applied sciences

MDPI

Article

FA-RCNet: A Fused Feature Attention Network for Relationship Classification

Jiakai Tian †, Gang Li †, Mingle Zhou, Min Li * and Delong Han

Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan 250316, China

* Correspondence: limin@qlu.edu.cn

† These authors contributed equally to this work.

Abstract: Relation extraction is an important task in natural language processing. It plays an integral role in intelligent question-and-answer systems, semantic search, and knowledge graph work. For this task, previous studies have demonstrated the effectiveness of convolutional neural networks (CNNs), recurrent neural networks (RNNs), and long short-term memory networks (LSTMs) in relational classification tasks. Recently, due to the superior performance of the pre-trained model BERT, BERT has become a feature extraction module for many relational classification models, and good results have been achieved in work related to BERT. However, most of such work uses the deepest levels of features. The important role of shallow-level information in the relational classification task is ignored. Based on the above problems, a relationship classification network FA-RCNet (fusion-attention relationship classification network) with feature fusion and attention mechanism is proposed in this paper. FA-RCNet fuses shallow-level features with deep-level features, and augments entity features and global features by the attention module so that the feature vector can perform the relational classification task more perfectly. In addition, the model in this paper achieves advanced results on both the SemEval-2010 Task 8 dataset and the KBP37 dataset compared to previously published models.

Keywords: relationship classification; attentional mechanisms; feature fusion

Citation: Tian, J.; Li, G.; Zhou, M.; Li, M.; Han, D. FA-RCNet: A Fused Feature Attention Network for Relationship Classification. *Appl. Sci.* **2022**, *12*, 12460. https://doi.org/10.3390/app122312460

Academic Editor: Pengjie Ren

Received: 3 November 2022
Accepted: 1 December 2022
Published: 6 December 2022

1. Introduction

Relation classification [1,2] is an important task in natural language processing. In natural language processing task architectures, relational classification is an important antecedent between tasks, such as document summarization [3], question-and-answer systems [4], machine translation [5], and knowledge graphs [6]. The relationship classification task is a supervised task. The main objective is to extract relation information from a sentence with an entity identifier, determine the relationship between two entities in a sentence, and classify the relationship. Take the sentence in Table 1 as an example, the sentence includes <e_1>, </e_1>, <e_2>, </e_2>, which are four position markers. Tags are used to mark the exact location of entities in a sentence. The example sentence is labeled with two entities "elephant" and "animal". The task of relationship classification is to identify the relationship between the two entities as entity-origin(e_1, e_2).

Table 1. Sample sentences.

Sentence: "The <e_1> elephant </e_1> descended from an aquatic <e_2> animal </e_2>."		
E_1: elephant	E_2: animal	Relation: Entity-Origin(e_1, e_2)

Earlier traditional methods of relational classification used kernel-based approaches [7–9]. Kernel-based algorithms mainly include support vector machine (SVM), radial basis func-

tion (RBF), and linear discriminate analysis (LDA). SVM is mainly used in relation classification tasks to classify entity relations. SVM is a classification model that solves binary classification problems through supervised learning. Such methods are of course helpful to improve the accuracy of relational classification tasks. However, SVM is based on the binary classification of the relationship based on the kernel function. Not all features can be divided according to the kernel function. When the relationship between entities is complex or there are multi-category relationships between entities, the error probability will increase. Therefore, the kernel-based relationship classification method has problems, such as a high error rate, low classification accuracy, and serious feature loss. Relational classification methods based on neural networks and deep learning have become a hot issue for research in recent years [10]. Such methods require only simple pre-work or even methods without pre-work for automatic learning of feature parameters, such as convolutional neural network (CNN), recurrent neural network (RNN), and long short-term memory network (LSTM), and neural networks based on attention mechanisms. In recent years, research has mainly been based on CNN and RNN-based relationship classification models. Such models typically use the traditional form of mapping literals to vectors. After the pre-training model was published, the focus of research gradually shifted to the pre-training model. The latest models basically use the feature output by the pre-trained model for downstream tasks. However, when using the pre-training model, the feature output of the last layer is habitually used for tasks, resulting in the importance of shallow features being ignored. The above model surpasses the traditional model in terms of relational classification effect but still has the following problems.

First, the shallow features are ignored. The existing models [11–14] retain only the deep features extracted by the neural network. Deep features have rich semantic information and are more suitable for performing relational classification tasks. However, shallow-level features have richer fine-grained features and clearer location information [15]. The shallow information can be used as auxiliary information in the classification network to enhance the effective part of the deep features.

Second, only the semantic information of whole sentences is extracted and analyzed in existing networks. In this paper, we argue that in addition to the overall features of the whole sentence, the features corresponding to each entity are also important for the relational classification task. By fitting the feature vectors corresponding to the two entities to the overall vector of the whole sentence, in-depth, not only the quality of the features can be improved, but also the influence of irrelevant words in the sentence on the classification network can be weakened. In response to the above issues and challenges, the main contributions of this paper can be summarized as follows:

(1) In this paper, we propose a model FA-RCNet for relational classification tasks to improve the effectiveness of relational classification models through feature fusion and channel attention mechanisms.

(2) A feature fusion module is proposed to combine different hidden layer features in the BERT model with sentence features, realize the combination of shallow fine-grained features and deep abstract features, and further enrich the semantic information in the feature vector.

(3) An attention module is proposed. This attention module analyzes the importance of each semantic feature. Weights are assigned according to the degree of importance to enhance the role of important semantic information in the model and reduce the negative impact of irrelevant semantic noise on the model.

(4) Through experimental tests, the relationship classification model proposed in this paper achieves good results on the SemEval-2010 Task8 dataset and the KBP37 dataset. The accuracy of recognition in different directions of the same relationship was also improved on the SemEval-2010 Task8 dataset.

The remainder of this paper is structured as follows. Section 2 provides an overview of the work involved. Section 3 presents the general architecture and details of the relational classification model proposed in this paper. The data set, hyperparameters, and

environment of the experiments, as well as the experimental results and analysis, are given in Section 4. Section 5 provides a summary and outlook on the work of this paper.

2. Related Work

Relational classification is an important part of the natural language processing task. Neural networks are used in relational classification tasks to take advantage of the fact that they can automatically learn data features. No manual setting of features is required. Among the existing neural network models for relational classification, there are primarily relational classification networks based on recurrent neural networks (RNNs), convolutional neural networks (CNNs), and attention-based relational classification networks.

2.1. CNN-Based Relational Classification Model

CNNs are characterized by local features of the elements and they improve model performances through the extraction of local features. A multi-window CNN model consisting of convolutional kernels of different sizes was proposed by Nguyen et al. [16]. The model has a much-improved ability to extract local information compared to a single-size convolutional kernel, thus improving the effectiveness of the model. The CR-CNN relational classification model was proposed by Socher et al. [17]. In their work, they redefined the loss function and calculated the loss by a two-by-two ranking. Wang et al. [18]. applied the attention mechanism to CNN relational classification networks and proposed a multi-attention-based CNN model. The model focuses more on the relationships between words in a sentence while analyzing the correspondence between tags and sentences for relational classification tasks.

2.2. RNN-Based Relationship Classification Model

Recurrent neural networks are better at processing time-series information than convolutional neural networks [19] and are more concerned with contextual information. The RNN-based model outperforms the CNN-based model when the sentences in the dataset are long. Li et al. [20] vectorized the data by RNN and improved the classification accuracy by obtaining semantic features related to entities in the context of the relationship classification task. Zhang and Wang et al. [21] proposed a bidirectional RNN model for extracting sentence features. Feature transformation of sentence contextual information by positional tagging has achieved relatively good results in relational classification models. Lee et al. [22] proposed an end-to-end RNN model for relationship classification and added entity-aware mechanisms to the BiLSTM followed by a relational classification task (in combination with the entity type).

