

Article

Not peer-reviewed version

Research on Entity Pair Relation Classification Based on Contrastive Learning and Biaffine Model

[Songhua Hu](#)*, [ZiMing Zhang](#), Hengxin Wang, Lihui Jiang

Posted Date: 20 March 2025

doi: 10.20944/preprints202503.1462.v1

Keywords: Entity Pair Relation Classification; Contrastive Learning; natural language processing; text classification



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Research on Entity Pair Relation Classification Based on Contrastive Learning and Biaffine Model

Songhua Hu *, Ziming Zhang, Hengxin Wang and Lihui Jiang

School of Artificial Intelligence and Big Data, Hefei University, Hefei, 230601, China

* Correspondence: hush@hfu.edu.cn

Abstract: In the field of natural language processing, Biaffine model is a typical neural network structure based on double affine transformation, which helps to understand sentence structure and the relationships between words, which can be used for tasks such as text classification and relation extraction. However, this model faces challenges in entity pair relation classification tasks, such as imbalanced relation types and unclear entity pair feature information. Therefore, this paper proposes an improved relation classification model named Bert-CL-Biaffine, which is based on bidirectional entity and contrastive learning, combining a global pointer network with contrastive learning to enhance the Biaffine model for relation classification tasks. By training the model to identify the start and end positions of entities in sentences, it performs better in classifying overlapping entity pairs in complex scenarios. Experimental results show that on the NYT and WebNLG datasets, the F1 score of Bert-CL-Biaffine model improves by 1% and 1.2%, respectively, compared to baseline models, indicating that the improved relation classification model effectively enhances performance in complex scenarios.

Keywords: entity pair relation classification; contrastive learning; natural language processing; text classification

1. Introduction

Relation classification tasks are widely used in natural language processing to predict potential relationships between two entities in unstructured text. As shown in Figure 1, the entity pairs in relation classification tasks can be categorized into three types: 1) Normal class, where the sentence contains only ordinary triplets with no association between entity pairs; 2) EPO (entity pair overlap) class, where entity pairs in the sentence have multiple relationships; and 3) SPO (single entity overlap) class, where entity pairs in the sentence have overlapping associations. The core issue in relation classification tasks is how to handle overlapping entity pairs.

Normal	<div>The [United States] President [Trump] has a meet with [Tim Cook], the CEO of [Apple Inc].</div> <div>country_president</div> <div>company_ceo</div>
EPO	<div>Demolition has begun on another purchase in [Long Island City],[Queens].</div> <div>neighborhood_of</div> <div>contains</div>
SEO	<div>The New [York] officers drove to [Poughkeepsie], N.Y. , from [Albany].</div> <div>contains</div> <div>contains</div>

Figure 1. Entity Pair Category Classification.

The Biaffine model is a dependency parsing model based on bidirectional LSTM (BiLSTM), which uses a biaffine attention mechanism to predict the head node and dependency label in

dependency relations. It employs two independent multi-layer perceptrons (MLPs) to generate head representations and dependent representations, respectively. The model calculates the score of whether a dependency relationship exists between two words through biaffine transformation, thereby modeling complex interactive relationships. The Biaffine model achieved state-of-the-art performance in dependency parsing tasks and has become the foundation for many subsequent works, which widely applied in other structured prediction tasks. After the Biaffine model was proposed, many scholars improved and optimized it. For example, Ke et al. [1] proposed a Chinese named entity recognition (CNER) method based on multi-feature fusion and biaffine mechanisms, improving recognition performance by integrating Chinese character glyph and pinyin features and introducing biaffine mechanisms. Xiao et al. [2] proposed a method based on span annotation and biaffine attention mechanisms for aspect sentiment triplet extraction in fine-grained sentiment analysis. Aziz et al. [3] improved UrduAspectNet by incorporating biaffine attention mechanisms into the model architecture, combining XLM-R embeddings, bidirectional LSTM, and dual graph convolutional networks, and using dependency parsing to create adjacency matrices for graph convolutional networks. Piao et al. [4] proposed a novel extraction model, BiLSTM-BGAT-GCN, which cleverly integrates graph neural networks with improved biaffine attention mechanisms, significantly enhancing the interpretation and extraction of text features by leveraging the powerful capabilities of graph attention networks and convolutional networks in processing graph-structured data.

This paper addresses the shortcomings of the BERT-Biaffine model by using a global pointer network to construct the head and tail positions of entity pairs in sentences to predict the relationship types between entity pairs, thereby handling some overlapping entity pair issues. Due to the imbalanced relation labels in the training set and some entity pairs with complex feature information, the model's performance is significantly affected, leading to errors in classification tasks. This paper improves the Biaffine model's performance in complex scenarios by using contrastive learning methods, better understanding the semantic associations between overlapping entity pairs in sentences, and further enhancing the model's understanding of relation labels through adversarial training. Experimental results on the NYT and WebNLG datasets show that the proposed model effectively improves relation classification performance in complex scenarios.

