Neural Processing Letters (2026) 58:18
https://doi.org/10.1007/s11063-025-11831-0

Relation Extraction from the Perspective of the Frequency Domain: Frequency-Domain Aware Gated Graph Attention Network

Zhan'ao Yao[1] · Wenli Geng[1] · Hongxin Yang[1] · Zhongtian Sun[2] · Tingwei Chen[1]

Received: 25 March 2025 / Accepted: 27 December 2025 / Published online: 11 January 2026

Abstract
In relation extraction task, graph attention network, as the dominant model, often faces the challenge of attention bias caused by complex semantic environment. Existing approaches ignore decoupling and build fine-grained models to filter and directly interact the multiple levels of information overlapping in word vectors (word, phrase, clause, and sentence level), but using methods that focus only on context (such as additional knowledge or structure). Ignoring overlapping multilevel information leads to limited performance improvement of the model for attention bias, but also increases the processing cost. To overcome this core limitation, we propose a model to decouple and process multiple levels of semantic information from the spectrum domain: Frequency-domain aware Gated Graph Attention Network (FD-GGAN-RE). The network first uses spectral decomposition to decouple contextual word vectors into spectral domain vectors containing different levels of semantic information. Then, use Frequency Feature Selective Gate layer to realize adaptive semantic filtering, reducing the influence of irrelevant semantics on the subsequent graph attention calculation. Final the Frequency-domain graph attention layer realizes the direct interaction of multiple levels of semantic information in the spectrum domain, avoiding the attention bias caused by the context graph attention mechanism interacting with word vectors containing multiple levels of semantic overlap. SemEval and KBP37 scored 90.33 and 69.06 respectively for F1,

Zhan'ao Yao, Wenli Geng: These authors contributed equally to this work.

✉ Tingwei Chen
twchen@lnu.edu.cn

Zhan'ao Yao
4032232407@smail.lnu.edu.cn

Wenli Geng
4032332453@smail.lnu.edu.cn

Hongxin Yang
yanghongxin@lnu.edu.cn

Zhongtian Sun
z.sun-256@kent.ac.uk

1 Faculty of Information, Liaoning University, 66 Chongshan Middle Road, Huanggu Qu 110031, Shenyang, China

2 School of computing, University of Kent, University Road, Canterbury CT2 7NZ, Kent, United Kingdom

which was 27% faster than GATs while F1 scored 0.15 and 0.84 higher, respectively. Frequency graph attention visualization further demonstrates the model's capability to capture complex key semantics, while presenting a frequency-based approach that holds potential for application in other natural language processing tasks.

Keywords Relation extraction · Attention mechanism · Frequency-domain techniques

1 Introduction

In Natural Language Processing (NLP), Relation Extraction (RE) focuses on identifying relations between the given entity pair from unstructured text, which is critical for applications such as knowledge graph construction [1], information retrieval [2], and question-answering systems [3]. In neural relation extraction (RE) tasks, graph neural networks (GNN) [4], Graph Convolutional Networks [5] and graph attention mechanisms(GAT) [6] have become effective modeling methods. Among them, GATs, with its unique attention allocation strategy, makes full use of contextual information in sentences through information transfer mechanism, and shows excellent performance and advantages in dealing with entity relationship modeling. However, GATs assigns scores of higher attention score to language elements that may not be task-relevant in complex semantic environments, thus reducing the accuracy of RE. For example, in the sentence "The narrative comprises the interplay of two themes that...", the relation between "narrative" and "interplay" is "Other". But because the model focuses on "comprises", "themes" or other words, the model may mispredict its relation as "component-whole."

Existing methods are mainly context-based improved modeling, which can be summarized into two technical routes: methods based on external knowledge enhancement, such as syntactic information [7], and methods based on graph neural network structure optimization, such as multi-head mechanism [8]. Because it is difficult to decouple the multi-level semantics (including sentence, clause, phrase, and word information) in the implicit space of word vectors, the context method does not model the filtering of an interaction among multi-level semantic features. Instead, it directly performs coarse-grained interaction on word vectors that overlap multi-level semantic information using the context graph attention. These directly lead to the limited optimization effect of the context-improved model with additional consumption, and the graph attention weight is still skewed.

Inspired by [9, 10], we use frequency domain variation to achieve word vector decoupling and solve this problem in the frequency domain. After the spectral decoupling of the word vector, the frequency domain vector will be recognized to contain the information of words, phrases, clauses and sentences according to the frequency level, which realizes the multi-layer information decoupling which is difficult to achieve in context method [11]. Then, a frequency feature selection gate module reduces the deviation of the graph attention by achieving fine-grained feature dimension spectrum filtering on the frequency independent component of the task. As a supplement, the frequency domain graph attention module solves the attention shift caused by semantic overlap in coarse-grained context attention by directly interacting the multi-level context information in the frequency domain. To sum up, in this paper, we formally propose **F**requency-**D**omain aware **G**ated **G**raph **A**ttention **N**etwork for **R**elation **E**xtraction (**FD-GGAN-RE**). This is an innovative frequency-domain enhanced graph attention framework, which realizes the difficult problem of multilevel information

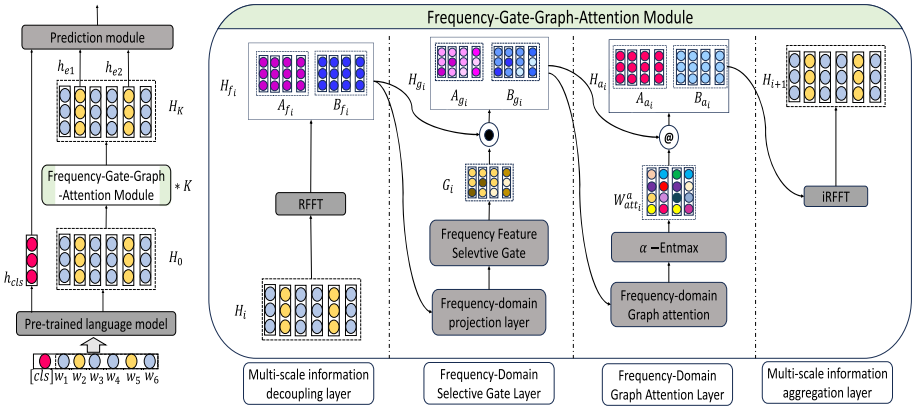

Fig. 1 The left side shows the overall architecture of the FD-GGAN-RE model framework, which consists of three main parts: pre-training Language model, frequency domain gated attention module and prediction module. The frequency domain gating attention module connects Multi-scale information decoupling layer, Frequency-Domain Selective Gate Layer, Frequency-Domain Graph, Attention Layer, Multi-scale information aggregation layer connected in sequence. On the right is the specific architecture of these four layers

overlap in context graph attention with innovative initiatives in spectral decoupling and spectral domain processing, and significantly reduces the attention bias in the GATs.

The main contributions of this paper are as follows:

(1) A learnable Frequency Feature Selective Gate layer is proposed to realize fine-grained semantic component selection by spectral filtering, effectively suppress noise interference and strengthen mission-critical frequency components to achieve constraints on graph attention weight shift in RE.
(2) An interpretable Frequency-Domain Graph Attention layer is established for the first time, which solves the problem that traditional graph attention is difficult to distinguish multi-layer semantic features in RE. This breakthrough also provides a new way to study graph attention from spectral perspective in other natural language processing.
(3) Our FD-GGAN-RE achieves F1 score 90.33 and 69.06 on SemEval and KBP37, outperforming existing context-based graph RE models by 0.15 and 0.84 F1 score with 27% faster computation. Frequency-specific attention visualization demonstrates interpretable pattern alignment with linguistic relational markers.

2 Method

In this section, the FD-GGAN-RE model is introduced. As shown in Fig. 1, the proposed FD-GGAN-RE model consists of the Pre-trained language module, the Frequency-Gate-Graph-Attention module, and the Prediction module.

2.1 Pre-Trained Language Module

Given an input sentence $S = \{w_1, w_2, \ldots, w_n\}$ and a pair of entities (w_{e_1}, w_{e_2}), where each entity may consist of multiple words, for example, entity w_{e_1} could be a sub-sequence of words

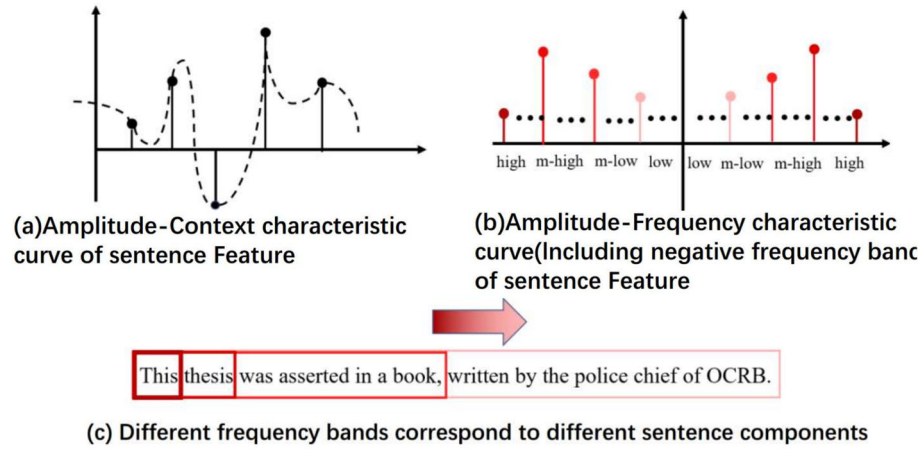

Fig. 2 The figure shows the situation of different sentence components corresponding to different frequency segments from the perspective of frequency domain. (**a**) represents the context-amplitude characteristic curve of the sentence under a certain feature after embedding layer. (**b**) is the amplitude-frequency characteristic curve of negative frequency band sentence features, which is divided into four parts according to frequency band level: low, medium and low, medium and high. (**c**) represents the relationship between elements in the sentence and different frequency bands. As the length of the box in (**c**) increases, it contains more and more words, and the slower the overall change, the lower the corresponding frequency in the band

$\{w_i, w_{i+1}, \ldots, w_j\}$, and entity w_{e_2} could be a sub-sequence of words $\{w_p, w_{p+1}, \ldots, w_q\}$, where $1 \leq i \leq j \leq n$ and $1 \leq p \leq q \leq n$.

First, pre-trained language models, such as BERT[12], are utilized as contextual encoders to generate the hidden contextual representation $H_0 = \{h_{0-1}, h_{0-2}, \ldots, h_{0-n}\}$, $H_0 \in \mathbb{R}^{n \times d}$ of the input sentence S, where d denotes the dimension of the word representation, and n represents the sentence length.

Following the extraction of the contextual representation, the content of the [CLS] token, h_{cls}, is employed to capture the global information of the entire sentence.

2.2 Frequency-Gate-Graph-Attention module

In this chapter, the i-th Frequency-Gate-Graph-Attention module is taken as an example to elaborate four sub-layers of the Frequency-Gate-Graph-Attention module: Multi-scale information decoupling layer, Frequency Feature Selection Gate layer, Frequency-Domain Graph Attention layer, and Multi-scale information aggregation layer. Specifically: the module consists of k modules, where i $(1 \leq i \leq k)$ represents the i-th module.Its input is H_{i-1}.

2.2.1 Multi-Scale Information Decoupling Layer

After obtaining the sentence representation H_{i-1}, we use the spectral method to solve the problem that the traditional context graph model is difficult to decouple the entangled word vectors of multilevel language components. Based on the prior research [10], spectral decomposition can be achieved through Fourier transform in the sentence dimension, enabling hierarchical deconvolution of linguistic information across four levels of granularity (words, phrases, clauses, and sentences). Specifically, as shown in Fig. 2, amplitude-frequency vectors systematically encode these different language levels through frequency distributions -

lower frequencies correspond to global sentence-level semantics, while progressively higher frequencies capture fine-grained clauses, phrases, and word-level features. Formally, we implement this via Real Fast Fourier Transform (RFFT) following [13]. Specifically, H_{i-1} is first transposed and input it to the decoupling layer as Equation 1.

$$H_{f_i} = \text{RFFT}(H_{i-1}^T) \tag{1}$$

where $H_{i-1}^T \in \mathbb{R}^{d \times n}$ is the transposed input contextual representation matrix and the dimensions of H_{f_i} are $H_{f_i} \in \mathbb{R}^{d \times (\frac{n}{2}+1)}$ which represents the frequency-domain representation without the negative frequency band. The frequency-domain representation H_{f_i} contains the real part A_{f_i} and the imaginary part B_{f_i} as Equation 2:

$$H_{f_i} = A_{f_i} + B_{f_i} i \tag{2}$$

Due to Fourier decomposition, the multilevel information coupled in the sentence is divided into different frequency bands, which is convenient for subsequent models to directly operate on the hierarchical information.

2.2.2 Frequency Feature Selective Gate layer

After obtaining the decoupled frequency domain tensor H_{f_i}, filtering is required. Unselective retention of all hierarchical features may compromise relationship prediction effectiveness because some frequency components introduce interference noise. Therefore, it is necessary to integrate band selection module to suppress interference when screening core relation features. In the context graph attention method, attention bias occurs because the decoupling difficulty processing does not distinguish between these information.

We propose the requency Feature Selective Gate layer to filter the noise information adaptively by taking the frequency feature as the basic unit to suppress the interference. Because the traditional fixed threshold method (such as low-pass, high-pass, bandpass filtering) may lose key information for uniform filtering of specific frequency bands, for example, low-pass filtering may inhibit high-frequency information, resulting in word-level information loss. Meanwhile, the previous adaptive method [13] takes a single frequency vector as the basic unit of spectrum filtering, but because the vector contains multiple feature dimensions of heterogeneous distribution, the use of a unified filter transform operator will damage feature retention and reduce the accuracy of pattern recognition.

Specifically, it is believed that the frequency-domain variation information of a single dimension d can reflect how information is retained across the $\frac{n}{2}+1$ dimensions. Therefore, a Frequency-domain projection layer is used to consider whether each $\frac{n}{2}+1$ dimensional information component under each d dimension should be removed, as follows: first, the filtering component is computed through the Frequency-domain projection layer as Equation 3:

$$H_{p_i} = (W_a)A_{f_i} + (W_b)B_{f_i} + b \tag{3}$$

where A_{f_i} and B_{f_i} are the real and imaginary parts of the frequency-domain representation H_{f_i}, respectively. W_a and W_b are the weight matrices of the projection layer, and b is the bias vector. Through this approach, we integrate and project each feature dimension of the real and imaginary parts of the frequency vector into H_{p_i}. By obtaining the distribution features of each heterogeneous feature dimension, this mechanism can build an adaptive gating module with adaptive resolution of feature distribution. Next, the filtering gate G_i is calculated using the sigmoid function as Eq. 4:

$$G_i = \text{sigmoid}(H_{p_i}) \tag{4}$$

In this way, the distribution of real and virtual information of each heterogeneous feature component can be considered at the same time, so as to realize the learning of the whole information. Finally, the obtained H_{gi} is multiplied with both A_{f_i} and B_{f_i}, resulting in the selected frequency information H_{g_i} as Eq. 5 , Eq. 6 and Eq. 7:

$$H_{g_i} = A_{g_i} + B_{g_i} i \tag{5}$$

$$A_{g_i} = G_i \circ A_{f_i} \tag{6}$$

$$B_{g_i} = G_i \circ B_{f_i} \tag{7}$$

where $\circ$ denotes element-wise multiplication.

Through these steps, the adaptive filtering of the difficult multi-level information in the context domain is realized, which provides a better input for the subsequent graph attention mechanism.

2.2.3 Frequency-Domain Graph Attention layer

After the filtering operation is complete, we need to interact with the retained task-related information. In the traditional graph attention performs attention operations on word vectors to interact relevant information. However, due to the overlapping of multiple layers of information in the context vector hiding space, the attention of context graph can not achieve accurate weight allocation of multiple information interactions, resulting in errors. By achieving fine-grained global information interaction under the condition of avoiding overlapping of multiple information by paying attention to the spectral domain vector directly, so as to ensure the reduction of the graph attention weight offset. In addition, since the spectral domain vector represents the change of the whole sentence information, the spectral domain attention can also realize the global information interaction that the context graph attention mechanism is difficult to achieve. On this basis, this paper proposes a method to exchange information from frequency domain instead of context domain: frequency-domain graph attention mechanism.

Specifically, in the frequency domain environment, the frequency domain vector composed of real and imaginary parts exhibits unique physical characteristics. In view of this, from the new perspective of frequency domain physics, we comprehensively consider three common similarity calculation methods to determine the graph attention weight. They are cosine similarity method [14], Gaussian kernel method [15], and dot product attention method [8]. Next, we will introduce these three methods.

(1) **Cosine Similarity Method**

Cosine similarity method quantifies the similarity between two vectors by calculating the angle between them. In this method, the L2 norms of the input matrices $\|H_{g_i}^T\|$ is first calculated using the following formulas Eq. 8:

$$\|H_{g_i}^T\| = \sqrt{(A_{g_i}^T)^2 + (B_{g_i}^T)^2} \tag{8}$$

To obtain the vector in the standard direction, we performed L2 normalization on $H_{g_i}^T$ as Eq. 9:

$$H_{norm_i} = \frac{H_{g_i}^T}{\|H_{g_i}^T\|} \tag{9}$$

Finally, the cosine similarity is used to obtain the attention score matrix W_{att_i} as Eq. 10:

$$W_{att_i} = \Re(H_{norm_i} \cdot \text{conj}(H_{norm_i}^T)) \tag{10}$$

where conj$(*)$ denotes the conjugate of $*$, $\Re$ represents the real part of a complex number.

(2) **Gaussian Kernel Method**

Gaussian kernel method considers the actual distance between vectors in a multidimensional space rather than the direction. In this method, we first compute the pairwise differences between the complex vectorsdis_{g_i} as Eq. 11:

$$(D_{g_i})_{mn} = \sqrt{\|(A_{g_i})_m^T - (A_{g_i})_n^T\|^2 + \|(B_{g_i})_m^T - (B_{g_i})_n^T\|^2} \tag{11}$$

where $D_{g_i} \in \mathbb{R}^{(\frac{n}{2}+1)\times(\frac{n}{2}+1)}$.

Next, we construct the Gaussian kernel function as Eq. 12:

$$W_{att_i} = \exp\left(-\frac{D_{g_i}^2}{2\sigma^2}\right) \tag{12}$$

where σ is the width parameter of the Gaussian kernel.

(3) **Dot Product Method**

The dot product attention method considers both the direction and the magnitude of the vectors. In this method, we take into account the physical properties of complex domains, and the final similarity score we obtain is Equation 13:

$$W_{att_i} = \frac{\Re(H_{g_i}^T \cdot \text{conj}(H_{g_i}))}{\sqrt{d_k}} \tag{13}$$

where $\sqrt{d_k}$ is the scaling factor used to normalize the dot product, ensuring that the values do not become too large.

In the above three methods, we capture the similarity of frequency vectors that can be explained from a physical point of view. In a specific application, choose one of these methods to calculate the final attention score.

In this paper, the α-Entmax [16] activation function is introduced to implement the soft pruning of attention in order to enhance the sparsity of W_{att_i} to reduce the focus on irrelevant information. This facilitates the sparsity of the attention weight matrix , reducing the offset of attention, as in the Equation 14:

$$W_{att_i}^{\alpha} = \alpha\text{-Entmax}(W_{att_i}) \tag{14}$$

where $W_{att_i}^{\alpha}$ is the sparsified attention score matrix.This sparse operation adaptively removes the parts of the connection that are similarly too small, and in this way adaptively trims the connection.

Finally, the sparsified attention score matrix $W_{att_i}^{\alpha}$ is combined with the selected frequency information H_{g_i} through weighted summation to obtain the attention output matrix in the frequency domain H_{a_i} as Eq. 15:

$$H_{a_i} = \sigma(A_{g_i}) + \sigma(B_{g_i})i = \sigma(W_{att_i}^{\alpha} \cdot A_{g_i}^T)^T + i \cdot \sigma(W_{att_i}^{\alpha} \cdot B_{g_i}^T)^T \tag{15}$$

where A_{g_i} and B_{g_i} are the real and imaginary parts of the selected frequency information H_{g_i} as Eq. 5, respectively, $\sigma(*)$ is the nonlinear function and i is the imaginary unit.

By paying attention to the graph directly in the frequency domain, the model can avoid the sparse interaction of the multi-dimensional information in the context graph network in the state of semantic overlap and achieve stronger global information interaction ability than the context graph attention.

2.2.4 Multi-Scale Information Aggregation Layer

After obtaining the sentence frequency domain tensor H_{a_i} after filtering and interacting, it needs to be used to predict the final relation. We reverse it back into context to get a scale that contains the context domain as Equation 16:

$$H_i^T = \text{IRFFT}(H_{a_i}) \tag{16}$$

where the function IRFFT$(*)$ is the Inverse Real Fast Fourier Transform (IRFFT) and the dimension after the inverse transform is $H_i^T \in \mathbb{R}^{d\times n}$. Next, H_i^T is transposed back to its original shape, yielding the processed contextual representation H_i as Equation 17:

$$H_i = \text{Transpose}(H_i^T) \tag{17}$$

where $H_i \in \mathbb{R}^{n\times d}$.

2.3 Prediction Module

After obtaining the output H_K from the K-th layer, the feature representations of entities e_1 and e_2 need to be extracted. The feature representation of each entity is obtained through the following max-pooling operation:

The feature representation of entities e_1 and e_2 are given as Eq. 18:

$$h_{e_1} = \max(H_{e_1}), \quad h_{e_2} = \max(H_{e_2}) \tag{18}$$

where $H_{e_1} = \{h_{k-i}, h_{k-(i+1)}, \ldots, h_{k-j}\}$, $H_{e_1} \in \mathbb{R}^{(j-i+1)\times d}$ and $H_{e_2} = \{h_{k-p}, h_{k-(p+1)}, \ldots, h_{k-q}\}$, $H_{e_2} \in \mathbb{R}^{(p-q+1)\times d}$ represent the contextual representation corresponding to entity e_1 and e_2 within $H_k = \{h_{k-1}, h_{k-2}, \ldots, h_n\}$, $H_k \in \mathbb{R}^{n\times d}$ respectively.

Next, these two entity feature representations are concatenated with the content of the [CLS] token h_{cls} and input into the final Prediction module. The concatenated feature representation is given as Eq. 19:

$$h_{final} = [h_{cls}; h_{e_1}; h_{e_2}] \tag{19}$$

where $[h_{cls}; h_{e_1}; h_{e_2}]$ denotes the concatenation of the [CLS] token's content with the feature representations of entities e_1 and e_2.

Finally, a fully connected layer is used to classify the concatenated features to predict the relation between the entity pair (or another task objective). The specific classification formula is as Eq. 20:

$$y = \text{softmax}(W_{out} \cdot h_{final} + b_{out}) \tag{20}$$

where y is the probability distribution of the final prediction, W_{out} is the weight matrix of the full connection layer, and b_{out} is the bias term of the full connection layer.

Through these operations, the feature representations of entities e_1 and e_2 are combined with the global information from the [CLS] token and input into the prediction module for the prediction task.

3 Experiments

This chapter provides a detailed overview of the experiment setup, including the datasets used, experiment settings, evaluation metrics, hyper-parameter design, and the baseline models compared to the proposed approach. The specific experimental platform is shown in Appendix A.

3.1 Datasets

SemEval 2010 Task 8[17] is a relation classification dataset containing relation pairs extracted from natural language sentences. The dataset includes 19 relation categories, consisting of 9 initial relations, each with two directional versions, making a total of 2*9 directional relations, along with an "Other" category. The dataset is divided into two parts: the training set with 8,000 samples and the test set with 2,717 samples.

KBP37[18] is a relation classification dataset extracted from news texts, containing 37 relation categories and a total of 21,046 samples. The dataset is split into three parts: 15917 samples for training, 1,724 samples for validation, and 3405 samples for testing. The large number of classes and the increase in sentence length in this dataset pose a greater challenge to relational classification.

3.2 Evaluation Metrics

For the SemEval 2010 Task 8 dataset, the official evaluation standard is followed, with the Macro F1 score as the primary evaluation metric. The Macro-F1 score represents the average of the F1 scores across all classes, making it particularly suitable for scenarios involving imbalanced class distributions. The specific formula is shown in Eq. 21, Eq. 22, and Eq. 23.

$$\mathrm{Pre}_{\mathrm{mac}_i} = \frac{1}{C}\sum_{i=1}^{C}\frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FP}_i} \tag{21}$$

$$\mathrm{Rec}_{\mathrm{mac}_i} = \frac{1}{C}\sum_{i=1}^{C}\frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i} \tag{22}$$

$$\mathrm{F1}_{\mathrm{mac}} = \frac{1}{C}\sum_{i=1}^{C}\frac{2 \times \mathrm{Pre}_{\mathrm{mac}_i} \times \mathrm{Rec}_{\mathrm{mac}_i}}{\mathrm{Pre}_{\mathrm{mac}_i} + \mathrm{Rec}_{\mathrm{mac}_i}} \tag{23}$$

where C is the number of classes, $\mathrm{Pre}_{\mathrm{mac}_i}$ is the precision for class i, and $\mathrm{Rec}_{\mathrm{mac}_i}$ is the recall for class i.

For the KBP37 dataset, the Micro-F1 score is used as the evaluation metric. The Micro-F1 score measures overall accuracy across all test samples, making it appropriate for cases where the class distributions are relatively balanced. The specific formula is shown in Equation 24, Equation 25, and Equation 26.

$$\mathrm{Pre}_{\mathrm{mic}} = \frac{\sum_{i=1}^{C}\mathrm{TP}_i}{\sum_{i=1}^{C}(\mathrm{TP}_i + \mathrm{FP}_i)} \tag{24}$$

$$\mathrm{Rec}_{\mathrm{mic}} = \frac{\sum_{i=1}^{C}\mathrm{TP}_i}{\sum_{i=1}^{C}(\mathrm{TP}_i + \mathrm{FN}_i)} \tag{25}$$

Table 1 Hyper-parameter in two task

Test module	SemEval	KBP37
learning rate	2.2e-05	2.2e-05
batch size	32	32
layers	2	2
warming up	0.07	0.14
weight decay	0.03	0.02
dropout	0.001	0.001
α	1.6	1.9
attention method	Cosine Similarity	Cosine Similarity

$$\mathrm{F1}_{\mathrm{mic}} = \frac{2 \times \mathrm{Pre}_{\mathrm{mic}} \times \mathrm{Rec}_{\mathrm{mic}}}{\mathrm{Pre}_{\mathrm{mic}} + \mathrm{Rec}_{\mathrm{mic}}} \tag{26}$$

where TP_i, FP_i, and FN_i are the true positives, false positives, and false negatives for class i, respectively.

3.3 Hyper-parameter Settings

In the experiments, fine-tuning was performed on the hyper-parameters of the model. Specifically, the hyper-parameter settings of two datasets are shown in Table 1. Additionally, the spectral processing parameters were set to the default configurations of the torch.fft.rfft and torch.fft.irfft modules in the PyTorch library [19]. No modifications were made to these parameters in this study, and all operations were based on the characteristics of the input data. The specific parameter ranges are detailed in Appendix B.

3.4 Baseline models

To verify the validity of the frequency domain model proposed in this paper: FD-GGAN-RE, this chapter will choose the current advanced context-domain graph neural network as the relationship extraction model, and compare the model proposed in this paper. The comparison model is as follows:

C-AGGCN [20] a complete dependency tree structure to construct the attention weights of sentences through multi-head adaptive attention.

C-DAGCN [7] took into account the impact of the distance between words on the model and added the distance as an external knowledge to the graph neural network path weight to guide model update.

C-GCN [21] proposed a dependency tree pruning method with the shortest path center distance K to reduce the influence of noise words and irrelevant words on the model print.

A-GCN [22] incorporated syntactic knowledge into graph neural networks to improve the accuracy of attention weight update.

C-GCN-MG [23] proposed a graph segmentation strategy that divides dependency trees into parts and applies the GCN model to multiple subgraphs to learn information from different parts.

DP-GCN [24] proposed a method to dynamically select whether there is a connection between words through hard gating and the reflection mechanism to achieve the pruning goal.

Table 2 The previous experimental results obtained on this dataset are listed in the table, where * represents the results reproduced under the same hyper-parameters as the FD-GGAN-RE model and $ denotes methods that were not originally used in RE tasks but have been reproduced in the context of RE. ø represents experimental results that are not available on this dataset. Modified method represents the way in which the model is improved on a context-based graph neural network. KG stands for external knowledge, MS stands for model structure and FQ stands for frequency domain method

Test module	Modified method	SemEval			KBP37		
		Pre	Rec	F1	Pre	Rec	F1
C-AGGCN	MS	ø	ø	85.7	ø	ø	ø
C-DAGCN	MS+KG	ø	ø	90.15	68.31	68.13	68.22*
C-GCN	KG	ø	ø	84.8	ø	ø	ø
A-GCN	MS+KG	ø	ø	89.85	68.42	67.80	68.11*
C-GCN-MG	KG	ø	ø	82.4	ø	ø	ø
DP-GCN	MS	ø	ø	86.4	ø	ø	ø
Bi-SDP-Att	MS+KG	83.5	86.4	85.1	ø	ø	64.39
DAGCN	MS+KG	ø	ø	86.0	ø	ø	ø
CEGCN	KG	ø	ø	86.1	ø	ø	ø
WAGCN	MS	ø	ø	87.1	ø	ø	ø
DPR-GHAN	MS+KG	ø	ø	89.53	ø	ø	66.61
SPB	FQ	89.21	89.32	89.27$	66.91	67.33	67.12$
AFS	FQ	89.67	89.61	89.64$	69.46	68.84	68.65$
FD-GGAN-RE	FQ	90.52	90.05	90.33	69.5	68.63	69.06

Bi-SDP-Att [25] proposed a dual syntactic fusion dual graph neural network, using multiple external grammars to improve attention accuracy.

DAGCN [26] proposed a two-graph neural network using a context graph neural network connected by double affine modules and a grammar-dependent graph neural network.

CEGCN [27] proposed a method to guide the graph attention weight by the dependency connection distance between entities and words.

WAGCN [28] proposed a graph attention network with different links, and directly used the attention mechanism to update the weight of attention.

DPR-GHAN [29] Based on the pruning of the original dependency tree, this method expanded the context information of each word and added dependency syntax information for graph attention network modeling.

SPB [30] This method proposes an adaptive gating mechanism for processing vectors after frequency-domain transformation. However, it has not been applied to RE tasks. In this paper, we reproduce this method on RE tasks while ensuring that the input for prediction consists of sentence and entity pair representations.

AFS [31] This method proposes an adaptive gating mechanism to process vectors after frequency-domain transformation, and incorporates a self-attention mechanism to handle the context vectors after inverse transformation. Although this approach has not been applied to RE tasks, in this paper, we reproduce it on RE tasks while ensuring that the input for prediction consists of sentence and entity pair representations.

Table 3 F1 scores of the ablation studies

Test module	SemEval	KBP37
BERT only	88.85	66.08
BERT w/ Frequency Feature Selective Gate layer	89.43	67.32
BERT w/ Frequency-Domain Graph Attention layer	89.86	67.86
BERT w/ All Layers (**FD-GGAN-RE**)	90.33	69.06

3.5 Comparison of results

As shown in Table 2, the FD-GGAN-RE model outperforms the context graph attention and frequency domain model compared to the F1 scores of the SemEval-2010 Task 8 dataset and the KBP37 dataset. This improvement is due to the fact that our model takes into account the errors caused by multi-level information coupling in the context domain. After decoupling by frequency method, adaptive frequency gate is used to filter information and frequency domain graph attention mechanism is used to fine-grained exchange information of different information dimensions, thus making the model more accurate in complex language structure. The improved gating mechanism combined with frequency-domain graph neural networks surpasses the original frequency domain approaches.

4 Analysis

This chapter analyzes the proposed model, focusing on the impact of different modules on performance, particularly the contributions of the frequency-selective gating and frequency attention mechanisms.

4.1 Ablation Studies

By progressively removing or replacing key modules, their impact on performance was observed. The following model configurations were compared: a baseline model using only the BERT encoder, a model incorporating the Frequency Feature Selective Gate layer, a model incorporating the Frequency-Domain Graph Attention layer, and the full model combining both the Frequency Feature Selective Gate layer and Frequency-Domain Graph Attention layer.

The results of the ablation test are shown in Table 3. With the addition of additional modules, the performance of the model is gradually improved. At the same time, we observe that the F1 score of SemEval and KBP37 is increased by 0.58 points and 1.24 points by the introduction of frequency feature selection gate layer alone, and the F1 score is increased by 1.01 and 1.78 points, respectively, by the introduction of frequency domain attention layer alone. The higher performance of the Frequency-Domain Graph Attention layer compared to the Frequency Feature Selective Gate layer indicates that the interaction model is more important than the filtering model for the relationship extraction task. The integration of the filtering and interaction model at the same time can realize the learning of information, and the effect of the model is better.

Table 4 F1 Scores for Different Counts of Frequency-Domain Gate Attention module

Count	SemEval	KBP37
1	89.76	68.21
2	90.33	69.06
3	89.43	68.06
4	88.86	67.64

Table 5 Time and space complexity of the four models

Model	Time Complexity	Space Complexity
BERT Only	$O(bnd_c)$	$O(bnd)$
FD-GGAN-RE	$O(bdn \log n + bn^2 d_h)$	$O(bnd)$
A-GCN	$O(bn^2(d + d_e))$	$O(bn^2(d + d_e))$
C-DAGCN	$O(bn^2(d + d_e + d_{\text{extra}}))$	$O(bn^2(d + d_e + d_{\text{extra}}))$

4.2 Analysis of Frequency-domain Gate Attention module Count and Task Efficiency

This section investigates how the number of Frequency-domain Gate Attention modules impacts model accuracy. By varying the number of these modules, the study evaluates their effect on the model's performance for a given task, aiming to identify the optimal module count. The number of modules was adjusted between 1 and 4, while all other hyper-parameters remained constant. Experiments on two datasets produced the results shown in Table 4.

The findings indicate that the best F1 score is achieved when the module count is set to 2. Increasing the number to three or more layers did not enhance accuracy and led to a slight decline. This decline is likely due to increased model complexity, resulting in overfitting or redundancy, which hindered the model's ability to effectively capture information. On the other hand, using only one module led to insufficient convergence. As a result, two modules were determined to be the optimal choice.

4.3 Speed and Runtime Memory Usage Comparison: Frequency domain model vs. Syntax-Enhanced Model

In this section, the FD-GGAN-RE model is compared in terms of speed and memory to the two most competitive contextual graph neural network improvement models: the A-GCN [22] model, the C-DAGCN [7] model, and the BERT pre-trained model.

As shown in Fig. 3, experimental results show that FD-GGAN-RE model can process the same task significantly faster than the context-based improved graph neural network model. Context graph network improvement models can significantly improve accuracy by integrating external information or more additional connection modules such as multi-head modules, but this improvement comes at the cost of reduced processing speed and increased memory consumption. Since the FD-GGAN-RE model only relies on the sentence itself to realize the adaptive operation, it achieves better performance in terms of time and memory.

As shown in Table 5, despite sharing the same asymptotic time complexity in their relational modeling components, A-GCN and C-DAGCN exhibit significantly slower inference speed compared to BERT Only and FD-GGAN-RE. A-GCN and C-DAGCN incorporate rich

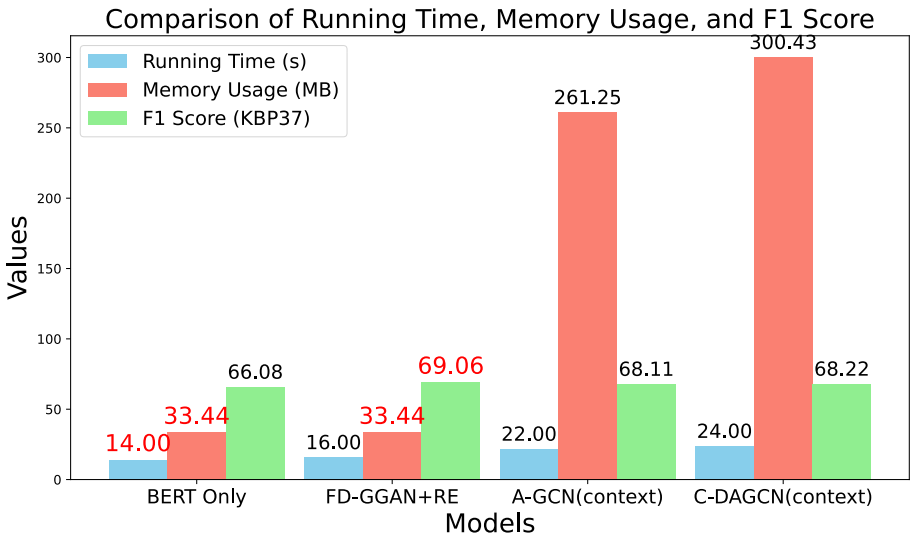

Fig. 3 The total running time(s), memory usage for each batch(MB), and F1 score on the KBP37 test set for the different model tests when the test batch size is 50. The best values have been highlighted

Table 6 F1 Scores for Different frequency domain Attention Methods

Count	SemEval	KBP37
Gaussian Kernel	89.83	68.32
Dot Product	89.97	68.62
Cosine Similarity	90.33	69.06

syntactic representations to enhance linguistic expressiveness. These enhancements are fused into the $n \times n$ pairwise interaction matrix, effectively increasing the hidden dimension from d to $d + d_e + d_{\text{extra}}$. As a result, the constant factor in their time complexity becomes substantially larger, leading to higher computational load per token pair. Moreover, the explicit construction of high-dimensional relational tensors (e.g., $[b, n, n, d + d_e]$) causes a sharp increase in peak memory usage.

4.4 Performance Comparison of Different Attention Methods

In this experiment, the performance comparison of attention mechanisms showed clear differences among the three methods (see Table 6). Cosine similarity achieved the best results, with a noticeable advantage over the other two methods. The dot product method performed moderately well but lagged behind cosine similarity, while the Gaussian kernel method had the lowest accuracy, showing a more significant drop in performance.

Specifically, after the Fourier transform, the numerical ranges of the real and imaginary parts of different vectors may vary significantly due to the characteristics of the signal and the computations involved in the transformation process. In this case, when using dot product or Gaussian kernel methods to compute attention weights, the model may overly focus on the magnitude of the vectors (i.e., semantic intensity) rather than the directional similarity between vectors (i.e., semantic similarity), thereby introducing errors. In contrast, cosine

Table 7 F1 Scores of Different Domain Gates in RE Model

Layers	SemEval	KBP37
Context gate + Context GAT	89.46	66.96
Frequency gate + Context GAT	89.93	68.32
Context gate + Frequency GAT	89.70	67.73
Frequency gate + Frequency GAT (**FD-GGAN-RE**)	90.33	69.06

similarity eliminates the influence of length differences by normalizing the vectors, focusing on the angular similarity between them. This approach is better able to capture semantic relevance, especially in the vector space after Fourier transformation, avoiding attention allocation biases caused by excessive differences between the real and imaginary parts.

Overall, cosine similarity provided a clear advantage, while the dot product method remained competitive but slightly weaker. The Gaussian kernel method, however, demonstrated a substantial performance gap, making it the least favorable choice.

4.5 Effectiveness of selective gate in frequency domain

In order to verify the necessity of fine-grained gating in spectral domain after spectral domain decoupling, we carry out experiments to compare the combinations of different modules in different domains. Four experimental Settings were designed: frequency gate combined with frequency GAT, context gate combined with frequency GAT, frequency gate combined with context GAT, context gate combined with context GAT. It can be divided into two groups: frequency gate and context GAT and context gate and context GAT, frequency gate and frequency GAT and context gate and frequency GAT. The four models are represented on two data sets as shown in Table 7.

According to the experimental results of Table 7, the performance of the model combined with context gates is degraded in the experiment. This phenomenon is similar to [32]. The main reason for this decline is thought to be the way context gates process information. Because each feature point in a context contains interwoven information from multiple layers of context, applying a uniform gating operation can weaken or lose fine-grained information, introducing errors. Instead, frequency gates decouple multiple levels of information by Fourier transform, and then selectively filter the information dimensions to avoid potential information loss that may occur. This also proves the necessity of filtering in the spectral domain.

4.6 Effectiveness of Selective Gate in Feature Dimension

In order to verify the necessity of adaptive learning in the feature dimension of the proposed selection gate, I compare the pre-defined non-adaptive method with the vector-level adaptive filtering: the pre-defined gated vector method and the mean feature filtering method. Here is an introduction to the two methods:

Predefined Gating Vector Method follows [30] and uses a predefined gating vector ξ to filter the frequency components across all feature dimensions uniformly. The input frequency domain feature tensor is $H_{f_i} \in R^{h\times(n/2+1)}$, where h represents the number of feature dimensions and $n/2+1$ is the number of frequency components. The predefined

Table 8 F1 scores of different filtering mechanism in RE model

Filtering mechanism	SemEval	KBP37
Predefined Gating Vector Filtering Method	89.72	67.78
The Mean Feature Vector Filtering Method	89.89	67.85
The Feature Filtering method(**FD-GGAN-RE**)	90.33	69.06

vector $\xi \in R^{n/2+1}$ generates gating values via the gate function, and these values are applied to all feature vectors through element-wise multiplication. After those operations, the selected frequency information H_{g_i} is as Equation 27:

$$H_{g_i} = A_{f_i} \odot \sigma(\xi) + (B_{f_i} \odot \sigma(\xi))i \tag{27}$$

where $\sigma(\xi)$ represents the output of the predefined vector after passing through the function Equation 4.

Mean Feature Vector Filtering Method first calculates the mean of each frequency component across the feature dimensions. The input tensor H_{f_i} is averaged along the feature dimensions to obtain $\overline{H}_{f_i} \in R^{n/2+1}$. This mean feature representation is then divided into the real part $\overline{A}_{f_i}$ and the imaginary part $\overline{B}_{f_i}$. The frequency-domain full connection layers are applied to these components using weight matrices W_a and W_b, with an added bias term b. The vector filtering is then performed using the Equation 4 function. As shown in the Equation 28, the selected frequency information H_{g_i} is:

$$\begin{aligned} H_{g_i} = {} & A_{f_i} \odot \sigma\left((W_a \cdot \overline{A}_{f_i}) + (W_b \cdot \overline{B}_{f_i}) + b\right) \\ & + (B_{f_i} \odot \sigma\left((W_a \cdot \overline{A}_{f_i}) + (W_b \cdot \overline{B}_{f_i}) + b\right))i \end{aligned} \tag{28}$$

As shown in Table 8, the proposed Feature Filtering method have the highest performance, and the overall performance of the Mean Feature Vector Filtering method is slightly better than that of the Predefined Gating Vector Filtering method. This proves the necessity of filtering from the feature dimension. The predefined gating vector method uses a fixed gating mechanism for all feature dimensions, but it cannot be dynamically adjusted according to different word vector distributions. This may account for its weaker performance. The mean eigenvector filtering method enhances adaptability under different distributions by learning the average variation of the feature dimensions. However, this method may lose the distribution change information specific to each feature dimension, and the performance is lower than the feature dimension filtering effect.

4.7 Effectiveness of Attention in Frequency domain

A comparative experiment was conducted on the model in both the context and frequency domains to validate the necessity of applying the fine-grained graph attention mechanism in the frequency domain. Specifically, after performing the frequency gating operation, the inverse Fourier transform was immediately applied to restore the contextual information, followed by the attention operation to compute the attention matrix. For this attention matrix, several pruning methods were tested, including pruning based on the shortest dependency path (SDP), the shortest dependency path alone, entity-centered pruning with a distance of 1 (SDP& location word)[22], and no pruning. Unlike time-domain models, current frequency-

Table 9 F1 Scores of Attention in Two Domains

Attention Domain	Model	F1 Score	
		SemEval	KBP37
Context domain	No pruning	89.93	68.32
	SDP	89.52	67.77
	SDP& location word	89.63	68.32
Frequency domain	**FD-GGAN-RE**	90.33	69.06

domain models lack highly interpretable pruning schemes, such as those based on entity-pair's short dependency path pruning. Comparative experiments were conducted on two datasets with the results shown in Table 9.

The experimental results show that although the context-domain graph attention mechanism can capture some key information, its performance lags behind that of the frequency-domain attention mechanism. This phenomenon proves the inadequacy of context graph attention in dealing with word vectors with multiple layers of interweaving context information in implicit space.

Through the comparison of different pruning methods without context pruning, it is found that the effect of pruning only considering SDP is not as good as considering SDP + position words, and the performance of both methods decreases to a certain extent. From a context perspective, overpruning can remove important information and degrade model performance. From the perspective of poor compatibility between pruning methods and frequency domain gating, pruning-based graph attention ignores the global spectrum information of the context, which also leads to performance degradation. This also shows that adding additional knowledge in the context brings additional noise to the model and does not address the core deficiency of coarse-grained attention.

4.8 Case study

In this case, two sentences are selected to verify the effect of the Frequency Feature Selective Gate layer and the Frequency-Domain Graph Attention layer. The selected two sentences are from the SemEval test set and the kbp37 test set. The first original sentence is: "The dramatic <e1> streaks </e1> we see in the sky are caused by <e2> particles </e2> that incinerate before they hit the ground." The marking relation of this sentence is Cause-Effect(e2, e1). Figure 4 shows the Attention weights generated by the Frequency-Domain Graph Attention layer w/o and w/ the Frequency Feature Selective Gate layer. Without frequency gating, the model failed to accurately capture causality in the sentence, incorrectly predicting it to be of the Other type. After adding gating, the model successfully identifies the correct Cause-Effect(e2, e1) relation. The second sentence is: "After local newspapers reported that funds from <e1> Arizona </e1> State ' s foundation were going to support undocumented students and that public funds may have been involved in administering the aid Mr . <e2> Martin </e2> says his office was flooded with hundreds of calls including about a dozen from donors to the university who were very upset". If no gating is added, the relation is predicted as "per:stateorprovinces_of_residence(e2,e1)", and if gating is added, the relation is predicted as "per:employee_of(e2,e1)". The real relation is "no relation". Figure 6 shows the Attention weights of the sentence w/o and w/ the Frequency Feature Selective Gate layer, respectively. In addition, the entire frequency band is divided into four frequency

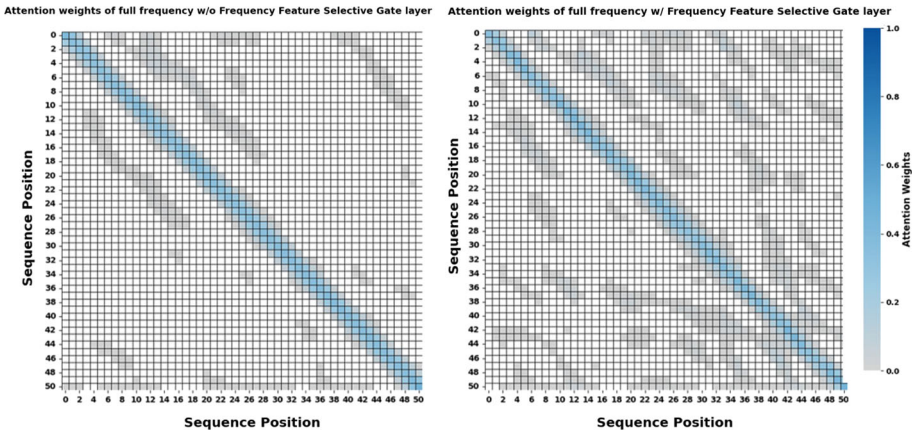

Fig. 4 Frequency-Domain Graph Attention weight of "The dramatic <e1> streaks </e1> we see in the sky are caused by <e2> particles </e2> that incinerate before they hit the ground."

bands: low frequency [0, 6), mid frequency [6, 13), mid frequency [13, 26)and high frequency [26, 50), corresponding to four semantic components: sentence level, clause level, phrase level and word level respectively to illustrate semantic interaction. By averaging all attention components in the frequency band, the attention score of each frequency in the four frequency bands is obtained, as shown in Fig. 5 and Fig. 7.

As shown in Figs. 4 and Fig. 6, the frequency-domain graph attention mechanism shares a common feature with the traditional context graph attention mechanism: both tend to assign a higher attention weight to the area near the diagonal, reflecting the attention to their own information. Moreover, even with the addition of frequency feature selection gates, frequency domain graph attention pays more attention beyond the diagonal than contextual attention. This broad focus is a key factor in the superiority of frequency-domain graph attention over traditional contextual attention. Due to the information decoupling, the frequency domain graph attention is not subject to the error caused by the overlap of multi-level information. At this time, the wide distribution of attention focuses on the frequency vector reflecting the change of global multi-level information, which is in sharp contrast to the coarse-grained word vector interaction in the context-based graph attention model.

We use sentence 1 to illustrate how the model behaves on a dataset with short sentences. When the model of Frequency Feature Selection Gate layer is added, as shown in Fig. 5, the attention scores of word level and phrase level hardly change. The attention of low-frequency sentence-level information to clave-level information decreased by 30%, the attention to phrase-level information increased by 57%, and the attention to word-level information increased by 900%. At the same time, the attention of clause information to phrase information increased by 30%, and the attention to word information increased by 800%. This interference is not an increase in noise, but an optimization of balanced information at all levels. And phrase information. From the semantic level of the word, the similarity of the entity pairs "stripes" and "particles" in sentence 1 is low. If there is no frequency feature to select the gate layer model, then the model tends to predict the wrong answer. This change in focus allows the model to focus more on the core message of "caused" at the word level and "caused by" at the phrase level.

We use sentence 1 to illustrate how the model behaves on a dataset with long sentences. As shown in Fig. 7, low-frequency sentence-level information's attention to itself increased by

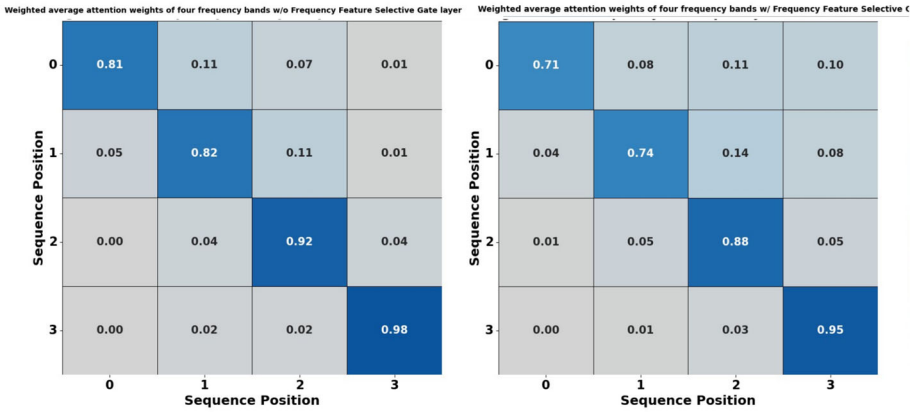

Fig. 5 Weighted average attention weight of "The <e1> singer </e1> demonstrates his sensitivity during the <e2> song </e2> by suggesting that he would bring flowers."

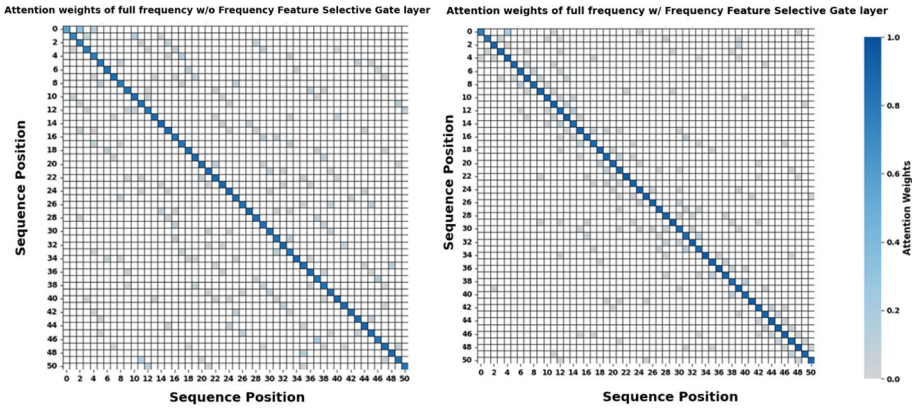

Fig. 6 Frequency-Domain Graph Attention weight of "After local newspapers reported that funds from <e1> Arizona </e1> State ' s foundation were going to support undocumented students and that public funds may have been involved in administering the aid Mr . <e2 > Martin </e2> says his office was flooded with hundreds of calls including about a dozen from donors to the university who were very upset."

10%, clave-level information's attention to itself increased by 12.5%, phrase-level information's attention to itself increased by 8%, and word information's attention to itself increased by 4%. We infer that this is due to the overall length of the kbp37 data set being higher than the Semeval data set. In this case, sentence-level, clause-level, and phrase-level information will have more task-independent information shifts due to the length of the sentence. However, from the perspective of frequency, although the frequency feature selection gate improves the performance of the model by limiting the attention of the model to the frequency information similar to itself, it also brings shortcomings. From the perspective of the context domain, the change of gated selection and attention due to the change of sentence length distribution will increase the attention weight deviation caused by similar semantics. This also shows that more modules are needed to constrain the information errors caused by long sentences.

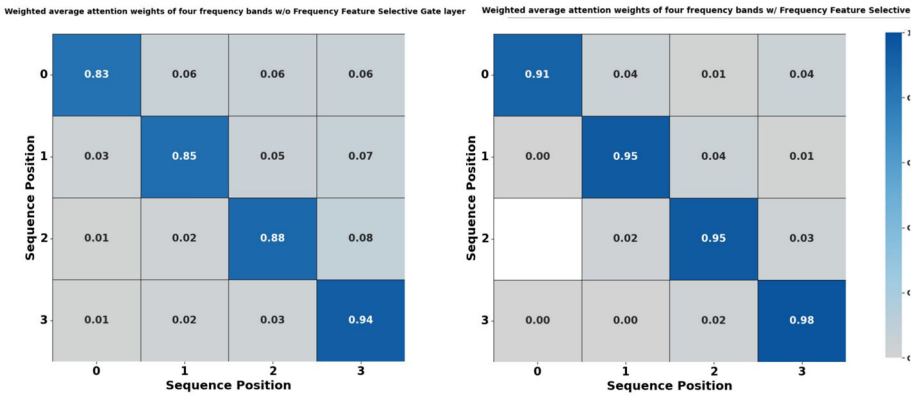

Fig. 7 Weighted average attention weight of "After local newspapers reported that funds from <e1> Arizona </e1> State ' s foundation were going to support undocumented students and that public funds may have been involved in administering the aid Mr . <e2> Martin </e2> says his office was flooded with hundreds of calls including about a dozen from donors to the university who were very upset."

5 Related Work

5.1 Frequency Domain Methods in Natural Language

Modeling text context from the frequency domain is a new approach to understanding natural language tasks. [33] proposes a spectral perspective method for constructing word sequence embedment. [11] and [30] propose manual selection and adaptive frequency filters to filter frequency vectors, respectively. [13] and [32] introduce adaptive vector filters to filter the context frequency vector and word feature vector respectively in the aspect based sentiment analysis task to increase the context attention and exchange information. All of these methods consider the direction level, not the feature level, which can lose information. Aggregate the completed information directly and model the information interaction using context diagrams. This method ignores the coarse-grained attention of context graph neural networks, which lacks the in-depth study of frequency domain tasks.

5.2 Context based Graph Attention Neural Networks in Relation Extraction

The core challenge of graph neural network in relation extraction task is how to effectively assign interaction weights between nodes in the process of attention mechanism to ensure reasonable interaction between different parts of the sentence. [20] suggests modeling tasks using multi-head graph attention networks. In order to further improve the performance of the model, a dense connection mechanism is introduced to change the graph network link mode to obtain more information. [28] proposes a hierarchical connection method on the basis of words to obtain different levels of information. However, this approach of adding more layers by adding more links causes the model to execute slower and also causes the model to introduce more intensive information resulting in irrelevant errors.

Another way is to provide additional prior knowledge to the model by introducing external information, which improves the accuracy of attention weight distribution. Therefore, the combination of external knowledge and graph neural network is also applied to the relation extraction task [23]. Considering the different importance of different position information in

the task, the graph neural network of the sentence is split into multiple subgraphs according to the relationship between the word to be predicted and the entity pair. [22] incorporates dependency types into the network to adjust the allocation of attention weights and integrates them into word representations to improve accuracy. Building on this method, [7] further includes information about the distance between words. [29] considers the importance of context and adds a further layer of context information to each graph neural network layer that already contains external information, thus expanding the scope of information interaction. These methods are limited in improving the accuracy of the model, and they also introduce the prior information constructed by non-relational extraction tasks, which leads to the increase of preprocessing overhead and computational cost while introducing additional errors.

[26] proposes the use of multiple models to combine multiple external knowledge separately. [23] Changes to the model were made while external knowledge was introduced. But these approaches do not escape the core problem contained above. At the same time, the introduction of these two methods is to fit the features of the sentence context by introducing more structures without considering the embedding layer, which leads to multiple layers of information overlapping. This is also the shortcoming of the above method.

6 Limitations and Future Work

This study focuses on Fourier-based frequency-domain methods for relation extraction but lacks exploration of diverse frequency variations and evaluation on broader NLP tasks. Future work will explore advanced frequency techniques, such as wavelet transforms, and extend applications to tasks like entity recognition and sentiment analysis, potentially paving the way for more innovative research directions.

7 Conclusion

This paper proposes a novel Frequency-Domain aware Gated Graph Attention Network for Relation Extraction (FD-GGAN-RE), designed to address the limitation of conventional context-based graph attention networks in fine-grained modeling of multi-level semantic information—a shortcoming that often causes the attention mechanism to deviate from the true semantic focus. By decomposing semantic signals in the frequency domain, our model explicitly separates high-frequency components (corresponding to fine-grained, local features) from low-frequency components (representing global, structural features), effectively reducing semantic redundancy and enabling targeted modeling at different semantic granularities. Experimental results show that FD-GGAN-RE achieves F1 scores of 90.33 on SemEval 2010 Task 8 and 69.06 on KBP37, outperforming state-of-the-art context-based graph neural network methods while also significantly improving inference speed and reducing GPU memory consumption. Furthermore, analyses on both long and short sentences demonstrate that the model can accurately identify and suppress redundant semantic information, highlighting its strong interpretability.These results validate the effectiveness of integrating frequency-domain analysis into relation extraction. We believe this approach opens up new directions for future research in this field. Moreover, frequency-domain graph attention holds promise for advancing other NLP tasks beyond relation extraction, such as sentiment analysis and machine translation.

Appendix A Computing Platform and Environment Configuration

The following Table 10 provides detailed information about the computing platform and environment configuration used in this study.

Table 10 Computing Platform and Environment Configuration

Category	Details
CPU	16 vCPU Intel(R) Xeon(R) Platinum 8352V CPU @ 2.10GHz
GPU	RTX 4090
Memory	90GB
Storage	50GB
ubuntu	22.04
Python	3.7.16
PyTorch	1.12.0
CUDA	11.3
cuDNN	8.3.2

Appendix B Hyper-parameter Tuning Range and Step Size

This paper employs a grid search method to systematically explore all possible combinations of hyper-parameters within predefined ranges in order to determine the optimal configuration. The hyperparameter for the attention method is selected from three options: Dot Product Method, Gaussian Kernel Method, and Cosine Similarity Method. For numerical hyper-parameters, their tuning ranges are specified in Table 11.

Table 11 Hyperparameter Tuning Range and Step Size

Hyperparameter	Tuning Range	Step Size
learning rate	$[2.0 \times 10^{-5}, 3.0 \times 10^{-5}]$	1.0×10^{-6}
batch size	$2^{[3,6]}$	1
layers	[1, 4]	1
Warming up	[0.01, 0.2]	0.01
weight decay	[0.01, 0.1]	0.01
dropout	[0.001, 0.01]	0.001
α	[1.0, 2.0]	0.1

Author Contributions ZY and WG contributed equally to this work. They were primarily responsible for developing the methodology and drafting the initial manuscript. Their efforts laid the foundation for the research presented in this study. HY, ZS, and TC played key roles in revising and refining the manuscript. Their critical feedback and edits significantly improved the quality and clarity of the final document. TC also provided overall guidance and supervision during the revision process. All authors have read and approved the final version of the manuscript.

Funding This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data Availability No datasets were generated or analysed during the current study.

Declarations

Conflicts of Interest The authors declare no Conflict of interest.

References

1. Yih W-T, Chang M-W, He X, Gao J (2015) Semantic parsing via staged query graph generation: Question answering with knowledge base. In: Zong C, Strube M (eds) Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Association for Computational Linguistics, Beijing, China, pp 1321–1331. https://doi.org/10.3115/v1/P15-1128
2. Baeza-Yates R, Ribeiro-Neto B (2011) Modern Information Retrieval: The Concepts and Technology Behind Search, 2nd edn. Addison-Wesley Publishing Company, USA
3. Yu M, Yin W, Hasan KS, Santos C, Xiang B, Zhou B (2017) Improved neural relation detection for knowledge base question answering. In: Barzilay R, Kan M-Y (eds.) Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 571–581. Association for Computational Linguistics, Vancouver, Canada. https://doi.org/10.18653/v1/P17-1053 . https://aclanthology.org/P17-1053
4. Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G (2009) The graph neural network model. IEEE Trans Neural Networks 20(1):61–80. https://doi.org/10.1109/TNN.2008.2005605
5. Kipf TN, Welling M (2017) Semi-Supervised Classification with Graph Convolutional Networks
6. Veličković P, Cucurull G, Casanova A, Romero A, Liò P, Bengio Y (2018) Graph Attention Networks. arXiv:1710.10903
7. Liao J, Du Y, Hu J, Li H, Li X, Chen X (2024) A contextual dependency-aware graph convolutional network for extracting entity relations. Expert Syst Appl 239:122366. https://doi.org/10.1016/j.eswa.2023.122366
8. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. NIPS'17, pp. 6000–6010. Curran Associates Inc., Red Hook, NY, USA
9. Chen W, Wang C, Xu K, Yuan Y, Bai Y, Zhang D (2024) D-fast: Cognitive signal decoding with disentangled frequency–spatial–temporal attention. IEEE Trans Cognitive Development Syst 16(4):1476–1493. https://doi.org/10.1109/TCDS.2024.3370261
10. Xu K, Qin M, Sun F, Wang Y, Chen Y-K, Ren F (2020) Learning in the frequency domain. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)
11. Tamkin A, Jurafsky D, Goodman N (2020) Language Through a Prism: A Spectral Approach for Multiscale Language Representations. arXiv:2011.04823
12. Devlin J, Chang M-W, Lee K, Toutanova K (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: Burstein J, Doran C, Solorio T (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota. https://doi.org/10.18653/v1/N19-1423 . https://aclanthology.org/N19-1423

13. Niu H, Xiong Y, Wang X, Yu W, Zhang Y, Guo Z (2023) Adaptive structure induction for aspect-based sentiment analysis with spectral perspective. In: Bouamor H, Pino J, Bali K (eds.) Findings of the Association for Computational Linguistics: EMNLP 2023, pp. 1113–1126. Association for Computational Linguistics, Singapore . https://doi.org/10.18653/v1/2023.findings-emnlp.79 . https://aclanthology.org/2023.findings-emnlp.79
14. Wojke N, Bewley A (2018) Deep cosine metric learning for person re-identification. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 748–756 . https://doi.org/10.1109/WACV.2018.00087
15. Li R, Wang S, Zhu F, Huang J (2018) Adaptive Graph Convolutional Neural Networks. arXiv:1801.03226
16. Peters B, Niculae V, Martins AFT (2019) Sparse sequence-to-sequence models. In: Korhonen A, Traum D, Màrquez L (eds.) Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 1504–1519. Association for Computational Linguistics, Florence, Italy. https://doi.org/10.18653/v1/P19-1146 . https://aclanthology.org/P19-1146
17. Hendrickx I, Kim SN, Kozareva Z, Nakov P, Ó Séaghdha D, Padó S, Pennacchiotti M, Romano L, Szpakowicz S (2010) SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In: Erk K, Strapparava C (eds) Proceedings of the 5th International Workshop on Semantic Evaluation. Association for Computational Linguistics, