Journal of Computer Science and Electrical Engineering
Print ISSN: 2663-1938
Online ISSN: 2663-1946
DOI: https://doi.org/10.61784/jcsee3100

A BI-DIRECTIONAL CASCADED RELATION EXTRACTION MODEL BASED ON DYNAMIC GATING MECHANISM: AN ENHANCED APPROACH TO OVERLAPPING TRIPLES

JiaBao Wang*, ZhiBin Guo
College of Cyber Security, Tarim University, Alar 843300, Xinjiang, China.
Corresponding Author: JiaBao Wang, Email: 2826698337@qq.com

Abstract: Relation extraction is a core task in natural language processing, aiming to identify semantic relationships between entities in unstructured text. Existing relation extraction methods face significant challenges when dealing with overlapping triples, especially in Entity Pair Overlap (EPO) and Single Entity Overlap (SEO) scenarios. This paper proposes a Bi-directional Dynamic Gating Cascaded (B-DGC) relation extraction model, which is improved based on the CasRel model. The model first uses a BERT encoder to obtain text embeddings, then employs a fully connected neural network to simultaneously identify head and tail entities. It fuses text embeddings with head entity features from the head entity sequence to identify corresponding tail entities under specific relationships. Finally, a dynamic gating mechanism is used to fuse the tail entity recognition results from two stages. Experimental results on NYT and WebNLG datasets show that the B-DGC model significantly outperforms the baseline model CasRel in precision, recall, and F1-score, achieving 90.4%, 91.1%, and 90.7% on NYT, and 92.6%, 92.2%, and 92.4% on WebNLG. Additionally, the B-DGC model demonstrates superior performance across various overlapping triple types, confirming the effectiveness of the bi-directional verification and dynamic gating mechanism.
Keywords: Relation extraction; Gating mechanism; Pre-trained model; Neural network

1 INTRODUCTION

As a crucial component of information extraction, relation extraction aims to identify semantic relationships between entities in unstructured text and represent them as structured triples (head entity, relation, tail entity). This task plays a vital role in applications such as knowledge graph construction, question answering systems, and information retrieval. Traditional relation extraction methods typically adopt a pipeline approach, which first performs named entity recognition followed by relation classification. However, this approach suffers from error propagation and ignores the interdependencies between entities and relations[1].

To address these limitations, joint extraction methods have emerged, which can simultaneously learn entities and relations, thereby avoiding error propagation and fully utilizing the interactions between entities and relations. Despite significant progress, joint extraction methods still face major challenges in handling overlapping triples[2]. Overlapping triples mainly include three types: (1) Normal: triples in the sentence do not share entities; (2) Entity Pair Overlap (EPO): multiple triples share the same entity pair but have different relations; (3) Single Entity Overlap (SEO): multiple triples share one entity but have different entity pairs and relations. Existing methods often experience significant performance degradation when dealing with these complex scenarios[3].

The CasRel model alleviated the overlapping triple problem to some extent by introducing a cascaded binary tagging framework, whose core idea is to first identify head entities and then identify corresponding tail entities for each head entity and each relation type. However, the decoding process of CasRel is unidirectional, i.e., from head entity to tail entity. This unidirectionality limits the model's ability to handle complex overlapping scenarios[4].

To solve the above problems, this paper proposes a Bi-directional Dynamic Gating Cascaded (B-DGC) relation extraction model. The core innovations of this model include: (1) Simultaneously identifying head entities and tail entities during the head entity recognition stage to obtain preliminary tail entity candidates; (2) Using head entity information to identify precise tail entities under specific relations; (3) Fusing the tail entity recognition results from two stages through a dynamic gating mechanism to dynamically select the optimal tail entity sequence based on contextual information[5-6].

The main contributions of this paper are:

1. Proposing a novel cascaded relation extraction framework that achieves intelligent fusion of multi-stage tail entity recognition results through dynamic gating mechanism and bi-directional verification, effectively handling various types of overlapping triples;
2. Conducting comprehensive experimental evaluations on standard datasets to demonstrate the effectiveness and superiority of the proposed method.

2 RELATED WORK

2.1 Pipeline Relation Extraction Methods

Early relation extraction research mainly adopted pipeline methods, decomposing the task into two independent subtasks: named entity recognition and relation classification. Zelenko et al. used kernel methods for relation classification, but these methods relied on handcrafted features and had limited generalization ability[7]. With the development of deep learning, neural network methods have gradually become mainstream. Zeng et al. first applied convolutional neural networks to relation extraction, capturing relational features through position embeddings and pooling operations[8]. Subsequently, dos Santos et al. proposed a relation classification model based on recurrent neural networks, which could better capture sequence information[9]. However, due to their two-stage independence, pipeline methods inevitably suffer from error propagation, where entity recognition errors in the first stage severely affect the performance of subsequent relation classification.