The remainder of this paper is organized as follows. Section 2 discusses related works, Section 3 designs the model, Section 4 presents experimental simulations to verify the proposed method, and Section 5 summarizes the paper and provides possible future work.

2. Related Work

2.1. Entity Relation Extraction

In early research, entity relation triplet extraction was typically divided into two independent tasks: entity extraction and relation prediction. Traditional pipeline methods first extract all entities in a sentence and then classify the relations for each entity pair. However, this approach is prone to error propagation [5,6].

To overcome this issue, joint extraction methods emerged, with mainstream sequence labeling methods transforming the triplet extraction task into multiple interrelated sequence labeling problems. For example, Zheng et al. [7] proposed a Novel Tagging framework that directly transforms the joint extraction task into a labeling problem, simultaneously extracting entities and their relations. Wei et al. [8] first identify all possible head entities and then use a relation-specific sequence tagger to identify the corresponding tail entities for each head entity. Zheng et al. [9] proposed the PGRC method, further introducing a component to predict potential relations, limiting entity recognition to the predicted subset of relations rather than all relations. Additionally, Ren et al. [10] proposed a bidirectional entity extraction framework, BIRTE, which simultaneously extracts entity pairs from both directions, improving extraction performance. These methods effectively

address the limitations of pipeline methods and significantly enhance the performance of entity relation triplet extraction.

2.2. Biaffine Model and BERT-Biaffine Model

2.2.1. Biaffine Model

The Biaffine model (double affine model) is a neural network model for structured prediction tasks, initially proposed for syntactic parsing tasks [11]. Its core idea is to model the potential relationship between two elements, such as two words in a sentence, through biaffine transformation. Biaffine transformation is an affine transformation that considers both inputs, i.e., a linear transformation plus a bias, and can model the relationship between two vectors. The mathematical form is:

$$\text{Score}(\mathbf{h}, \mathbf{d}) = \mathbf{h}^T \mathbf{W} \mathbf{d} + \mathbf{U} \mathbf{h} + \mathbf{V} \mathbf{d} + \mathbf{b} \quad (1)$$

where \mathbf{h} and \mathbf{d} are the input vectors, such as two words in a sentence, \mathbf{W} is the weight matrix, \mathbf{U} and \mathbf{V} are projection matrices, and \mathbf{b} is the bias term.

The Biaffine model is applied to structured prediction tasks. In dependency parsing, the model predicts the dependency relationship for each word in a sentence, such as subject-verb or verb-object relationships. The Biaffine model generates a score for each word pair (\mathbf{h}, \mathbf{d}) , indicating the likelihood of \mathbf{d} being a dependent \mathbf{h} , and selects the optimal relationship through Softmax. The model structure mainly consists of three parts: (1) Encoder: Typically, a bidirectional LSTM (BiLSTM) or Transformer is used to encode the input sentence, generating contextual representations for each word. (2) Head and dependent representations: Two independent representations are generated for each word: head representation and dependent representation. Head representation represents the feature of the word as a parent node (head), while dependent representation represents the features of the word as a child node (dependent). These two representations are generated through different fully connected layers (MLPs). (3) Biaffine classifier: The biaffine transformation is used to calculate the relationship score for each word pair (\mathbf{h}, \mathbf{d}) , and the relationship type is predicted through Softmax. The advantage of the Biaffine model is its explicit modeling of word pair relationships, making it suitable for structured tasks without the need for manually designed features.

2.2.2. BERT-Biaffine Model

The BERT-Biaffine model is an improved version of the Biaffine model, combining the contextual representation capabilities of pre-trained language models, such as BERT, with the relationship modeling capabilities of biaffine transformation, significantly improving performance [12]. The core improvements also summarized as follows: (1) BERT as the encoder: BERT replaces the traditional BiLSTM encoder, directly generating richer context-dependent word vectors. BERT's word vectors capture global semantic and syntactic information, making it particularly suitable for tasks requiring deep semantic understanding. (2) End-to-end training: BERT and the Biaffine classifier are jointly trained, with BERT parameters fine-tuned to adapt to downstream tasks. The structure of the BERT-Biaffine model remains consistent with the original Biaffine model. It mainly includes input representation: the input sentence is encoded by BERT to generate context-dependent word vectors. Head and dependent representations: similar to the original Biaffine model, two independent MLPs are used to generate head and dependent representations. Biaffine classifier: same as the original Biaffine model, calculating word pair relationship scores and predicting dependency relationships. (3) The advantages of the BERT-Biaffine model include stronger contextual representation. BERT's pre-trained weights provide high-quality semantic and syntactic features. Higher accuracy: achieves state-of-the-art performance in tasks such as dependency parsing and named entity recognition. Transfer learning capability: by fine-tuning BERT, the model can adapt to multiple languages and domains. The BERT-Biaffine model is applied in dependency parsing: predicting dependency trees between words in sentences. Named entity recognition (NER):

identifying entities and their types in text e.g., person names, location names. Relation extraction: extracting relationships between entities in sentences e.g., "founder-company" relationships. In summary, the Biaffine model models structured relationships through biaffine transformation, suitable for tasks such as syntactic parsing. The BERT-Biaffine model combines BERT's strong representation capabilities with the Biaffine classifier, significantly improving task performance, and is one of the mainstream methods in NLP.

