_a, W_b, W_c, W_k, W_k, W_v, W_s \in R^{d^*d}$ are trainable parameters. w_i' is the combined representation of the i-th original word vector and the filtered dual-feature vector. $[;]$ denotes the concatenation operation between the two vectors. σ is the updated vector of the original word vector. w_i^{gate} is the activation function, and H is the output vector from the gate mechanism.

4.4 Relation Extraction

We combine the final word vector output from the last layer of the bidirectional updating network, the word vector output from the long-term memory gate mechanism, and the most relevant dual-feature vectors from the feature filtering module to form the ultimate word vector. This ensures that the word vector contains entity and relationship information, context, and semantic information. To maximize the full interaction between each pair of words in the sentence, we obtain the final tabular feature vector using the following computational formula:

$$\begin{aligned} & w_e = \left[w_i^h; w_i^{\text{gate}}; l_i\right] \\ & w_{i,j} = W_r \sigma\left(w_{e,i} \circ w_{e,j}\right) \end{aligned} \tag{9}$$

where w_e represents the final word vector, $\circ$ denotes the element-wise Hadamard product, and C is the joint vector representation of the i-th and j-th words. We obtain the labels for word pairs using the following formula:

$$\begin{aligned} & P\left(z_{i,j}\right) = \operatorname{softmax}\left(W_p w_{i,j}\right) \\ & \operatorname{label}\left(w_i, w_j\right) = \arg\max_t P\left(z_{i,j} = t\right) \end{aligned} \tag{10}$$

where $P(z_{i,j})$ is the probability of word pair (w_i, w_j) being recognized as label t.

4.5 Loss Function

We use the cross-entropy loss function to calculate the difference between the predicted results and the true labels:

$$L(t, \tilde{t}) = -\sum_{c=1}^{2R+2} t_c \log \tilde{t}_c \tag{11}$$

where t_c is the true label, $\tilde{t_c}$ is the predicted label.

5 Experiments

5.1 Datasets

We validate the performance of our model on two Chinese datasets CMedCausal (Li et al., 2022c) and CFinCausal [0]. CMedCausal is from the CHIP2022 academic evaluation task for medical causal entity relation extraction, The text is derived from online consultations and medical encyclopedia data from medical websites, with a length exceeding 200 Chinese characters. It includes causal relationships, conditional relationships, and hierarchical relationships. CFinCausal is a financial causal relation extraction dataset provided by the Tonghuashun platform in 2022, The dataset only contains causal relationships. The statistical results for the two datasets are shown in the Table 1. It is worth noting that in the CMedCausal dataset, the tail entity of the conditional relation tuple constitutes a complete causal relation tuple, We split the conditional relationships in Figure 1 into three relationship triplets. A complete conditional relation tuple is formed only when both the head entity and the tail entity of a causal relation are simultaneously related to another entity, forming a complete conditional relation tuple.

Datasets	Train	Vaild	Test
CMedCausal	899	100	1000
CFinCausal	1800	200	500

Table 1: Statistics of the data sets used in the experiment.

5.2 Baselines and Evaluation Metrics

To evaluate the model performance, we used the following powerful models as baselines: **Casrel** (Wei et al., 2019), **GPLinker** (Su, 2022), **TPLinker** (Wang et al., 2020), **GRTE** (Ren et al., 2021), **DEPT** (Liu et al., 2023b), **Onerel** (Shang et al., 2022), **UniRel** (Tang et al., 2022), **PRGC** (Zheng et al., 2021). We use $Macro\text{-}F1$ as the evaluation metric, and the specific calculation process is as follows:

$$Macro\text{-}F1 = \frac{1}{n} \sum_{i=1}^{n} \frac{2P_i R_i}{P_i + R_i} \tag{12}$$

where P_i is the ratio of true predicted samples for the i-th relation to the predicted samples for the i-th relation, and R_i is the ratio of true predicted samples for the i-th relation to the actual samples for the i-th relation.

5.3 Experimental Parameters

We set the hidden layer size of the pre-trained model to 1024. On the CMedCausal/CFinCausal datasets, we set the learning rate to 1e-4/5e-5, maximum character length of input sentences to 512/256, batch size for training to 8, and the dimension of the embedded double feature vectors to 1024. We employed dropout with a rate of 0.2 to prevent overfitting during training.

[0]The dataset can be obtained from http://contest.aicubes.cn/.

5.4 Experimental Results

Table 2 presents the results of our model and other baseline methods on the datasets. Our model outperforms other baseline methods on both datasets, with an improvement of 3.5% on the CMedCausal dataset and 1.4% on the CFinCausal dataset. The experimental results demonstrate the superiority of our approach. Casrel performs poorly on both datasets, likely due to its two-stage relation triplet extraction process, which is susceptible to error accumulation and puts it at a disadvantage compared to models that extract triplets simultaneously. Our approach and other baseline methods share the similarity of extracting triplets in one step, effectively avoiding the drawbacks of two-stage extraction models. The improvement in F1 score confirms the effectiveness of incorporating entity and relation information in our model, which enhances the efficiency of joint extraction.

Method	CMedCausal	CFinCausal		
	Macro-F1	Prec.	Rec.	Macro-F1
Casrel	21.6	61.9	51.0	55.9
PRGC	32.8	61.9	53.1	57.2
TPLinker	33.2	57.0	56.8	56.9
DEPT	34.5	50.8	51.8	51.3
Onerel	35.9	56.9	58.0	57.4
UniRel	37.9	62.3	54.3	58.0
GPLinker	38.3	**63.5**	54.3	58.5
GRTE	38.4	57.4	58.9	58.1
Ours	**41.9**	60.5	**59.3**	**59.9**

Table 2: Results of different methods on CMedCausal and CFinCausal datasets.

Method	Causal			Conditional			Hypernym		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Casrel	28.3	26.9	27.6	3.0	1.7	2.2	41.4	30.3	35.0
PRGC	47.7	46.7	47.2	7.7	2.8	4.1	49.8	44.7	47.1
TPLinker	46.4	47.2	46.8	7.8	3.9	5.2	44.7	50.9	47.6
DEPT	48.5	46.2	47.3	10.4	3.9	5.7	53.1	48.0	50.4
Onerel	49.4	50.0	49.7	8.8	5.3	6.6	48.6	54.3	51.3
UniRel	53.7	54.3	54.0	6.2	3.9	4.8	52.7	57.3	54.9
GPLinker	**58.3**	49.8	53.7	11.8	3.9	5.9	58.1	52.6	55.2
GRTE	50.3	**54.5**	52.3	11.5	**9.8**	10.6	45.3	**61.9**	52.3
Ours	58.2	51.4	**54.6**	**22.6**	8.5	**12.3**	**59.7**	58.3	**59.0**

Table 3: Detailed results of the three relationships for different methods on the CMedCausal dataset.

5.5 Detailed Results of Different Relationships

Table 3 shows the detailed performance metrics for three relations in the CMedCausal dataset, where our proposed method outperforms others across these relations. Although this dataset does not categorize complex triplets explicitly, our decomposition of conditional relations implies that we can treat them as complex relation triplets. The performance metrics for this relation also indicate that it poses the highest extraction difficulty. Among the compared methods, GRTE leads in extracting this relation triplet with an $F1$ score of 10.6, while ours achieves an $F1$ score of 12.3, representing a 1.7% improvement, demonstrating our method's advantage in extracting complex relation triplets. GRTE achieved the best recall rates on all three types of relationships because this method does not separately label entities using a table. When decoding the relationship table, all possible triplets are extracted, which effectively reduces the omission of correct relationship tuples. However, this approach lacks individual entity judgment, which increases the probability of incorrect relationship tuples. Furthermore, our method exhibits the most significant lead in extracting conditional relations, surpassing the best baseline method by 3.8%.

5.6 Ablation study

To verify the effectiveness of each component, we conducted ablation experiments on the CMedCausal dataset, and the results are shown in Table 4. Removing either of the two gating mechanisms had a negative impact on the model performance, indicating the positive role of the gating mechanisms in the model. Substituting the updated word vectors with raw word vectors directly, which results in the absence of feature-guided semantic updates, demonstrated that this component not only utilizes the original contextual information but also effectively updates semantic information. Removing the feature filtering module led to a significant decrease in model performance, highlighting the importance of the most relevant features for subsequent tasks. Ceasing the update of feature vectors affected the utilization of updates for entity and relation information from word vectors, while halting the update of word vectors resulted in the worst model performance, severely impacting the deep fusion of feature information and word information. Missing entity and relation information in word vectors directly affected the update of feature vectors, indirectly affecting the subsequent feature filtering module and the utilization of original information.

Method	Macro-F1
Complete model	41.9
-word vector update	38.9
-feature filtering module	39.4
-feature vector update	40.5
-raw word vectors update	40.8
-memory gate mechanism	41.4
-gate mechanism in the update network	41.6

Table 4: Abaltion experimental results on CMedCausal dataset.

5.7 The Number of Layers for Bidirectional Update Networks

The impact of different layers of bidirectional update networks on model performance is illustrated in Figure 5. The comparison between 0-layer and 1-layer networks shows a significant improvement in model performance, indicating the effectiveness of our proposed method. The 2-layer network shows a slight improvement over the 1-layer network, reaching optimal performance for the model, which suggests that as the number of update network layers increases, the model utilizes more entity and relation information. However, when the network reaches 3 layers, the model performance starts to decline. Therefore, we set the optimal number of layers for the model to be 2.

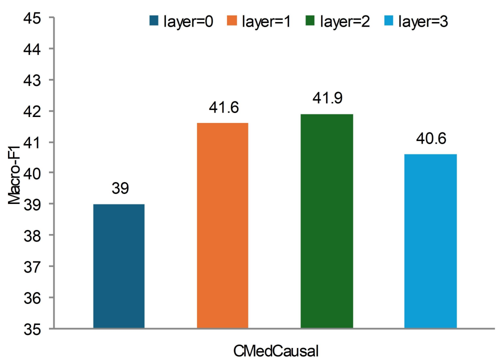

Figure 5: Results of different bidirectional update network layers on the CMedCausal dataset.

6 Conclusion

In this study, we propose a novel joint extraction model that treats joint extraction as a table filling problem, incorporating the joint representation of entity and relation information as prior knowledge and iteratively updating and enriching the information of word vectors. We achieve the retention and utilization of original information through a long-term memory gate mechanism. Experimental results on two datasets demonstrate that our approach effectively integrates entity and relation information, outperforming existing strong baseline models. It also shows superiority in extracting complex relationships.

Acknowledgements

This work was supported by the 1331 Engineering Project of Shanxi Province, China.

References

Rui Cai, Xiaodong Zhang, and Houfeng Wang. 2016. Bidirectional recurrent convolutional neural network for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 756–765.

Yee Seng Chan and Dan Roth. 2011. Exploiting syntactico-semantic structures for relation extraction. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 551–560.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3504–3514.

Chen Gao, Xuan Zhang, Hui Liu, Wei Yun, and Jiahao Jiang. 2022. A joint extraction model of entities and relations based on relation decomposition. *International Journal of Machine Learning and Cybernetics*, 13(7):1833–1845.

Shengbin Jia, E Shijia, Ling Ding, Xiaojun Chen, and Yang Xiang. 2022. Hybrid neural tagging model for open relation extraction. *Expert Systems with Applications*, 200:116951.

Wenjun Kong and Yamei Xia. 2023. Care: Co-attention network for joint entity and relation extraction. *arXiv preprint arXiv:2308.12531*.

Zhe Li and Luoyi Fu. 2022. A relation-aware span-level transformer network for joint entity and relation extraction. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Hao Li, Md Tauhidul Islam, Kehan Huangliang, Zugang Chen, Kaiyang Zhao, and Haixian Zhang. 2022a. Setff: A semantic enhanced table filling framework for joint entity and relation extraction. In *Pacific Rim International Conference on Artificial Intelligence*, pages 169–182. Springer.

Lishuang Li, Zehao Wang, Xueyang Qin, and Hongbin Lu. 2022b. Dual interactive attention network for joint entity and relation extraction. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 259–271. Springer.

Z Li, M Chen, Z Ma, et al. 2022c. Cmedcausal: Chinese medical causal relationship extraction dataset. *J. Med. Inform*, 43(12):23–27.

Yang Liu, Jinpeng Hu, Xiang Wan, and Tsung-Hui Chang. 2022a. A simple yet effective relation information guided approach for few-shot relation extraction. *arXiv preprint arXiv:2205.09536*.

Yibo Liu, Feng Wen, Teng Zong, and Taowei Li. 2022b. Research on joint extraction method of entity and relation triples based on hierarchical cascade labeling. *IEEE Access*, 11:9789–9798.

PeiYu Liu, JunPing Du, YingXia Shao, and Zeli Guan. 2023a. Relation extraction model based on semantic enhancement mechanism. In *2023 5th International Conference on Data-driven Optimization of Complex Systems (DOCS)*, pages 1–7. IEEE.

Xu Liu, Ruiqi Cai, and Yonggang Zhang. 2023b. A 2d entity pair tagging scheme for relation triplet extraction. In *International Conference on Knowledge Science, Engineering and Management*, pages 16–29. Springer.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. 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*, pages 1003–1011.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. *arXiv preprint arXiv:1601.00770*.

Makoto Miwa and Yutaka Sasaki. 2014. Modeling joint entity and relation extraction with table representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1858–1869.

Tapas Nayak and Hwee Tou Ng. 2020. Effective modeling of encoder-decoder architecture for joint entity and relation extraction. In *Proceedings of the AAAI conference on artificial intelligence*, number 05, pages 8528–8535.

Jinzhong Ning, Zhihao Yang, Yuanyuan Sun, Zhizheng Wang, and Hongfei Lin. 2023. Od-rte: A one-stage object detection framework for relational triple extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11120–11135.

Feiliang Ren, Longhui Zhang, Shujuan Yin, Xiaofeng Zhao, Shilei Liu, Bochao Li, and Yaduo Liu. 2021. A novel global feature-oriented relational triple extraction model based on table filling. *arXiv preprint arXiv:2109.06705*.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part III 21*, pages 148–163. Springer.

Yu-Ming Shang, Heyan Huang, and Xianling Mao. 2022. Onerel: Joint entity and relation extraction with one module in one step. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 11285–11293.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 885–895.

J Su. 2022. Gplinker: entity-relation joint extraction based on globalpointer.

Ryuichi Takanobu, Tianyang Zhang, Jiexi Liu, and Minlie Huang. 2019. A hierarchical framework for relation extraction with reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7072–7079.

Wei Tang, Benfeng Xu, Yuyue Zhao, Zhendong Mao, Yifeng Liu, Yong Liao, and Haiyong Xie. 2022. Unirel: Unified representation and interaction for joint relational triple extraction. *arXiv preprint arXiv:2211.09039*.

Jue Wang and Wei Lu. 2020. Two are better than one: Joint entity and relation extraction with table-sequence encoders. *arXiv preprint arXiv:2010.03851*.

Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020. Tplinker: Single-stage joint extraction of entities and relations through token pair linking. *arXiv preprint arXiv:2010.13415*.

Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2019. A novel cascade binary tagging framework for relational triple extraction. *arXiv preprint arXiv:1909.03227*.

Wenqiang Xu, Shiqun Yin, Junfeng Zhao, and Ting Pu. 2021. Deep semantic fusion representation based on special mechanism of information transmission for joint entity-relation extraction. In *PRICAI 2021: Trends in Artificial Intelligence: 18th Pacific Rim International Conference on Artificial Intelligence, PRICAI 2021, Hanoi, Vietnam, November 8–12, 2021, Proceedings, Part II 18*, pages 73–85. Springer.

Yue Yuan, Xiaofei Zhou, Shirui Pan, Qiannan Zhu, Zeliang Song, and Li Guo. 2021. A relation-specific attention network for joint entity and relation extraction. In *International joint conference on artificial intelligence*. International Joint Conference on Artificial Intelligence.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th international conference on computational linguistics: technical papers*, pages 2335–2344.

Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 506–514.

Xiangrong Zeng, Shizhu He, Daojian Zeng, Kang Liu, Shengping Liu, and Jun Zhao. 2019. Learning the extraction order of multiple relational facts in a sentence with reinforcement learning. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 367–377.

Daojian Zeng, Haoran Zhang, and Qianying Liu. 2020. Copymtl: Copy mechanism for joint extraction of entities and relations with multi-task learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9507–9514.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2017. End-to-end neural relation extraction with global optimization. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 1730–1740.

Ranran Haoran Zhang, Qianying Liu, Aysa Xuemo Fan, Heng Ji, Daojian Zeng, Fei Cheng, Daisuke Kawahara, and Sadao Kurohashi. 2020. Minimize exposure bias of seq2seq models in joint entity and relation extraction. *arXiv preprint arXiv:2009.07503*.

Kang Zhao, Hua Xu, Yue Cheng, Xiaoteng Li, and Kai Gao. 2021. Representation iterative fusion based on heterogeneous graph neural network for joint entity and relation extraction. *Knowledge-Based Systems*, 219:106888.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. *arXiv preprint arXiv:1706.05075*.

Hengyi Zheng, Rui Wen, Xi Chen, Yifan Yang, Yunyan Zhang, Ziheng Zhang, Ningyu Zhang, Bin Qin, Ming Xu, and Yefeng Zheng. 2021. Prgc: Potential relation and global correspondence based joint relational triple extraction. *arXiv preprint arXiv:2106.09895*.

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