2.3. Classification Models Based on Attention Mechanisms

Attentional mechanisms [23,24] take many forms, ranging from multiple self-attention, hard attention, and soft attention. The self-attentive mechanism [25,26] aims to allow the model to automatically identify the relevance of different parts of the input. Hard attention [27] is a stochastic prediction process, distinguishing whether relevant features are noted by 0–1 classification and the process is not trivial, whether relevant features are noted by the 0–1 classification, and whether the process is non-differentiable. Soft attention is divided into spatial attention [28], channel attention [29], mixed attention (spatial attention combined with channel attention) [30], and positional attention [31]. In this paper, channel attention includes GSoP-Net [32], ECANet [33], EncNet [34], NAM [35], etc. The core idea of channel attention is to enhance the role of effective features in the task by assigning large weights to effective channels through parameters in the weight assignment module. There are also attention mechanisms specific to the NLP domain [18,36,37]. As a result of BERT [38], numerous BERT-based fine-tuning models have emerged in the field of NLP, all with good results.

Attention-based relational classification models have also achieved good results due to the effectiveness of the attention mechanism. Zhou et al. [36] combined the attention

mechanism with BiLSTM. The sentence is transformed into a feature vector by BiLSTM, and the weight of each word in the whole sentence is calculated by the attention mechanism. The aim is to obtain word vectors that are favorable for the relational classification task. Xu et al. [11] proposed to improve the performance of relational classification models by identifying important phrases through attention to the output features of BERT fed into an important phrase extraction network. Geng et al. [39] applied attention to CNNs by adding the influence of the relationship matrix weights of two entities in a sentence while ignoring the calculation of irrelevant terms. Liu and Guo et al. [37] proposed an application of the attention mechanism to BiLSTM networks with convolution. By extracting global and local semantics, it is the model that can better understand contextual information. Wu et al. [12] proposed a relational classification network based on the BERT pre-training model. The BERT pre-training model uses an attention mechanism for the extraction of word features. Improve network performance by splicing global features with entity features. Li et al. [40] proposed a relational classification model based on a hybrid attention and confusion loss function. Classification accuracy is improved by fusing information from head and tail entities with entity word-level attention and designing dedicated confusion loss functions. The first use of attention blocks for a sentence encoder was made by Yan et al. [41]. It primarily uses multi-headed self-attention to capture word-level grammatical information.

The channel attention mechanism is more widely used in the field of computer vision and has also yielded good results. Little work has applied channel attention to the task of relational classification in natural language processing. In this paper, channel attention is used in the enhancement process of features with some success.

3. Methodology

The most recent relational classification models use BERT, a pre-trained model, for sentence feature extraction. Then the sentence features are further processed to obtain the classification probability for performing classification tasks. The BERT pre-trained model is applied to a variety of downstream tasks in natural language processing including relational classification. BERT-GMAN [11] proposed a relationship extraction model based on BERT-gated multi-window attention network, which achieved good results in the Semeval-2010 Task 8 dataset. BertSRC [42] proposed to extract the entity relationship in the medical field based on BERT and achieved good results in the data set of the proprietary field. AugFake-BERT [43] proposes a BERT-based data augmentation model. Using the enhanced data set for model training can obtain better model weights. CRSAtt [44] proposes a BERT-based relation classification model. CRSAtt processes BERT output features by fusing sentences and entity features. At the same time, the attention mechanism is used instead of the fully connected layer to predict the relationship category. The above work proves the effectiveness of the features extracted by BERT for downstream tasks, so this paper uses BERT as the feature extraction module. Since the attention mechanism was proposed, it has achieved good results in many fields. In computer vision tasks, the visual attention mechanism [45,46] can make the model better notice the key information in the picture. In natural language processing tasks, the attention mechanism [44,47,48] can analyze the relationship between each word in the text and words of different parts of speech through the attention mechanism. The purpose of this is to obtain higher-quality semantic features. Therefore, this paper uses the attention mechanism to further process the feature vector.

3.1. Overview

In this section, a three-layer FA-RCNet is proposed to solve the relationship classification task in NLP. The first layer is the embedding layer. It is used to form word embedding information after fusing word embedding, positional embedding and segmentation embedding of the sentence. The second layer is the feature fusion and attention module. The role is to fuse the feature vectors of each layer of the preprocessing model. At the same time, the attention module captures the rich relational features in the semantic information. The feature weights corresponding to the relationships between entities in the semantic features

are increased. The third layer is the relationship classification layer. The entity feature outputs from the second layer are spliced with the global features. The weight of positive features is enhanced by the attention module. Moreover, the negative impact of negative features on classification accuracy is reduced. Figure 1 shows the general framework of the FA-Net model. The definitions of the symbols used in this paper are shown in Table 2.

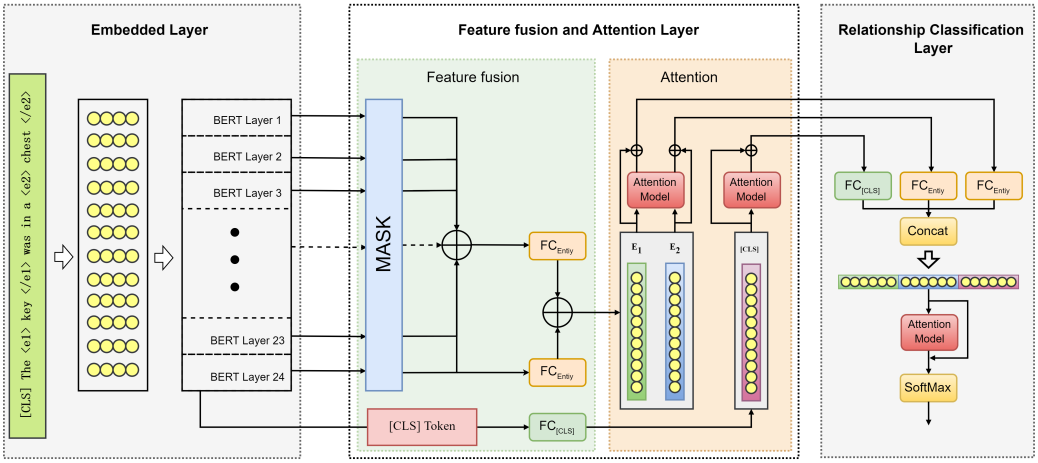

Figure 1. The general architecture of the FA-RCNet model.

Table 2. Symbols and definitions.

Symbol	Definition
E_1,E_2	The two entities correspond to the hidden layer input fusion vectors.
e_1^{24},e_1^{24}	The last hidden layer output of the two entities.
T_1,T_2	E_1,E_2 is summed with the last hidden layer output to obtain the fusion vector.
e_0	The global feature vector output by the last hidden layer.
$e_0^{'}$	The output of e_0 after passing through the fully connected layer.
μ_B,σ_B	The mean and variance of a set of eigenvalues.
T_{1-out},T_{2-out}	Feature amplification attention module's entity feature output.
e_{0-out}	Global feature output of the feature amplification attention module.
$T_{1-out}^{'}$,$T_{2-out}^{'}$	The fully connected layer output of T_{1-out},T_{2-out}.
$e_{0-out}^{'}$	The fully connected layer output of e_{0-out}.
T	The feature vector after cascading.
P	The probability vector used for classification.

3.2. Embedded Layers

In the embedding layer, the data in the dataset are preprocessed and the two entities in the sentence are annotated by "$" and "#", respectively. For example, in the following sentence, entity I corresponds to "woman" and entity II corresponds to "village".

The $ woman $ was born in the # village #.

The pre-processed data are embedded by word embedding, location embedding, and segmentation embedding to obtain the embedding vector. The role of word embedding is to convert the text into vectors. The text information is fed into the feature extraction model in the form of vectors. In this paper, word embeddings obtained from pre-training are used. It was shown that word embeddings obtained by training with unlabeled samples have better results than randomly generated word embeddings in the task of relation extraction and relation classification [35].

The word embedding contains only the information corresponding to a word and does not record the information on the position of the word appearing in the sentence. In this paper, each sentence as a whole is numbered sequentially from 0 according to the order of word occurrences in the sentence. This allows the position of words in the sentence

to be mapped to numbers. The order of the numbers by their size enables the model to accurately identify the position of the words in the sentence.

In a practical relational classification task, two entities may appear in two sentences. The same entity in two sentences may have an impact on the meaning of the entities. Even the different order of appearance of two entities in a sentence may have this problem. In this case, it is necessary to use split embedding to distinguish the two sentences. The purpose is to allow the model to obtain an accurate correspondence between entities and clauses. Figure 2 shows the flow of the embedding layer.