2.3. Contrastive Learning

Entity relation extraction tasks need to address both the modeling of semantic similarity between entity pairs and the challenge of imbalanced sample distribution, where the number of negative samples significantly exceeds that of positive samples. To address this issue, contrastive learning constructs positive and negative sample pairs through data augmentation, enhancing semantic representation learning based on similarity comparison mechanisms. This approach balances sample distribution while optimizing the discriminative power of the feature space, thereby improving model generalization performance. The theoretical evolution of this method can be traced back to the field of natural language processing: Mikolov et al. [13] pioneered the Word2Vec model, which explored the semantic compositionality of words through distributed representations, laying the foundation for contrastive learning. Yang et al. [14] further addressed the issue of semantic loss in machine translation by comparing the long-range dependency differences between original and generated samples. These technological breakthroughs provide key methodological support for entity relation modeling in complex scenarios.

2.4. R-Drop Contrastive Learning Method

R-Drop (Regularized Dropout) is a regularization method that combines the Dropout mechanism with contrastive learning, aiming to enhance model generalization and robustness by enforcing consistency in model outputs under Dropout perturbations [15]. The following summarizes its core points and working principles. R-Drop leverages the randomness of Dropout to generate different subnetwork outputs for the same input and constrains the consistency of these outputs as a regularization term, such as KL divergence, making the model more robust to Dropout perturbations. It is essentially an implicit contrastive learning method, comparing different Dropout perspectives of the same input to encourage the model to learn stable representations.

The advantages of the R-Drop contrastive learning method include: (1) Enhanced robustness: the model is less sensitive to Dropout perturbations, resulting in more stable outputs. (2) No need for complex data augmentation: only Dropout is used to generate different perspectives, simplifying the process. (3) Broad applicability: suitable for tasks such as classification, generation (e.g., NMT), and representation learning.

The difference between R-Drop contrastive learning and ordinary Dropout: Ordinary Dropout: only prevents overfitting by randomly deactivating neurons and is turned off during testing. R-Drop: explicitly constrains the consistency of outputs from different Dropout paths during training, strengthening regularization effects and improving model stability. The computational efficiency of R-Drop contrastive learning: Double forward propagation: each sample requires two forward computations, increasing training time by approximately double, but can be optimized through parallelization. Performance trade-off: experiments show that in NLP tasks, BLEU scores improve by 1-2 percentage points, with performance gains significantly outweighing computational costs. Typical application scenarios for R-Drop contrastive learning include: Text classification: improving model robustness to noise. Machine translation: forcing the decoder to generate consistent outputs under different Dropout settings, improving translation quality. Semi-supervised learning: combining unlabeled data with consistency loss for self-training. R-Drop provides a new approach to regularization by integrating Dropout and contrastive learning, enhancing model performance in a simple and efficient manner. Its core idea is to generate contrastive signals through internal model perturbations rather than relying on external data augmentation.

3. Model Design

3.1. Model Structure

This paper uses the Child-Tuning model [16] and the optimized bert-base-cased [17] pre-trained model as the encoder. The encoder's structure is complex, with bert-base-cased being a classic pre-trained language model based on the Transformer encoder. Bert-base-cased is the case-sensitive version of BERT-Base, with the tokenizer retaining English case information. It contains 12 layers of Transformer encoders, with the following structure:

- (1) Input embedding: converts text into token, segment, and position embedding vectors. Token Embeddings: word vectors using Word Piece tokenization, with a vocabulary size of 30,522. Segment Embeddings: distinguishes sentence pairs, such as sentence A and sentence B. Position Embeddings: position encoding with a maximum sequence length of 512. Layer Normalization: standardizes the input.
- (2) Transformer encoder layer: each layer contains the following components: Multi-head self-attention mechanism (MHSA), number of attention heads is 12, the number of dimension of each head is 64, which is calculated by $768 / 12 = 64$.
- (3) Feed-forward neural network (FFN): the number of intermediate layer dimension is 3072, hidden layer dimension: expands from 768 layers to 3072 layers, then mapping back to 768 layers, and each sub layer is applied by residual connection and layer normalization.
- (4) Output: the number of output vector dimension for each token is 768, the [CLS] token vector from the last layer can be used for classification tasks, the key parameters of bert-base-cased are shown in Table 1.

BERT is pre-trained through the following tasks: (1) Masked language modeling (MLM): randomly masks input tokens with a probability of 15%, and the model predicts the masked tokens. (2)Next sentence prediction (NSP): if inputs a sentence, which is a pairs with A and B, and predicts whether B is the next sentence of A.

The characteristics of bert-base-cased are summarized as follows: (1) Case-sensitive: the tokenizer retains English case information. (2) Bidirectional context: captures bidirectional semantics through the Transformer encoder. (3) Generality: it can be fine-tuned for various NLP tasks, such as text classification, NER, and question answering.

Table 1. Key Parameters of bert-base-cased.

Parameter	Value
Number of layers (Layers)	12
Hidden layer dimension (Hidden Size)	768
Number of attention heads (Attention Heads)	12
FFN intermediate layer dimension	3072
Maximum sequence length	512
Number of parameters	110M

This paper uses the Child-Tuning optimized bert-base-cased pre-trained model as the encoder and designs a Biaffine model [18] combining a global pointer network and contrastive learning to complete the relation classification task. The model identifies the start and end positions of entities in sentences through training, enabling better classification of overlapping entity pairs in complex scenarios. This paper uses the improved R-Drop [19] contrastive learning method for experiments with the Biaffine model, constructing positive sample entity pairs through the Dropout module. Figure 2 shows the framework of the proposed model.As shown in Figure 2, the sentence "Modi and Mahajan are leaders in India where Amdavad ni Gufa is located" is input into the Bert Encoder for Subject Module to form an entity pair module. After R-Drop contrastive learning processing, which is input into the Biaffine module where KL divergence loss and cross entropy loss are calculated, then the output is divided sub objects, then form entity pairs and perform relationship extraction.

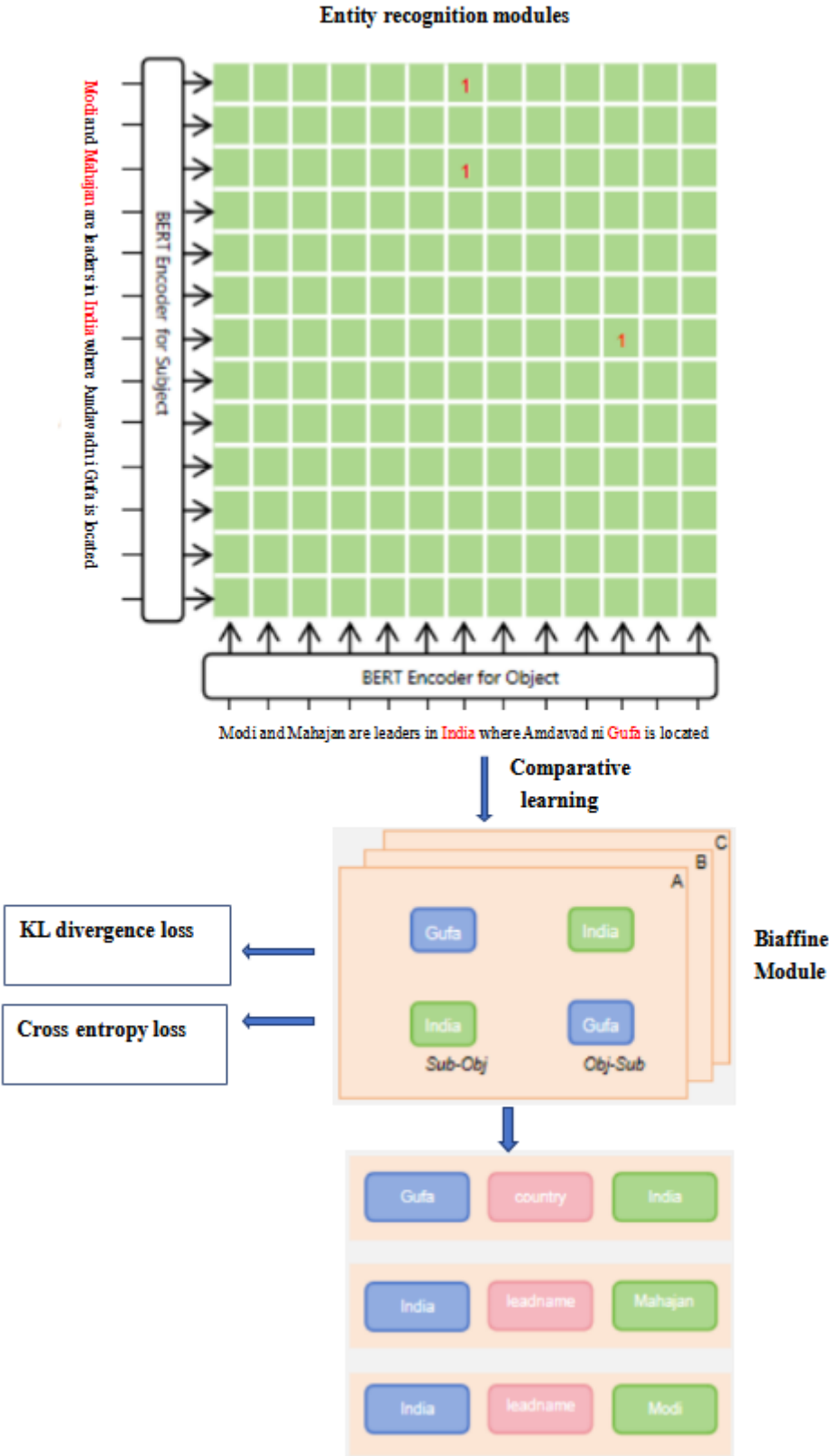


Figure 2. Framework of the Proposed Model.

3.2. Child-Tuning_D Optimized Bert-Base-Cased Pre-Trained Model

In previous work, the performance of models in downstream tasks varied significantly due to the parameter scale of pre-trained models, leading to overfitting. Xu et al. [16] proposed the Child-Tuning method, which updates part of the network parameters in pre-trained models and calculates

Fisher information, effectively reducing overfitting. This paper incorporates Child-Tuning into bert-base-cased and uses the Adam optimizer for fine-tuning. D is the total number of samples, and x, y are the input and output.

$$F^{(i)}(w) = \frac{1}{|D|} \sum_{j=1}^{|D|} \left(\frac{\partial \log p(y_j | x_j; w)}{\partial w^{(i)}} \right)^2 \quad (2)$$

3.3. Using R-Drop Contrastive Learning to Improve Biaffine Method

The Biaffine model based on the global pointer network is widely used in relation classification tasks. The model constructs a two-dimensional table of head and tail entity pairs in sentences and calculates the probability of their corresponding relationships, as shown in Figure 2. In complex scenarios, the model is easily affected by imbalanced relation labels, leading to errors in relation label classification.

This paper uses the R-Drop contrastive learning method to construct a two-dimensional table of positive sample entity pairs, making the semantics of positive sample entity pairs more similar, thereby enhancing the robustness of the Biaffine model when facing imbalanced relation labels.

This paper performs two custom Dropout operations on the head and tail entity pair vectors a and c of the pre-trained model to obtain positive sample vector combinations b and c . The KL divergence is used to weight the sum of the predicted probability distributions of the ab and ac combinations. The loss function L_1 is as follows, where $P_{\theta}^{(a,b,c)}(y_i, |x_i)$ represents the loss of the abc samples.

$$L_1 = -\log P_{\theta}^{(a)}(y_i, |x_i) - \log P_{\theta}^{(b)}(y_i, |x_i) - \log P_{\theta}^{(c)}(y_i, |x_i) \quad (3)$$

The KL divergences of the ab and ac entity pair combinations are L_2 and L_3 respectively. The total loss function is L , where α and β are the hyper parameters of the two contrastive learning processes.

$$L_2 = \frac{1}{2} [\text{KL}(P_{\theta}^{(a)}(y_i, |x_i) | P_{\theta}^{(b)}(y_i, |x_i)) + \text{KL}(P_{\theta}^{(b)}(y_i, |x_i) | P_{\theta}^{(a)}(y_i, |x_i))] \quad (4)$$

$$L_3 = \frac{1}{2} [\text{KL}(P_{\theta}^{(a)}(y_i, |x_i) | P_{\theta}^{(c)}(y_i, |x_i)) + \text{KL}(P_{\theta}^{(c)}(y_i, |x_i) | P_{\theta}^{(a)}(y_i, |x_i))] \quad (5)$$

$$L = L_1 + \alpha L_2 + \beta L_3 \quad (6)$$

4. Experimental Results

4.1. Datasets and Evaluation Indicators

This paper conducted experiments on the widely used NYT [20] and WebNLG [21] datasets, which contain multiple overlapping types of triplets divided into Normal, EPO, and SEO classes, as shown in Table 2. This paper adopts the general model performance evaluation indicators of recall rate R , accuracy rate P , and F1 value, and the calculation formula is:

$$R = \frac{TP}{TP + FN} \quad (7)$$

$$P = \frac{TP}{TP + FP} \quad (8)$$

$$F_1 = \frac{2RP}{R + P} \quad (9)$$

TP represents the number of correctly predicted triples in the test set, FN represents the number of triples that were not predicted in the test set, and FP represents the number of triples incorrectly predicted by the model. Accuracy P represents the proportion of correctly predicted triples in all

predicted triples, recall R represents the proportion of correctly predicted triples in the test set, and F1 is a comprehensive evaluation metric.

The NYT dataset contains 1.18 million sentences and 24 predefined relation types. The WebNLG dataset is derived from tasks created for natural language generation. The test set of NYT10 predominantly consists of the "normal" category, and to validate the model's generalization capability, experiments were conducted on NYT10. There are two different versions of NYT and WebNLG, based on the following two annotation criteria: (1) annotating the last token of each entity, and (2) annotating the entire entity span. To further investigate the proposed model's ability to handle complex scenarios, the test set was divided according to triple overlap patterns and the number of triples. Detailed statistical results of the two datasets are shown in Table 2. EPO, SEO, HTO, and Normal refer to entity pair overlapping triples, single entity overlapping triples, entity pair overlapping (multi-relation) triples, and non-overlapping triples, respectively.

Table 2. Statistics of two datasets based on data types.

Category	Dataset		Details of Test Set										
	Train	Vaild	Test	Relations	Normal	SEO	EPO	HTO	N=1	N=2	N=3	N=4	N≥5
NYT*	56,195	4,999	5,000	24	3,266	1,297	978	45	3,244	1,045	312	291	108
WebNLG*	5,019	500	703	171	245	457	26	84	266	171	131	90	45
NYT	56,195	5,000	5,000	24	3,222	1,273	969	117	3,240	1,047	314	290	109
WebNLG	5,019	500	703	216	239	448	6	85	256	175	138	93	41
NYT10	70,339	351	4,006	29	2,963	742	715	185	2,950	595	187	239	35

4.2. Experimental Parameters

The pre trained model used in this experiment is BERT [17]. Set the batchsize to 6 on WebNLG and 18 on NYT, with a learning rate of $3e-5$. The maximum sequence length is set to 100. Run the model 5 times on all datasets and take the average as the final result, The experimental hyperparameters are shown in Table 3.

Table 3. Open dataset experiment hyperparameter settings.

datasets	learning rate	batchsize	pre trained model
WebNLG	$3e-5$	6	bert-base-cased
WebNLG*	$3e-5$	6	bert-base-cased
NYT	$5e-5$	18	bert-base-cased
NYT*	$5e-5$	18	bert-base-cased
NYT10	$5e-5$	18	bert-base-cased

4.3. Experimental Results

4.3.1. Comparative Experiment

The experimental results of the model designed in and the baseline model on the NYT and WebNLG datasets are shown in Table 4. The Bert-CL-Biaffine model has shown improvements on both datasets. Compared to the baseline model CasRel, Bert-CL-Biaffine outperforms by 4.0% and 4.6% in F1 score on NYT and WebNLG, respectively. When compared to TPLinker, Bert-CL-Biaffine shows an improvement of 1.0% and 4.3% in F1 score on NYT and WebNLG, respectively. Additionally, compared to BIRTE, Bert-CL-Biaffine achieves a 1.7% higher F1 score on WebNLG. This improvement is attributed to the integration of the model and the incorporation of contrastive learning methods into the Biaffine relation classification model, leading to enhanced performance.

Table 4. Comparison experimental results between Bert-CL-Biaffine model and the baseline model.

Model	NYT			WebNLG		
	Prec.	Rec.	F1	Prec.	Rec.	F1
ETL-Span [1]	85.5	71.7	78.0	84.3	82.0	83.1
RIN [2]	83.9	85.5	84.7	77.3	76.8	77.0
CasRel [7]	89.8*	88.2*	89.0*	88.3*	84.6*	86.4*
TPLinker [4]	91.4	92.6	92.0	88.9	84.5	86.7
PMEI [3]	88.4	88.9	88.7	80.8	82.8	81.8
PRGC [9]	93.5	91.9	92.7	89.9	87.2	88.5
BIRTE [8]	91.9	93.7	92.8	89.0	89.5	89.3
Bert-CL-Biaffine	93.8	92.0	93.0	91.5	89.1	90.3

4.3.2. Experiments and Analysis in Complex Scenarios

To evaluate the performance of the proposed Bert CL Biaffine model in complex scenarios, experiments will be conducted on sentences with different overlap types and triplet counts. The experimental data of the test set was split by type and quantity to evaluate the ability of our model in processing complex types of data. The experimental results are shown in Table 5. On the NYT * dataset, the model improved by 1.3% on EPO type datasets and 0.6% on SEO type datasets. On the WebNLG * dataset, the model in this chapter improved by 2.3% on the EPO type dataset, indicating an improvement in performance when facing overlapping entity pairs.

Table 5. Experimental comparison of overlapping types with different entities.

Model	NYT*			WebNLG*		
	Normal	EPO	SEO	Normal	EPO	SEO
CasRel [7]	87.3	92.0	91.4	89.4	94.7	92.2
PRGC [9]	91.0	94.5	94.0	90.4	95.9	93.6
BIRTE [8]	91.4	94.2	94.7	90.1	94.3	95.9
Bert-CL-Biaffine	91.1	95.5	95.3	90.7	96.6	95.9

The experiments with different numbers of triples are shown in Table 6. The model showed significant improvement in sentences with multiple triples compared to the baseline model on both datasets, indicating that the model can better handle complex situations.

Table 6. Experimental comparison of different triplet numbers.

Model	NYT*					WebNLG*				
	N=1	N=2	N=3	N=4	N≥5	N=1	N=2	N=3	N=4	N≥5
CasRel [7]	88.2	90.3	91.9	94.2	83.7	89.3	90.8	94.2	92.4	90.9
PRGC [9]	91.1	93.0	93.5	95.5	93.0	89.9	91.6	95.0	94.8	92.8
BIRTE [8]	91.5	93.7	93.9	95.8	92.1	90.2	92.9	95.7	94.6	92.0
Bert-CL-Biaffine	91.0	94.7	94.8	96.9	93.7	90.9	93.9	96.4	95.0	93.7

4.3.3. Experiment and Analysis of Relationship Classification Subtasks

In order to further verify the ability of the proposed model to perform relationship classification on overlapping entity pairs, a comparative experiment was designed on the relationship classification subtask. By analyzing the prediction results of the test set, the data of erroneous relationship classification on overlapping entity pairs were separated and statistically calculated. The experimental results are shown in Table 7. Among the relationship classification results on the three datasets, the model improved the F1 of the baseline model by 2.4% and 1.6% in NYT * and WebNLG *, respectively.

Table 7. Experimental results of model and baseline model in relation classification subtask.

Model	Element	NYT*			WebNLG*		
		Prec.	Rec.	F1	Prec.	Rec.	F1
CasRel	r	96.0	93.8	94.9	96.6	91.5	94.0
	(h,r,t)	89.7	89.5	89.6	93.4	90.1	91.8
PRGC	r	95.3	96.3	95.8	92.8	96.2	94.5
	(h,r,t)	93.3	91.9	92.6	94.0	92.1	93.0
BiRTE	r	95.8*	97.1*	97.0*	93.7*	94.3*	94.0*
	(h,r,t)	92.2*	93.8*	93.0*	93.2*	94.0*	93.6*
Bert-CL-	r	96.5	97.7	97.1	96.0	95.1	95.6
Biaffine	(h,r,t)	93.5	92.7	93.2	94.2	94.0	94.1

4.3.4. Ablation Experiment

In order to verify the effectiveness of each module of the model, ablation experiments were conducted, where “-” represents removal, - Child-Tuning_D indicates the removal of this method, while - contrastive learning indicates the removal of contrastive learning methods. The experimental results are shown in Table 8, and removing either method will have an impact on the model performance. And to verify the effectiveness of the contrastive learning method in this paper, the R-drop contrastive learning method is set as the control experiment.

Table 8. Model ablation experiment.

Model	NYT			WebNLG		
	Prec.	Rec.	F1	Prec.	Rec.	F1
-Bert-Biaffine	90.8	93.2	92.0	89.4	88.8	89.1
-Child-Tuning_D	91.1	94.4	92.7	90.8	89.2	90.0
-Contrastive Learning	91.7	93.3	92.5	90.7	88.9	89.8
R-drop	91.2	93.4	92.3	90.5	88.5	89.5
Bert-CL-Biaffine	92.2	93.8	93.0	91.5	89.1	90.3

In order to further verify the ability of each module of the model in handling overlapping entity pairs, based on the ablation experiment, the experimental data of the test set was split by type and quantity to evaluate the ability of the model in handling complex type data. The experimental results are shown in Table 9.

Table 9. Experimental results of different overlap types of Bert-CL-Biaffine model on NYT.

Model	SEO			EPO			Normal		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
-Child-Tuning_D	95.2	91.9	93.5	95.3	92.7	94.0	89.5	92.6	91.0
-Contrastive Learning	94.2	91.8	93.0	94.7	91.8	93.2	89.6	91.4	90.5
Bert-CL-Biaffine	97.1	91.7	94.3	96.8	92.7	94.7	90.5	93.0	91.5

5. Conclusion and Future Work

This paper uses contrastive learning methods to improve the relationship classification performance of the Biaffine model on overlapping entity pairs. The model can better understand the semantic associations between overlapping entity pairs in sentences, thereby enhancing the overall ability of the joint model to extract ternary groups from overlapping entity pairs. The experimental results show that the Biaffine relationship classification model based on global pointer network and contrastive learning method proposed in this paper significantly improves the relationship classification performance on NYT and WebNLG datasets. Compared to the baseline model, this

model performs better in dealing with imbalanced relationship types and unclear feature information of entities, especially in complex scenarios where the classification performance is significantly improved. This indicates that bidirectional entity contrastive learning can effectively enhance the feature representation of entity pairs, thereby improving the overall performance of relationship classification tasks.

Future work can be carried out from the following aspects: (1) Exploring more effective contrastive learning architectures to enhance the feature representation of entity pairs, combining multimodal data such as text and images to enhance the richness of entity representation. (2) Develop new data augmentation methods to improve model generalization ability, use generative models to create high-quality synthetic data, and expand the training set. (3) Research the adaptability of models in different fields, improve cross domain performance, use pre trained models for transfer learning, and reduce dependence on annotated data. Develop interpretable methods to help understand the decision-making process of the model, and visualize the feature representations and relationship classification results of entity pairs.

Author Contributions: S. H. conceived of and designed the study, H. W. performed the experiments, and analyzed the data. Z. Z. and L. J. revised and polished the manuscript. All authors have read and agreed to the published version of the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Discipline (Professional) Leader Cultivation Project of Anhui Province. Grant number: DTR2024039 and The APC was funded by Program for Scientific Research Innovation Team in Colleges and Universities of Anhui Province, grant number:2022AH010095.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to express their sincere thanks to the editor and the anonymous reviewers for their valuable comments and suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ke X, Wu X, Ou Z, et al. Chinese Named Entity Recognition method based on multi-feature fusion and biaffine[J]. *Complex & Intelligent Systems*,2024,10(5):6305-6318.
2. Xiao X, Gao B,Su Z , et al. STBA: span-based tagging scheme with biaffine attention for enhanced aspect sentiment triplet extraction[J].*Pattern Analysis and Applications*,2024,27(4):145-145.
3. Aziz K, Ahmed N, Hadi J H, et al. Enhanced UrduAspectNet: Leveraging Biaffine Attention for superior Aspect-Based Sentiment Analysis[J]. *Journal of King Saud University - Computer and Information Sciences*,2024,36(9):102221-102221.
4. Piao Y, Zhang X J. Text Triplet Extraction Algorithm with Fused Graph Neural Networks and Improved Biaffine Attention Mechanism[J]. *Applied Sciences*,2024,14(8):3524-.
5. Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. Kernel methods for relation extraction. *J. Mach. Learn. Res.*, 3:1083–1106, 2003.
6. Chan, Yee Seng, and Dan Roth. "Exploiting syntactico-semantic structures for relation extraction", *HLT '11: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1* (2011): 551-560.
7. Zheng, Suncong et al. "Joint Extraction Of Entities And Relations Based On A Novel Tagging Scheme", *Annual Meeting of the Association for Computational Linguistics* abs/1706.05075. (2017): 1227-1236.

8. Wei, Zhepei et al. "A Novel Cascade Binary Tagging Framework For Relational Triple Extraction", 58TH ANNUAL MEETING OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS (ACL 2020) 2020.acl-main (2020): 1476-1488.
9. Zheng, Hengyi et al. "PRGC: Potential Relation and Global Correspondence Based Joint Relational Triple Extraction.", Annual Meeting of the Association for Computational Linguistics abs/2106.09895 (2021): 6225-6235.
10. Ren, Feiliang et al. "A Simple but Effective Bidirectional Extraction Framework for Relational Triple Extraction", semanticscholar abs/2112.04940 (2021): 824-832.
11. LCSL. "ICLR 2015: International Conference on Learning Representations 2015." (2017).
12. TANG, Zhan, et al. Flat and Nested Protein Name Recognition Based on BioBERT and Biaffine Decoder. In: International Symposium on Bioinformatics Research and Applications. Singapore: Springer Nature Singapore, 2024. p. 25-38.
13. Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).
14. Yang, Zonghan et al. "Reducing Word Omission Errors in Neural Machine Translation: A Contrastive Learning Approach.", Annual Meeting of the Association for Computational Linguistics (2019): 6191-6196.
15. Wu, Lijun, et al. "R-drop: Regularized dropout for neural networks." Advances in neural information processing systems 34 (2021): 10890-10905.
16. Xu, Runxin et al. "Raise a Child in Large Language Model: Towards Effective and Generalizable Fine-tuning", Conference on Empirical Methods in Natural Language Processing (2021)
17. Devlin, Jacob et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", North American Chapter of the Association for Computational Linguistics abs/1810.04805 (2018): 4171-4186.
18. Dozat, Timothy, and Christopher D Manning. "Deep Biaffine Attention for Neural Dependency Parsing", International Conference on Learning Representations (2017)
19. liang, xiaobo et al. "R-Drop: Regularized Dropout for Neural Networks", Computing Research Repository (2021)
20. Riedel, S.; Yao, L.; and McCallum, A. 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, volume 6323, 148–163. Springer.
21. Gardent, C.; Shimorina, A.; Narayan, S.; and PerezBeltrachini, L. 2017. Creating Training Corpora for NLG Micro-Planners. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 179–188. Association for Computational Linguistics.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.