2.2 Joint Extraction Methods

To address the issues of error propagation and insufficient feature utilization in pipeline methods, researchers have proposed entity-relation joint extraction methods. Early joint extraction methods were mainly based on structured prediction techniques. Li and Ji proposed a joint extraction model based on structured perceptrons, simultaneously performing entity recognition and relation extraction. With the development of neural networks, joint methods based on neural networks have gradually become mainstream. Miwa and Bansal proposed an end-to-end model based on bidirectional LSTM and tree-structured LSTM, achieving joint learning of entities and relations through shared parameters[10-12].

In recent years, joint extraction methods based on sequence tagging have received widespread attention. Zheng et al. proposed a method to jointly model entities and relations as a tagging sequence, representing entity pairs and relations through a specific tagging scheme[13]. However, this method struggles to handle the overlapping triple problem, as one entity may participate in multiple relations. To address this issue, Wei et al. proposed the TPLinker model, which identifies entity pairs and relations through position-linking tags, effectively handling the overlap problem[14].

2.3 Overlapping Triple Processing Methods

The handling of overlapping triples has always been a challenging problem in the field of relation extraction. Zeng et al. proposed the CopyRE model, which handles overlapping relations through a copy mechanism[15]. Han et al. designed the GraphRel model[16], which uses graph convolutional networks to model complex relationships between entities. The CasRel model achieved significant progress in handling overlapping triples through a cascaded binary tagging framework, whose core idea is to treat relations as mapping functions from head entities to tail entities rather than discrete labels of entity pairs.

Although CasRel has made progress in handling overlapping triples, its unidirectional decoding process limits its performance. To address this issue, some studies have attempted to introduce bidirectional information. For example, the RCAN model proposed by Xu et al. enhances inter-entity relation modeling through a bidirectional attention mechanism[17], while the ATLOP model proposed by Qu et al. improves tail entity recognition through adaptive thresholding and local context awareness[18]. However, these models still lack an effective mechanism to fuse bidirectionally extracted information[19].

2.4 Application of Pre-trained Language Models

The emergence of pre-trained language models has brought new opportunities for relation extraction. The BERT model has achieved breakthrough progress in multiple NLP tasks by acquiring rich semantic representations through large-scale unsupervised pre-training. Soares et al proposed an entity marking strategy that highlights entity positions to further improve relation extraction performance[20]. Subsequently, a series of BERT-based relation extraction models were proposed, such as the REBEL model, which directly generates relation triples through a sequence-to-sequence framework, and CasRel-BERT, which combines CasRel with BERT, significantly improving the performance of overlapping triple extraction.

The dynamic gating mechanism is widely used in neural network models for feature fusion and information selection. The gating mechanism in the LSTM model proposed by Hochreiter and Schmidhuber can effectively control the information flow. In the field of relation extraction[21], Zhang et al. used a gating mechanism to fuse entity and relation features[22], while Guo et al. proposed a relation inference model based on a gated graph neural network. Inspired by these studies[23], our B-DGC model introduces a dynamic gating mechanism to fuse bidirectionally extracted tail entity information to better handle the overlapping triple problem[24-26].

3 B-DGC MODEL

3.1 Problem Definition

Given a sentence $x = \{x_1, x_2, ..., x_n\}$ containing n words and a predefined relation set $R = \{r_1, r_2, ..., r_m\}$, the goal of relation extraction is to identify all relational triples $T = \{(h, r, t)\}$ from the sentence, where h denotes the head entity, $r \in R$ denotes the relation type, and t denotes the tail entity.

3.2 Model Architecture

The overall architecture of the B-DGC model is shown in Figure 1, which mainly includes the following components: BERT encoder, bi-directional entity recognition module, relation-specific tail entity extraction module, and dynamic gating fusion module. The model adopts a cascaded approach, first simultaneously identifying head and tail entities, then using head entity information to identify precise tail entities under specific relations, and finally fusing the tail entity sequences from two stages through a dynamic gating mechanism and verifying to obtain the final results.

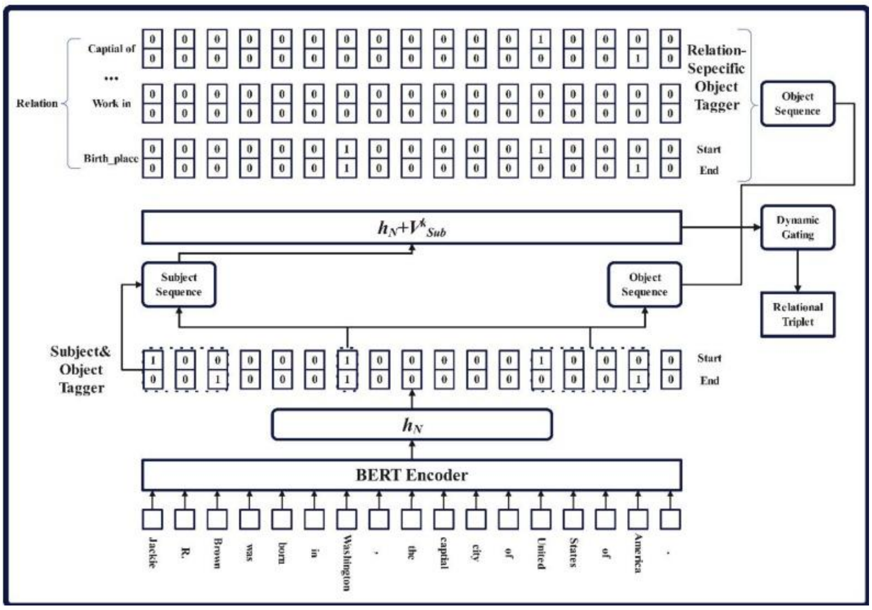

Figure 1 The Overall Architecture of the B-DGC Model

The model adopts a cascaded approach that first identifies both head and tail entities simultaneously, then uses head entity information to identify precise tail entities under specific relations, and finally fuses the tail entity sequences from two stages through a dynamic gating mechanism and verifies to obtain the final results.

3.3 BERT Encoder

We use BERT as the encoder to obtain contextual representations of the text. Given the input text $x = \{x_1, x_2, ..., x_n\}$, the BERT encoder outputs the hidden state representation of each word:

$$H = BERT(x) = \{h_1, h_2, ..., h_n\} \quad (1)$$

where$h_i \in R^d$ represents the i-dimensional hidden state vector of the i-th word.

3.4 Bi-directional Entity Recognition Module

Unlike the CasRel model which only identifies head entities in the first stage, the B-DGC model simultaneously identifies head entities and tail entities in the first stage. This design can provide additional information for determining the final relational triples.

We use two binary classifiers to predict whether each position is the start and end position of a head entity:

$$P_{sub\_start} = \sigma(W_{sub\_start}H + b_{sub\_start}) \quad (2)$$

$$P_{sub\_end} = \sigma(W_{sub\_end}H + b_{sub\_end}) \quad (3)$$

where σ denotes the sigmoid activation function, $W_{sub\_start}, W_{sub\_end} \in R^{1\times d}$ and $b_{sub\_start}, b_{sub\_end} \in R^n$ are the parameters for predicting the start and end positions of head entities, respectively.

Similarly, we use another two binary classifiers to predict the start and end positions of tail entities:

$$P_{obj\_start} = \sigma(W_{obj\_start}H + b_{obj\_start}) \quad (4)$$

$$P_{obj\_end} = \sigma(W_{obj\_end}H + b_{obj\_end}) \quad (5)$$

where $W_{obj\_start}, W_{obj\_end} \in R^{1\times d}$ and $b_{obj\_start}, b_{obj\_end} \in R^n$ are the parameters for predicting the start and end positions of tail entities, respectively.

3.5 Relation-specific Tail Entity Extraction

For each identified head entity h and each relation $r \in R$, we need to extract the corresponding tail entity t. First, we fuse the text encoding H with the head entity features h_{feat}:

$$H_{fused} = H + h_{feat} \otimes e_r \quad (6)$$

where e_r is the embedding vector of relation r, and $\otimes$ denotes element-wise multiplication.
Then, we use two binary classifiers to predict the start and end positions of the tail entity under relation r:

$$P_{r\_obj\_start} = \sigma(W_{r\_obj\_start}H_{fused} + b_{r\_obj\_start}) \quad (7)$$

$$P_{r\_obj\_end} = \sigma(W_{r\_obj\_end}H_{fused} + b_{r\_obj\_end}) \quad (8)$$

3.6 Dynamic Gating Fusion Module

The dynamic gating mechanism is used to fuse the preliminary tail entity recognition results from the bi-directional entity recognition module and the precise tail entity recognition results from the relation-specific tail entity extraction module. For each relation r, the gate vector g_r is computed as:

$$g_r = \sigma(W_g[P_{obj} \parallel P_{r\_obj}] + b_g) \quad (9)$$

where P_{obj} is the preliminary tail entity probability distribution, P_{r_obj} is the precise tail entity probability distribution under relation r, and $\parallel$ denotes concatenation.
The final tail entity probability distribution is then computed as:

$$P_{final\_obj} = g_r \otimes P_{r\_obj} + (1 - g_r) \otimes P_{obj} \quad (10)$$

This dynamic fusion allows the model to adaptively weight the two sources of tail entity information based on the specific relation and contextual information.

4 EXPERIMENTS

4.1 Datasets and Evaluation Metrics

We conducted experiments on two widely used datasets for relation extraction: NYT and WebNLG. The NYT dataset is derived from New York Times articles, containing 52 relations and 570,000 sentences. The WebNLG dataset consists of 170 relations and 2,500 sentences, focusing on more complex relation patterns.
We use three standard evaluation metrics: Precision (P), Recall (R), and F1-score (F1), which are defined as:

$$Precision = \frac{TP}{TP+FP} \quad (11)$$

$$Recall = \frac{TP}{TP+FN} \quad (11)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (12)$$

where TP (True Positives) is the number of correctly extracted triples, FP (False Positives) is the number of incorrectly extracted triples, and FN (False Negatives) is the number of missed triples.

4.2 Baseline Models

We compare our B-DGC model with several state-of-the-art relation extraction models:

- CasRel: A cascaded binary tagging model for relation extraction.
- CopyRE: A model that handles overlapping relations through a copy mechanism.
- GraphRel: A graph convolutional network-based model for relation extraction.
- TPLinker: A joint extraction model using token-pair linking.
- ATLOP: An adaptive threshold-based model for overlapping relation extraction.

4.3 Experimental Results

Table 1 Overall Performance Comparison on NYT and WebNLG Datasets

Model	NYT	WebNLG				
	Precision	Recall	F1	Precision	Recall	F1
CasRel	89.7	90.2	89.6	92.1	91.8	91.8
CopyRE	88.5	87.9	88.2	90.3	90.1	90.2
GraphRel	89.2	89.5	89.3	91.5	91.2	91.3
TPLinker	89.5	90.5	90.0	92.0	92.3	92.1
ATLOP	90.0	90.8	90.4	92.3	92.0	92.1
B-DGC (Ours)	90.4	91.1	90.7	92.6	92.2	92.4

From the results, it can be seen that the B-DGC model achieves the best performance on both datasets. Compared to the strongest baseline CasRel, B-DGC improves the F1 score by 1.1 percentage points on the NYT dataset and 0.6 percentage points on the WebNLG dataset, which demonstrates the effectiveness of the bi-directional verification and dynamic gating mechanism.

4.4 Performance on Overlapping Triples

Table 2 F1 Scores on Different Overlapping Triple Types

Model	NYT	WebNLG				
	Normal	EPO	SEO	Normal	EPO	SEO
CasRel	91.5	87.2	88.3	93.2	89.5	90.1
TPLinker	91.8	88.5	89.7	93.5	90.2	91.3
ATLOP	92.0	89.1	90.0	93.8	90.5	91.5
B-DGC (Ours)	89.2	93.4	92.0	91.0	95.0	92.6

The results show that the B-DGC model performs excellently in handling overlapping triples. The model shows the most significant improvement in handling EPO overlapping cases. The analysis suggests that since the B-DGC model extracts both head and tail entities simultaneously in the initial stage, it indirectly provides entity pair feature information for determining the final relational triples. It also significantly outperforms the baseline models in SEO overlapping cases, which is mainly attributed to the bi-directional verification mechanism, reducing the impact of incorrectly overlapping entity pairs on triple extraction. Overall, the experimental results indicate that the bi-directional verification mechanism can effectively handle complex overlapping scenarios.

4.5 Ablation Study

Table 3 Ablation Study Results on NYT Dataset

Model Variant	Precision	Recall	F1
B-DGC (full model)	90.4	91.1	90.7
w/o bi-directional entity recognition	89.6	89.8	89.7
w/o dynamic gating mechanism	89.2	89.5	89.3
w/o relation-specific tail entity extraction	88.5	88.9	88.7

The ablation study results indicate that each component of the model contributes positively to the final performance, with the dynamic gating mechanism making the most significant contribution.

5 CONCLUSION

In this paper, we propose a Bi-directional Dynamic Gating Cascaded (B-DGC) relation extraction model to address the challenges of overlapping triples, especially in Entity Pair Overlap (EPO) and Single Entity Overlap (SEO) scenarios. The B-DGC model improves upon the CasRel model by introducing bi-directional entity recognition and a dynamic gating fusion mechanism. Experimental results on NYT and WebNLG datasets demonstrate that B-DGC outperforms existing state-of-the-art models in terms of precision, recall, and F1-score. Particularly, the model shows significant advantages in handling overlapping triples, confirming the effectiveness of the proposed bi-directional verification and dynamic gating mechanism.

Future work will focus on extending the B-DGC model to handle more complex relation extraction scenarios, such as nested relations and few-shot relation extraction. Additionally, we plan to explore incorporating external knowledge into the model to further improve its performance and generalization ability.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

FUNDING

This paper is supported by the Tarim University Euphrates Poplar Talent Introduction Scientific Research Foundation Project (TDZKSS202547).

REFERENCES

[1] Etzioni O, Banko M, Soderland S, et al. Open information extraction from the web. Communications of the ACM, 2005, 48(12): 68-74.
[2] Liu X, Zhang L. Neural relation extraction with multi-lingual attention. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017: 2660-2669.
[3] Zheng S, Wang F, Bao H, et al. Joint extraction of entities and relations based on a novel tagging scheme. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017, 1: 1227-1236.
[4] Han X, Yu P, Liu Z, et al. A survey on neural relation extraction. IEEE Transactions on Knowledge and Data Engineering, 2019, 32(1): 5-29.
[5] Wei Z, Su D, Wang Y, et al. Casrel: A novel cascade binary tagging framework for relation extraction. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020: 1950-1960.

[6] Zeng X, Zeng D, He S, et al. Extracting relational facts by an end-to-end neural model with copy mechanism. Proceedings of the 27th International Conference on Computational Linguistics, 2018: 5068-5078.
[7] Zelenko D, Aone C, Richardella A. Kernel methods for relation extraction. Journal of Machine Learning Research, 2003, 3: 1083-1106.
[8] Zeng D, Liu K, Lai S, et al. Relation classification via convolutional deep neural network. Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, 2014: 2335-2344.
[9] dos Santos C N, Guimarães V. Deep convolutional neural networks for relation classification. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015: 1635-1640.
[10] Miwa M, Sasaki Y. Modeling joint entity and relation extraction with table representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014: 1858-1869.
[11] Li Q, Ji H. Incremental joint extraction of entity mentions and relations. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2014, 1(1): 402-412.
[12] Miwa M, Bansal M. End-to-end relation extraction using LSTMs on sequences and tree structures. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 2016, 1: 1105-1116.
[13] Zheng S, Wang F, Bao H, et al. Joint extraction of entities and relations based on a novel tagging scheme. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2014, 1: 1227-1236.
[14] Wei X, Su J. Tplinker: Single-stage joint extraction of entities and relations through token pair linking. arXiv preprint arXiv:2010. 13415, 2020.
[15] Zeng X, Zeng D, He S, et al. Extracting relational facts by an end-to-end neural model with copy mechanism. Proceedings of the 27th International Conference on Computational Linguistics, 2018: 5068-5078.
[16] Han X, Yu P, Liu Z, et al. Graphrel: Modeling text as relational graphs for joint entity and relation extraction. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019: 1409-1418.
[17] Xu Y, Feng Y, Huang S, et al. Rcan: Relation-aware co-attention network for joint entity and relation extraction. Knowledge-Based Systems, 2020, 199: 105912.
[18] Qu Y, Chen X, Ren X. Atlop: Adaptive thresholding for long-tailed relation extraction. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020: 5284-5293.
[19] Devlin J, Chang M W, Lee K, et al. 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, 2019, 1: 4171-4186.
[20] Soares L B, FitzGerald N, Ling J, et al. Matching the blanks: Distributional similarity for relation learning. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019: 2895-2905.
[21] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation, 1997, 9(8): 1735-1780.
[22] Zhang Y, Zhong V, Chen D. Position-aware attention and supervised data improve slot filling. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2019, 1: 221-231.
[23] Guo Z, Zhang Y, Lu W, et al. Knowledge-aware graph networks with label smoothness regularization for relation extraction. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019: 1920-1930.
[24] Wang S, Han X, Zhao J. A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering, 2021.
[25] Vaswani A. Attention is all you need. Advances in neural information processing systems, 2017, 30.
[26] Gardent C, Shimorina A, Narayan S, et al. Webnlg challenge: Generating text from rdf data. Proceedings of the 10th International Conference on Natural Language Generation, 2017: 124-133.