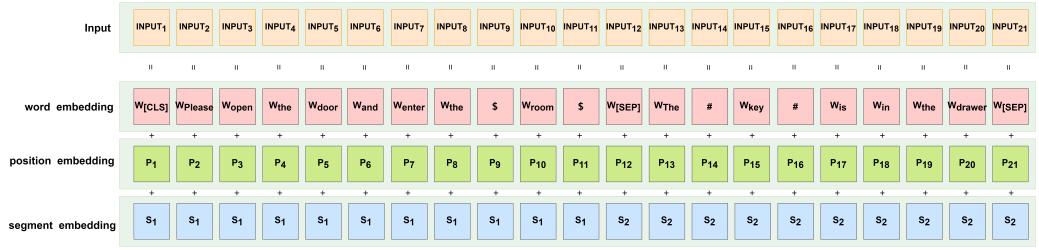

Figure 2. Embedded process.

3.3. Feature Fusion and Attention Layers

After receiving the word embeddings from the output of the embedding layer, the input word vectors are feature extracted using a pre-trained BERT model. Receive the hidden layer output from each layer of the BERT module. Extraction of two entity vectors from the hidden layer output by the mask matrix. The entity vectors from each layer are fused and sent to the attention module, with the aim of increasing the feature weights corresponding to the relationship between the two entities in the feature vectors by the attention module.

3.3.1. Feature Fusion Layer

The word vectors output from the embedding layer are trained by BERT to produce a hidden layer output H_i at each layer. The mask matrix consists of 0s and 1s and is generated automatically by reading the entity identifiers "$" and "#" in the sentence. The number of the position corresponding to the vector between two markers is 1, indicating retention, and the number of all positions outside the vector corresponding to the two entities is 0, indicating discard. Figure 3 shows the Mask operation flow.

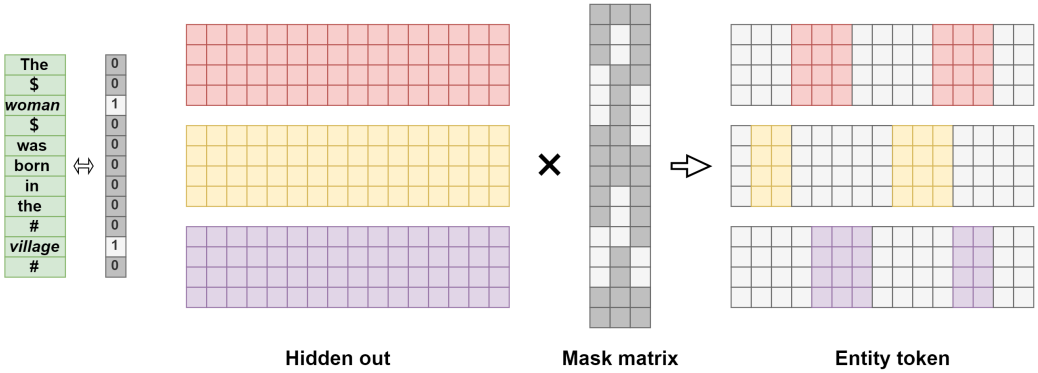

Figure 3. The mask matrix is automatically generated by reading the entity markers, and the mask matrix is multiplied with the hidden layer output vector to extract the entity's corresponding feature vector.

Hidden out is the output of the hidden layer of the BERT model. The mask matrix is the mask matrix for entity feature extraction. The entity token is the set of entity vectors obtained after the mask matrix.

One entity may correspond to more than one word, so one entity may correspond to more than one hidden layer output. Taking the first hidden layer output as an example, the hidden layer outputs corresponding to two entities in a sentence are extracted from H_1 by Mask matrix. This process is expressed mathematically as Equation (1):

$$\left[h_i \cdots h_j, h_k \cdots h_m\right] = \left[\begin{array}{ccc} H_{1-1} & \cdots & H_{1-n} \end{array}\right] \times \left[\begin{array}{c} m_1 \\ \vdots \\ m_n \end{array}\right] \tag{1}$$

where h_i–h_j and h_k–h_m can be represented as the set of vectors corresponding to entity I and entity II in the first hidden layer. H_{1-1} and H_{1-n} denote the first and last elements in the output of the hidden layer. m_1 and m_n denote the first and last elements of the mask matrix.

The set of vectors obtained by extraction is averaged to obtain the feature vectors corresponding to the two entities in the first hidden layer. The 24 hidden layer outputs of the BERT model are masked and averaged separately. The 24 feature vectors are obtained for each of the two entities. In order to make full use of the deep semantic information and the shallow semantic information in the hidden layer output, the sum operation is applied to these 24 feature vector pairs. This allows the semantic information contained in the output of each hidden layer to be combined. The combined vectors are all applied to the word vector generation process. This process can be expressed mathematically as Equations (2) and (3):

$$E_1 = \sum_{a=1}^{24} \left(\frac{1}{j_a - i_a + 1} \sum_{b=i}^{j} h_b \right) \tag{2}$$

$$E_2 = \sum_{a=1}^{24} \left(\frac{1}{m_a - k_a + 1} \sum_{b=k}^{m} h_b \right) \tag{3}$$

where E_1 is the feature sum corresponding to entity I; E_2 is the feature sum corresponding to entity II; i_a-j_a is the starting position of the feature vector corresponding to entity I at layer a; and k_a-m_a are the starting and ending positions of the feature vector corresponding to entity II at layer a.

In the previous relational classification models, only the hidden state output of the last layer or the last two layers of the BERT model was used for relational classification tasks, such as R-bert, etc. It is shown that the deeper feature vectors contain more semantic information and are more suitable for relational classification tasks, so the hidden state of the last layer is output separately as the first feature enhancement module. The classification task requires not only the semantic features of the two entities but also the global features of the whole sentence. In this paper, the global features of the whole sentence are output separately as the second feature enhancement module. An activation function is added to the fused features E_1, E_2, the last layer of the hidden state outputs e_1^{24}, e_2^{24}, and the global feature vector e_0, respectively. A fully connected layer is also connected.

The fusion vector is passed through the fully connected layer to produce a semantic feature vector. The final layer of hidden state output is passed through the fully connected layer to produce a semantic feature vector. The above two are added together to obtain a fusion feature that is more suitable for the classification task. This feature is a feature vector (T_1 and T_1) that focuses on deep semantics and fuses shallow sub-semantics. T_1 and T_2 are sent to the attention module for further feature extraction along with e_0' generated by the global feature vector after the fully connected layer. The above process can be expressed as Equations (4)–(6) using the mathematical formula.

$$T_1 = \left(W_{E_1}[\tanh(E_1)] + b_{E_1}\right) + \left(W_{e_1^{24}}\left[\tanh\left(e_1^{24}\right)\right] + b_{e_1^{24}}\right) \tag{4}$$

$$T_2 = \left(W_{E_2}[\tanh(E_2)] + b_{E_2}\right) + \left(W_{e_2^{24}}\left[\tanh\left(e_2^{24}\right)\right] + b_{e_2^{24}}\right) \tag{5}$$

$$e_0' = W_{e_0}[\tanh(e_0)] + b_{e_0} \tag{6}$$

where E_1 and E_2 are the fused feature vectors after the fully connected layer. e_1^{24} and e_2^{24} are the feature vectors of the last hidden layer output after the fully connected layer. E_0 is the global feature vector of the fully connected layer output. W_i ($i = E_1, E_2, e_1^{24}, e_1^{24}, e_0$) is the weight matrix in the fully connected layer. b_i ($i = E_1, E_2, e_1^{24}, e_1^{24}, e_0$) is the bias term in the fully connected layer.

3.3.2. Attention Layers

The attention modules used in existing relational classification models are all attention mechanisms based on encoder and decoder structures. The weight of each feature is determined by querying the similarity between the features. For the relational classification task, most of the features are extracted using the attention mechanism for all words in a sentence. However, the classification task is the discrimination of the relationship between two entities. Using the above attention mechanism weakens the relationship weights between two entities. To solve the above problem, the feature amplification attention module is proposed in this paper. The mean and variance in a feature vector are calculated and the importance of the feature vector is determined by the scale factor in the BN [49,50] mechanism. The feature values are also normalized in the linear part of the nonlinear function. The feature values are then scaled by a weight scaling factor $\frac{\gamma_i}{\sum_{j=0}\gamma_j}$. The purpose is to allow positive features to be feature-enhanced and negative features to be suppressed. This process can be expressed mathematically as Equation (7):

$$x_i' = BN(x_i) = \gamma\frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} + \beta \tag{7}$$

where γ and β are trainable affine transform parameters. x_i is the input to the BN layer. x_i' is the output of the BN layer. μ_B is the mean of the elements in a feature vector. σ_B is the variance of the elements in a feature vector with a constant $\epsilon > 0$ to ensure that it does not divide by zero. Figure 4 shows the feature amplification attention module.

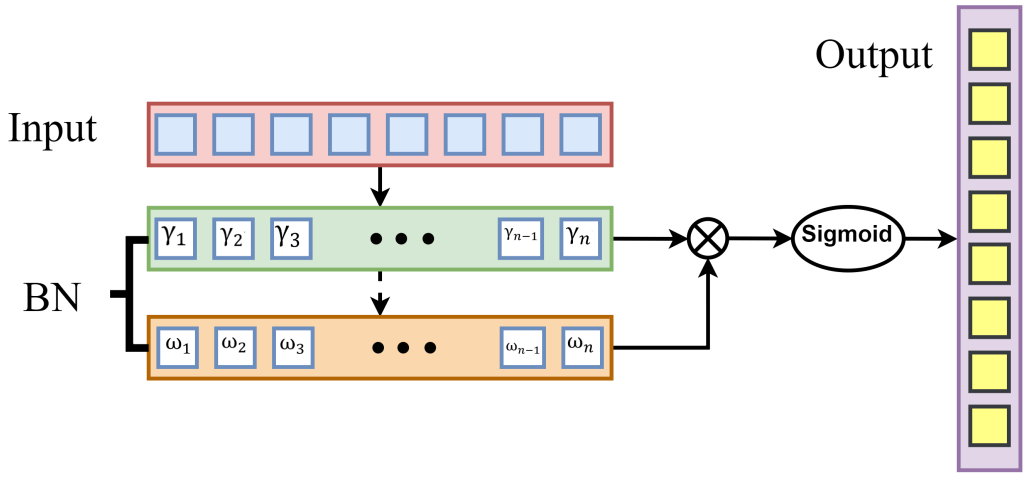

Figure 4. Channel Attention Modules.

In feature amplification attention, the feature vector of a single entity and the global feature vector containing the contextual information of the whole sentence are fed into the attention module, and each element of the feature vector is mapped to a single feature value. Through the attention mechanism, not only the important features in the global feature can be amplified to make the contextual information richer, but also the relationship between two entities can be better fitted. This not only preserves more positive semantic information

but also enhances the sensitivity of the model to the directionality of the relationship due to the further enrichment of contextual information.

Meanwhile, an attention residual module is designed in this paper. The input and output of the attention module are added with the purpose of further increasing the weights of the effective features while retaining the original features. In this way, all features can be involved in the subsequent learning and updating of the network. Another role of the residual network is to effectively avoid the problems of gradient explosion and gradient disappearance when the number of layers of the neural network increases. The above process can be expressed as Equations (8)–(10):

$$T_{1-\text{out}} = \text{Sigmoid}\left(\frac{\gamma_{T_1-i}}{\sum_{j=0}\gamma_j} BN(T_1)\right) + T_1 \tag{8}$$

$$T_{2-\text{out}} = \text{Sigmoid}\left(\frac{\gamma_{T_2-i}}{\sum_{j=0}\gamma_j} BN(T_2)\right) + T_2 \tag{9}$$

$$e_{0-\text{out}} = \text{Sigmoid}\left(\frac{\gamma_{e_0-i}}{\sum_{j=0}\gamma_j} BN(e_0)\right) + e_0 \tag{10}$$

where $\frac{\gamma_i}{\sum_{j=0}\gamma_j}$ is the weight scaling factor. T_1 and T_2 are the fusion vectors of the two entities output by the feature fusion module. e_0 is the feature vector corresponding to [CLS]. Sigmoid is the nonlinear activation function. T_{1-Att}, T_{2-Att}, and e_{0-Att} are the fusion vectors of the two entities with [CLS] corresponding to the attentional output. T_1, T_2, and e_0 correspond to the attentional input, respectively.

3.4. Relationship Classification Layer

The relational classification layer receives three outputs from the attention module. A tanh nonlinear activation function is added and a fully connected layer is connected, respectively. The fully-connected layer maps the feature space computed by the first two layers of the network to the feature space of the dataset samples. The purpose is to reduce the impact of feature positions on the results of the relational classification task. The robustness of the network is improved. This process is expressed mathematically as Equations (11)–(13):

$$T'_{1-out} = W_{1-out}[\tanh(T_{1-out})] + b_{1-out} \tag{11}$$

$$T'_{2-out} = W_{2-out}[\tanh(T_{2-out})] + b_{2-out} \tag{12}$$

$$e'_{0-out} = W_{0-out}[\tanh(e_{0-out})] + b_{0-out} \tag{13}$$

where T_{1-out}, T_{2-out}, and e_{0-out} are the feature vectors corresponding to entity I, entity II, and [CLS] tokens output from the attention layer, respectively, W_{1-out}, W_{2-out}, and W_{0-out} are the learnable parameter matrices, tanh is the nonlinear activation function, and b_i ($i = 1-out, 2-out, 0-out$) is the bias term.

In this paper, the feature vector outputs from the fully connected layer are concatenated. The tandem vector connects a feature amplification attention module with the residual structure. The purpose is to extract the association features in entities and sentences to further strengthen the semantic connection between entities and sentences. The features that identify the relationship between two entities are given greater weights. A softmax layer is connected after the attention module, which is used to map the extracted feature vectors to the number of relations. Finally, the probability value corresponding to each category is obtained, by virtue of which the entity relations are classified. This process can be expressed mathematically as Equations (14) and (15):

$$T = \text{Concat}\left(T'_{1-out}, T'_{2-out}, e'_{0-out}\right) \tag{14}$$

$$P = \text{Softmax}\left(\text{Sigmoid}\left(\frac{\gamma_{T-i}}{\sum_{j=0}\gamma_j}BN(T)\right) + T\right) \quad (15)$$

where T_{1-out}, T_{2-out}, e_{0-out} are the output feature vectors of the fully connected layer. T is the tandem vector. Sigmoid is the nonlinear activation function. Softmax is the classifier.

4. Experiments

4.1. Data Sets

To demonstrate the generalization ability and stability of the model in this paper. This paper adds a KBP37 dataset [21] to the SemEval-2010 Task8 dataset [51]. The KBP37 dataset has a higher data size and training difficulty than the SemEval-2010 Task8 dataset. Experimental validation of this paper's model on two public datasets yielded good results in both cases. F1 values of 90.25% and 70.05% were achieved on the two datasets, respectively.

There are a total of 10,717 data items in the SemEval-2010 Task8 dataset. Entity locations and inter-entity relationships have been labeled in the data. The entire dataset was divided into 8000 training data and 2717 test data. There are a total of 9 semantic relationships and "other" (indicating that no relationship exists between entities) in the dataset. KBP37 contains 18 directed relational types and one unrelated type, which will derive 37 categories ($18 \times 2 + 1 = 37$). The entire dataset was split into 15,917 training data items, 3405 test data items, and 1724 extension data items. The longest sentences in KBP37 are nearly twice as long as in the SemEval-2010 Task8 dataset, and the data are greater. In both respects, the KBP37 dataset is more complex and more difficult to train than the SemEval-2010 Task8 dataset. The quantitative information for each relationship in both datasets is shown in Tables 3 and 4.

Table 3. Data distribution of the SemEval-2010 Task8 dataset.

Relationship	Train	Dev	Test
Cause–Effect	1003	–	328
Component–Whole	941	–	312
Content–Container	845	–	292
Entity–Destination	717	–	231
Entity–Origin	716	–	258
Message–Topic	690	–	233
Member–Collection	634	–	261
Instrument–Agency	540	–	192
Product–producer	504	–	156
other	1410	–	454
Total	8000	–	2717

Table 4. Data distribution of the KBP37 dataset.

Relationship	Train	Dev	Test
org:alternate_names	177	24	46
org:city_of_headquarters	511	63	125
org:country_of_headquarters	266	28	65
org:founded	393	53	107
org:founded_by	355	34	80
org:members	703	82	160
org:stateorprovince_of_headquarters	517	65	126
org:subsidiaries	832	103	193
org:top_members/employees	575	68	136
per:alternate_names	511	63	125
per:cities_of_residence	1267	157	305

Table 4. *Cont.*

Relationship	Train	Dev	Test
per:countries_of_residence	1006	119	228
per:country_of_birth	355	50	89
per:employee_of	3472	273	568
per:origin	266	28	65
per:spouse	258	29	57
per:stateorprovinces_of_residence	720	66	125
per:title	641	75	137
no_relation	1545	210	419
Total	15,917	1724	3405

4.2. Experimental Setup

An official model evaluation script is available for the SemEval-2010 Task8 dataset. This script calculates and saves the accuracy values for the direction considered and the direction not considered (no other relationships are included). There is no official model evaluation script for the KBP37 dataset. This paper uses the same formula for calculating F1 as the SemEval-2010 Task8 dataset as an assessment metric for comparison with the baseline model. Our baseline model uses the output of the hidden layer from the last layer of BERT-large. The entity features are extracted from the hidden layer by a mask operation. The entity features are stitched with (CLS) features and fed into the fully connected layer for classification. The main parameters used in the experiments and the experimental environment are shown in Table 5.

Table 5. Experimental parameters and experimental environment.

Parameters and Environment	SemEval-2010 Task	KBP37
GPU	NVIDIA-A100	NVIDIA-A100
Programming language	Python3.9	Python3.9
Deep learning framework	PyTorch1.9	PyTorch1.9
BERT Version	BERT-large	BERT-large
Max sentence length	384	384
Learning rate	2×10^{-5}	7×10^{-6}
Dropout rate	0.1	0.1
Size of mini batch-train	16	23
Size of mini batch-test	32	46

4.3. Experimental Results and Analysis

The effectiveness and sophistication of this paper's relational classification network FA-RCNet is well demonstrated in this section. This paper compares more representative relationship classification models on the SemEval-2010 Task8 dataset. Among the models compared are CNN-based relational classification models. This type of model is less effective when considering directionality, and it is more difficult to notice the relationship between the different orientations of the entities. The models in this paper are also compared with RNN-based approaches, as well as with the performance of several recent relational classification models on the SemEval-2010 Task8 dataset. The KBP37 dataset has relatively few comparable experimental results, and this paper lists a few recently retrievable models for comparison of the experimental data. Moreover, to demonstrate the validity of the work in this paper, modular ablation experiments were carried out on each of the two datasets. Due to the small number of comparable models in the KBP37 dataset, the SemEval-2010 Task8 dataset is used as the main comparison experiment and the KBP37 dataset is used as a secondary comparison experiment in this paper.

4.3.1. Comparative Experimental Results and Analysis

The main comparison models in this paper on the SemEval-2010 Task8 dataset are RNN-based models with BiLSTM-Attention, Entity Attention BiLSTM, BLSTM + BTLSTM + Att. CNN models based on CR-CNN, Multi-Attention CNN, DesRC - CNN, TACNN. TRE, R - BERT, BERTEM + MTB, BERT - GMAN based on pre-trained models. F1 scores are shown in Table 6.

Table 6. A comparison test with other relational classification models on the SemEval-2010 Task8 dataset.

Model	F1
BiLSTM - Attention	84.0%
Entity Attention BiLSTM	85.2%
BLSTM + BTLSTM + Att	87.1%
Bi - LSTM + LET	85.2%
MALNet	86.3%
CR - CNN	84.1%
Multi - Attention CNN	88.0%
DesRC - CNN	86.6%
TACNN	85.3%
TRE	87.1%
R - BERT	89.25%
BERTEM + MTB	89.5%
BERT - GMAN	90.25%
FA-RCNet (our)	**90.33%**

As can be seen from Table 6, the model in this paper achieves good results on the SemEval-2010 Task8 dataset. The F1 score reached 90.33%, which is an improvement of 3.23%, 1.08%, 0.83%, and 0.08% for the models in this paper compared to the TRE, R-BERT, BERTEM + MTB, and BERT - GMAN models, which are also based on pre - trained models, respectively.

The main comparison models in this paper on the KBP37 dataset are the more classical models RNN, CNN, BiLSTM - CNN, structured block - CNN, Att - RCNN, Bi - SDP - Att. There is also recent work proposing relational classification models such as D - BERT, LGCNN, MALNet. F1 scores are shown in Table 7.

Table 7. A comparison test with other relational classification models on the KBP37 dataset.

Model	F1
MALNet	28.8%
CR - CNN	55.1%
Multi - Attention CNN	60.1%
DesRC - CNN	60.9%
TACNN	61.83%
TRE	64.39%
R - BERT	61.4%
BERTEM + MTB	63.2%
BERT - GMAN	69.2%
FA-RCNet (our)	**69.95%**

As can be seen from Table 7, the model in this paper also achieves good results on the KBP37 dataset. The F1 score reached 69.9%. Compared to the performance of the models LGCNN, MALNet, and D-BERT proposed in recent work on the KBP dataset, the F1 values of the models in this paper were improved by 8.55%, 6.75%, and 0.75%, respectively.

The above experimental results show that the model in this paper performs separate attention operations on entity features, which are then stitched with the global information of the whole sentence. It is not only possible to enhance the semantic representation of entity

features, but also to embed entity semantics into contextual information. The semantic information in the input features of the classification network can be better enriched, which in turn improves the accuracy of the relational classification task. Compared with the CNN-based relationship classification model, FA-RCNet can overcome the shortcomings of the CNN model in terms of time span. It can just capture the relationship characteristics between long-distance entities. Compared with the RNN-based relationship classification model, FA-RCNet can overcome the disadvantage that the RNN model cannot query the correlation between the current word and each word in the sentence. FA-RCNet extracts the semantic features of each word in the sentence through the self-attention mechanism, which can obtain higher-quality feature vectors. Using high-quality feature vectors can further improve the accuracy of relation classification models.

This paper also aims to demonstrate that the model is directionally sensitive. The Bi-LSTM+LET, a relational classification model based on the Bi-LSTM, was used as the comparison model. The Bi-LSTM model can identify high-quality directional features, so the Bi-LSTM-based relational classification model is ideal in terms of directional sensitivity. This is the reason why the Bi-LSTM+LET model based on Bi-LSTM is used as the comparison model in this paper. The F1 values for both models, when considering directionality, are shown in Table 8.

Table 8. F1 values for each category in the SemEval-2010 Task8 dataset for both models, taking directionality into account.

Model	F1-Our	F1-BiLSTM+LET	Δ
CE1	**93.38%**	93.28%	0.1%
CE2	**92.35%**	89.72%	2.63%
CW1	**91.30%**	82.21%	9.09%
CW2	**84.59%**	77.44%	7.15%
CC1	**91.19%**	86.16%	5.03%
CC2	**92.50%**	75.61%	16.89%
ED1	**94.12%**	89.00%	5.12%
ED2	0.00%	0.00%	0.00%
EO1	**90.19%**	84.93%	5.26%
EO2	**91.49%**	85.39%	6.1%
IA1	**73.91%**	62.22%	**11.69%**
IA2	**81.51%**	77.54%	3.97%
MC1	**84.75%**	73.33%	11.42%
MC2	**89.31%**	87.89%	1.42%
MT1	**92.86%**	87.01%	5.85%
MT2	**85.98%**	83.33%	2.65%
PP1	**86.51%**	81.48%	5.03%
PP2	**88.70%**	80.78%	7.92%
Other	**65.73%**	51.75%	13.98%

In the table, C-E1 is cause–effect (e_1, e_2) and C-E2 is cause–effect (e_2, e_1) in the opposite direction of C-E1. All other relations are abbreviated according to this rule.

Since entity–destination (e_2, e_1) has only one data item in the training set, a better training result was not obtained. The F1 values for both models are zero, a problem that is a data set defect and not sufficient to demonstrate the weak generalization of the model. Conversely, as can be seen from the rest of the table, the F1 values of the model in this paper are greater than or equal to those of the Bi-LSTM+LET model in different directions for each category. In particular, there was an 11.69% increase in the instrument–agency (e_1, e_2) compared to the Bi-LSTM+LET model. The recognition accuracy in the direction of the instrument–agency (e_1, e_2) is the bottleneck of other relational classification models, thus demonstrating that the model in this paper is sensitive to direction. As the directionality of the relationship is not defined in the KBP37 dataset, it is not possible to provide a comparative analysis of directional sensitivity on this dataset.

To further investigate the directional sensitivity of the models in this paper, the experimental results of the two models on the SemEval-2010 Task8 dataset are visualized. The visualization is shown in Figure 5. It can be visualized from the figure that the model in this paper is higher than the Bi-LSTM+LET model in any of the directions of each relationship. As mentioned in the previous section, this paper overcomes the model shortcomings in the instrument–agency (e_1, e_2) direction and improves the accuracy in this direction. At the same time, it can be seen from the Δ values that this paper has not only improved significantly in this direction, but also other non-short areas. For example, with content–container (e_2, e_1), member–collection (e_1, e_2), component–whole (e_1, e_2), and product-producer (e_2, e_1), the F1 values improved by 16.89%, 11.42%, 9.09%, and 7.92%.

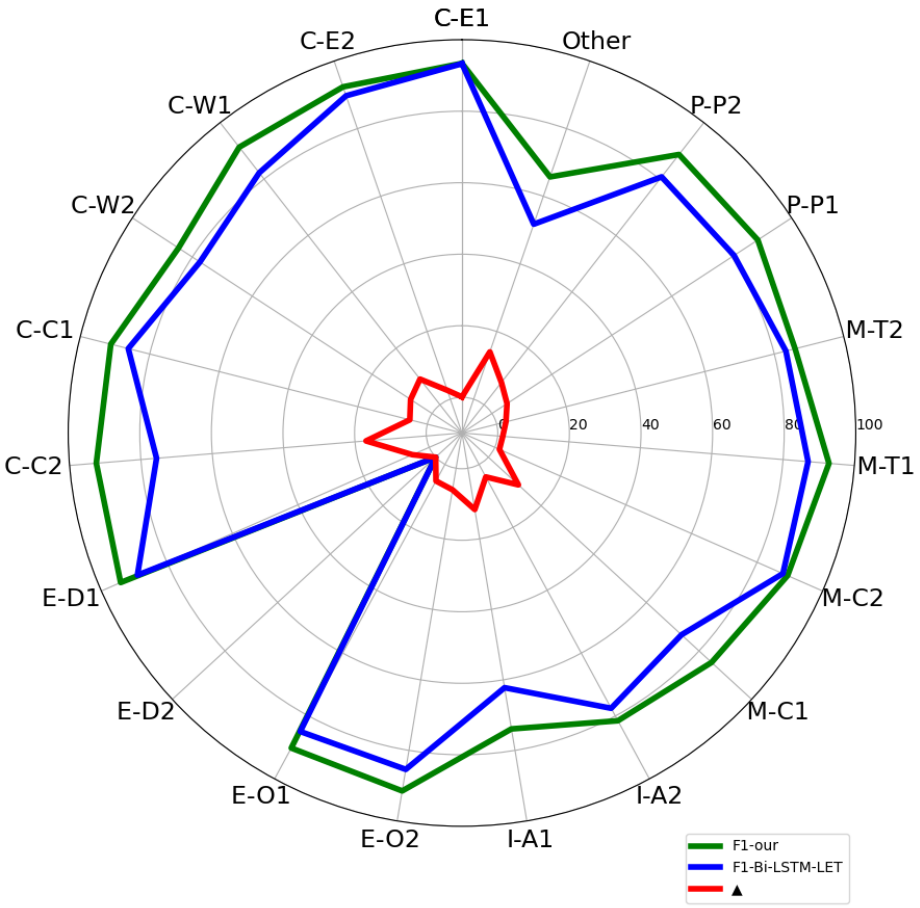

Figure 5. Visualization of the experimental effects of the two models on the SemEval-2010 Task8 dataset.

4.3.2. Results and Analysis of Ablation Experiments

The effectiveness of the module proposed in this paper is demonstrated by ablation experiments. Experiments were conducted by adding a feature fusion module to the baseline model, adding an attention module, and adding both an attention module and a feature fusion module. Testing the effectiveness of the two modules proposed in this paper for the relational classification task, the results of the ablation experiments of this paper's model on the two datasets are shown in Tables 9 and 10.

The data from the ablation experiment reveals that both the feature fusion module and the attention module proposed in this paper can improve the experimental results to a different level. The feature fusion module combines shallow, fine-grained information with deep, coarse-grained information that is rich in semantic information. It allows the model to better learn the relationship between two entities from the sentence and the two entities. The attention module amplifies the effective features and suppresses the

effect of invalid features on the accuracy of the model, thus achieving the accuracy of the classification results. The ablation experiments demonstrate the effectiveness of the module proposed in this paper on the SemEval-2010 Task8 and KBP37 datasets.

Table 9. Ablation experiments for each module on the SemEval-2010 Task8 dataset.

Model	P	R	F1
Base	82.43%	86.04%	84.19%
Base + Attention module	86.75%	90.81%	88.73%
Base + Feature fusion module	87.35%	91.25%	89.26%
Base + Feature fusion module + Attention module	**89.29**%	**91.38**%	**90.33**%

Table 10. Ablation experiments for each module on the KBP37 dataset.

Model	P	R	F1
Base	66.86%	72.37%	69.51%
Base + Attention module	67.34%	72.43%	69.79%
Base + Feature fusion module	67.6%	72.20%	69.79%
Base + Feature fusion module + Attention module	**67.80**%	**72.23**%	**69.95**%

4.4. Confusion Matrix Analysis

This paper focuses on comparing the prediction results of the FA-RCNet model with the real results using the confusion matrix method on the SemEval-2010 Task8 dataset. Confusion matrices were calculated for 10 classifications (without differentiation of orientation) and 19 classifications (with differentiation of entity orientation), respectively. The diagonal areas of the confusion matrix show the results of correct model predictions, while the values in the other areas show the results of the error distribution.

Figure 6 shows the confusion matrix for the results of the 10 classifications (without differentiation of direction). The graph shows that although the classification accuracy of the 'I-A' relationship has improved, it is still the smallest of all the categories. The most frequent error in all classifications is the classification of non-"other" as "other". The reason for this is that the model does not identify the type of relationship between the two entities or even that a relationship exists between the two entities. This paper argues that the reason for this phenomenon is that the semantics between each category is discrete and that to improve this phenomenon, we need to introduce external features or semantic logical reasoning.

Figure 7 shows the confusion matrix for the results of the 19 classifications (differentiated directions). In this matrix, it is possible to show the recognition accuracy in different directions for the same relationship. The figure shows that the datum in the E-D2 (entity–destination<e_2, e_1>) direction is 0, as explained in the previous section. Because there are 8000 data items in the training set of the dataset, only one datum has a relationship of E-D2, and there are 2717 data items in the test set, and only one datum has a relationship with E-D2. Deficiencies in the data set resulted in an accuracy of 0% in this direction. After excluding the E-D2 direction, I-A1 (instrument–agency<e_1, e_2>) had the lowest accuracy among the other directions, which explains the low accuracy of the I-A category in the 10 classification results regardless of direction. On the one hand, this is because the focus in the features of this text is more on the role of the two entity features, weakening the links between the different words in the whole sentence. On the other hand, a part of the global features corresponding to (CLS) in the shallow information is also lost. This leads to suboptimal performance of the model in the more dependent categories of relationships between words. In future work, we will focus on the role of shallow global features in classification tasks.

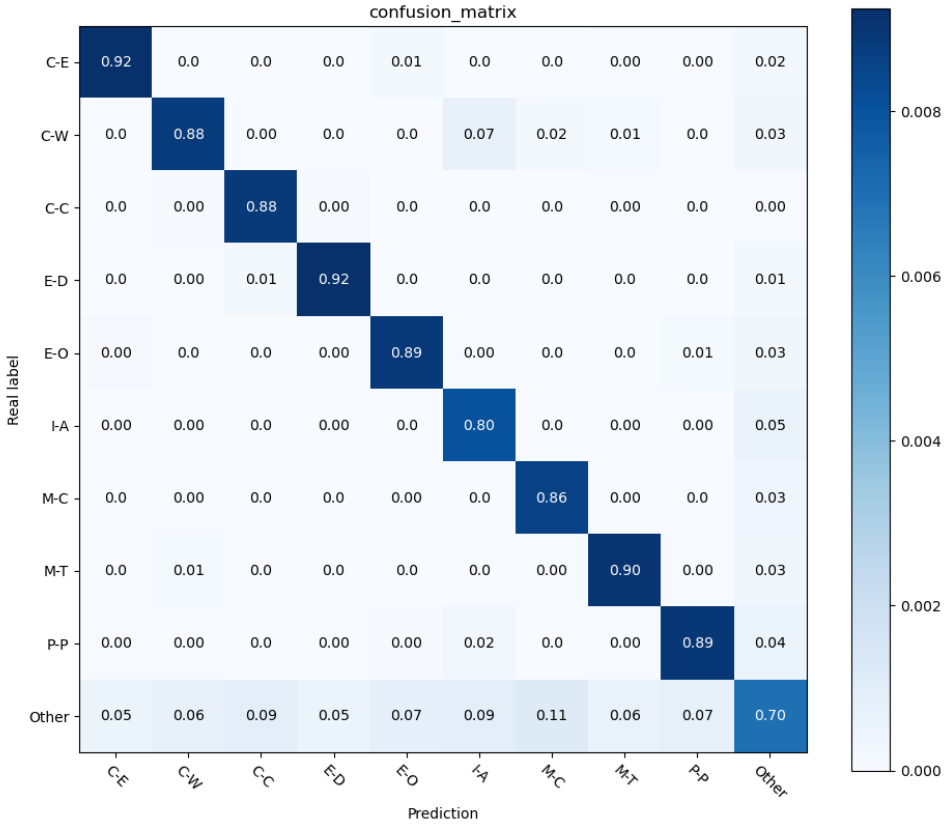

Figure 6. 10 Confusion matrix of classified (non-differentiated direction) results.

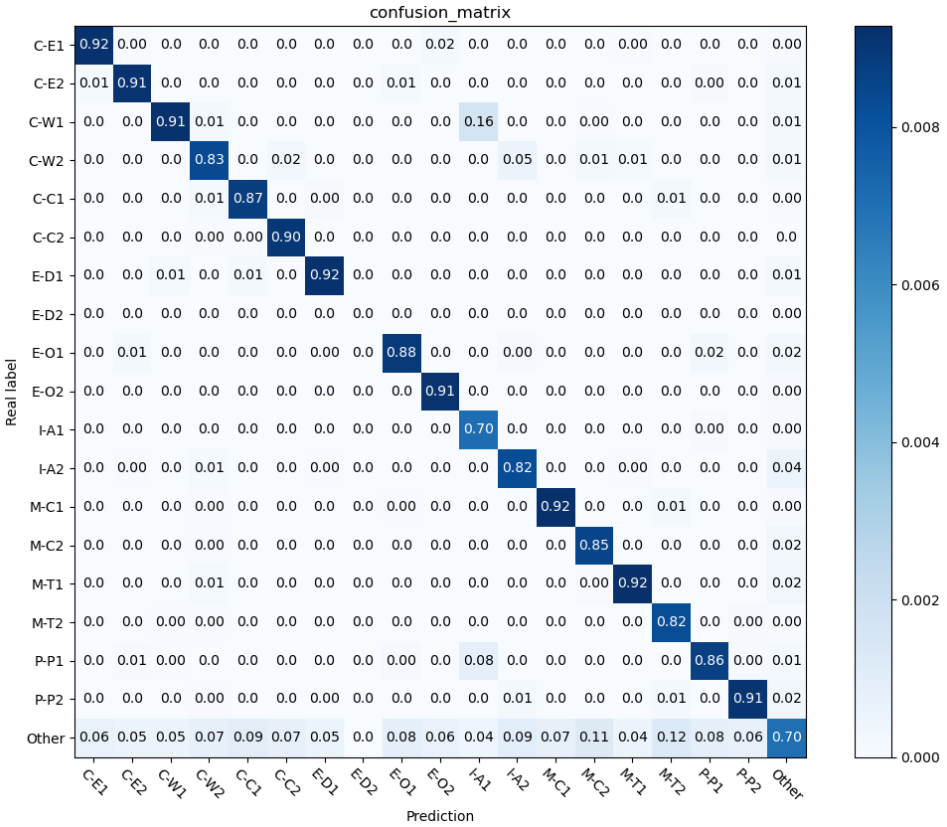

Figure 7. A total of 19 confusion matrices of classified (differentiated direction) results.

5. Conclusions and Future Work

In this paper, a new FA-RCNet entity relationship classification model is proposed. The study found that the shallow information of the BERT model also has a positive effect on the relationship classification task. Therefore, the FA-RCNet model fuses the feature outputs of different levels of the BERT pre-training model. A fusion feature containing shallow semantic information and deep semantic information is formed. Improving the accuracy of subsequent relation classification tasks by fusing rich features in features. At the same time, a feature amplification attention module is designed to amplify the positive features and suppress the negative features in the semantic features. Through this operation, the effect of highlighting positive features is achieved. The formed semantic features can further improve the accuracy of relation classification tasks. At the same time, the experiments of FA-RCNet on the SemEval-2010 Task 8 and KBP37 data sets show that the performance of the FA-RCNet model is better than those of the existing methods, and the F1 values reach 90.33% and 69.95%, respectively. Ablation experiments on two datasets show that different modules in the FA-RCNet model have positive effects on relation classification tasks.

In addition, we believe that the FA-RCNet model is still insufficient in dealing with multi-entity relationship problems. The model can only recognize two entities in a sentence, which is determined by the mask matrix that extracts entity features. When there are multiple entities in a sentence, there may be a problem that the relationship between a certain pair of entities cannot be identified. In addition, when there are multiple relationships between two entities, the model in this paper can usually only identify the relationship with a higher probability, and cannot accurately identify all of them. This is the problem with our model.

In future work, we will continue to conduct research on how to dynamically generate the mask matrix according to the data format in the dataset to realize the relationship extraction between multiple entities. At the same time, we will look into how to introduce external information in the feature fusion stage to accurately identify the various relationships between entities.

Author Contributions: Software, resources, methodology G.L.; methodology, validation, formal analysis, data curation, writing—original draft preparation J.T.; software, visualization M.Z.; project administration, funding acquisition M.L.; writing—review and editing D.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding from the National Key R&D Plan of China (2022YFF0608000).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Pawar, S.; Palshikar, G.K.; Bhattacharyya, P. Relation Extraction: A Survey. *arXiv* **2017**, arXiv:1712.05191.
2. Li, L. A distributed meta-learning system for Chinese entity relation extraction. *Neurocomputing* **2015**, *149*, 1135–1142. [CrossRef]
3. Aliguliyev, R.M. A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Syst. Appl.* **2009**, *36*, 7764–7772. [CrossRef]
4. Girju, R. Automatic Detection of Causal Relations for Question Answering. 2003. 8p. Available online: https://aclanthology.org/W03-1210.pdf (accessed on 11 July 2003).
5. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv* **2016**, arXiv:1409.0473.

6. Wang, Q.; Mao, Z.; Wang, B.; Guo, L. Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Trans. Knowl. Data Eng.* 2017, 29, 2724–2743. [CrossRef]
7. Kate, R.J.; Mooney, R.J. Using string-kernels for learning semantic parsers. In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Sydney, Australia, 17–21 July 2006; Association for Computational Linguistics: Stroudsburg, PA, USA, 2006; pp. 913–920. [CrossRef]
8. Paramesha, K.; Ravishankar, K.C. Exploiting dependency relations for sentence level sentiment classification using SVM. In Proceedings of the 2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), Coimbatore, India, 5–7 March 2015; pp. 1–4. [CrossRef]
9. Raut, P.P.; Patil, N.N. Classification of controversial news article based on disputant relation by SVM classifier. In Proceedings of the 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), Noida, India, 2–4 September 2015; pp. 1–5. [CrossRef]
10. Wang, H.; Qin, K.; Zakari, R.Y.; Lu, G.; Yin, J. Deep neural network-based relation extraction: An overview. *Neural Comput. Appl.* **2022**, *34*, 4781–4801. [CrossRef]
11. Xu, S.; Sun, S.; Zhang, Z.; Xu, F.; Liu, J. BERT gated multi-window attention network for relation extraction. *Neurocomputing* **2022**, *492*, 516–529. [CrossRef]
12. Wu, S.; He, Y. Enriching Pre-trained Language Model with Entity Information for Relation Classification. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; ACM: New York, NY, USA, 2019; pp. 2361–2364. [CrossRef]
13. Liang, D.; Xu, W.; Zhao, Y. Combining Word-Level and Character-Level Representations for Relation Classification of Informal Text. In Proceedings of the 2nd Workshop on Representation Learning for NLP, Vancouver, BC, Canada, 3 August 2017.
14. Li, C.; Tian, Y. Downstream Model Design of Pre-trained Language Model for Relation Extraction Task. *arXiv* **2020**, arXiv:2004.03786.
15. Ma, Y.; Sun, Z.; Zhang, D.; Feng, Y. Traditional Chinese Medicine Word Representation Model Augmented with Semantic and Grammatical Information. *Information* **2022**, *13*, 296. [CrossRef]
16. Nguyen, T.H.; Grishman, R. Relation Extraction: Perspective from Convolutional Neural Networks. In Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, Denver, CO, USA, 5 June 2015.
17. dos Santos, C.N.; Xiang, B.; Zhou, B. Classifying Relations by Ranking with Convolutional Neural Networks. *arXiv* **2015**, arXiv:1504.06580.
18. Wang, L.; Cao, Z.; de Melo, G.; Liu, Z. Relation Classification via Multi-Level Attention CNNs. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; Association for Computational Linguistics: Stroudsburg, PA, USA, 2016; pp. 1298–1307. [CrossRef]
19. Quan, Z.; Zeng, W.; Li, X.; Liu, Y.; Yu, Y.; Yang, W. Recurrent Neural Networks With External Addressable Long-Term and Working Memory for Learning Long-Term Dependences. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 813–826. [CrossRef] [PubMed]
20. Li, J.; Luong, M.-T.; Jurafsky, D.; Hovy, E. When Are Tree Structures Necessary for Deep Learning of Representations? *arXiv* **2015**, arXiv:1503.00185.
21. Zhang, D.; Wang, D. Relation Classification via Recurrent Neural Network. *arXiv* **2015**, arXiv:1508.01006.
22. Lee, J.; Seo, S.; Choi, Y.S. Semantic Relation Classification via Bidirectional LSTM Networks with Entity-aware Attention using Latent Entity Typing. *Symmetry* **2019**, *11*, 785. [CrossRef]
23. Chaudhari, S.; Mithal, V.; Polatkan, G.; Ramanath, R. An Attentive Survey of Attention Models. *ACM Trans. Intell. Syst. Technol. (TIST)* **2021**, *12*, 1–32. [CrossRef]
24. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2017**, arXiv:1706.03762
25. Im, J.; Cho, S. Distance-based Self-Attention Network for Natural Language Inference. *arXiv* **2017**, arXiv:1712.02047.
26. Shen, T.; Zhou, T.; Long, G.; Jiang, J.; Pan, S.; Zhang, C. DiSAN: Directional Self-Attention Network for RNN/CNN-Free Language Understanding. *arXiv* **2017**, arXiv:1709.04696.
27. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent Models of Visual Attention. *arXiv* **2014**, arXiv:1406.6247.
28. Jaderberg, M.; Simonyan, K.; Zisserman, A. Spatial Transformer Networks. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), Montreal, QC, Canada, 7–12 December 2015.
29. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *arXiv* **2019**, arXiv:1709.01507.
30. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual Attention Network for Image Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017, Honolulu, HI, USA, 21–26 July 2017; pp. 6450–6458. [CrossRef]
31. Huang, Z.; Wang, X.; Huang, L.; Huang, C.; Wei, Y.; Liu, W. CCNet: Criss-Cross Attention for Semantic Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision 2019, Seoul, Korea, 27 October–2 November 2019.
32. Gao, Z.; Xie, J.; Wang, Q.; Li, P. Global Second-Order Pooling Convolutional Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; p. 10.
33. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. *arXiv* **2020**, arXiv:1910.03151.

34. Zhang, H.; Dana, K.; Shi, J.; Zhang, Z.; Wang, X.; Tyagi, A.; Agrawal, A. Context Encoding for Semantic Segmentation. *arXiv* **2018**, arXiv:1803.08904.
35. Liu, Y.; Shao, Z.; Teng, Y.; Hoffmann, N. NAM: Normalization-based Attention Module. *arXiv* **2021**, arXiv:2111.12419.
36. Zhou, P.; Shi, W.; Tian, J.; Qi, Z.; Li, B.; Hao, H.; Xu, B. Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification. In Proceedings of the 54th Annual Meeting of the Association For Computational Linguistics, Berlin, Germany, 7–12 August 2016; Volume 2 Short Papers; Association for Computational Linguistics: Stroudsburg, PA, USA, 2016; pp. 207–212. [CrossRef]
37. Liu, G.; Guo, J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* **2019**, *337*, 325–338. [CrossRef]
38. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**, arXiv:1810.04805.
39. Geng, Z.; Li, J.; Han, Y.; Zhang, Y. Novel target attention convolutional neural network for relation classification. *Inf. Sci.* **2022**, *597*, 24–37. [CrossRef]
40. Li, Y.; Ma, Z.; Gao, L.; Wu, Y.; Xie, F.; Ren, X. Enhance prototypical networks with hybrid attention and confusing loss function for few-shot relation classification. *Neurocomputing* **2022**, *493*, 362–372. [CrossRef]
41. Xiao, Y.; Jin, Y.; Cheng, R.; Hao, K. Hybrid attention-based transformer block model for distant supervision relation extraction. *Neurocomputing* **2022**, *470*, 29–39. [CrossRef]
42. Lee, Y.; Son, J.; Song, M. BertSRC: Transformer-based semantic relation classification. *BMC Med. Inform. Decis. Mak.* **2022**, *22*, 234. [CrossRef]
43. Keya, A.J.; Wadud, M.A.; Mridha, M.F.; Alatiyyah, M.; Hamid, M.A. AugFake-BERT: Handling Imbalance through Augmentation of Fake News Using BERT to Enhance the Performance of Fake News Classification. *Appl. Sci.* **2022**, *12* 8398. [CrossRef]
44. Shao, C.; Li, M.; Li, G.; Zhou, M.; Han, D. CRSAtt: By Capturing Relational Span and Using Attention for Relation Classification. *Appl. Sci.* **2022**, *12*, 11068. [CrossRef]
45. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. *arXiv* **2018**, arXiv:1807.06521.
46. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. *arXiv* **2021**, arXiv:2103.02907.
47. Liu, H.-I.; Chen, M.-W.; Kao, W.-C.; Yeh, Y.-W.; Yang, C.-X. GSAP: A Hybrid GRU and Self-Attention Based Model for Dual Medical NLP Tasks. In Proceedings of the 2022 14th International Conference on Knowledge and Smart Technology (KST), Chon buri, Thailand, 26–29 January 2022; pp. 80–85. [CrossRef]
48. Jin, Y.; Wu, D.; Guo, W. Attention-Based LSTM with Filter Mechanism for Entity Relation Classification. *Symmetry* **2020**, *12*, 1729. [CrossRef]
49. Laurent, C.; Pereyra, G.; Brakel, P.; Zhang, Y.; Bengio, Y. Batch Normalized Recurrent Neural Networks. *arXiv* **2015**, arXiv:1510.01378.
50. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* **2015**, arXiv:1502.03167.
51. Hendrickx, I.; Kim, S.N.; Kozareva, Z.; Nakov, P.; Séaghdha, D.Ó.; Padó, S.; Pennacchiotti, M.; Romano, L.; Szpakowicz, S. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions, DEW '09, Boulder, CO, USA, 4 June 2009; Association for Computational Linguistics: Stroudsburg, PA, USA, 2009; p. 94. [CrossRef]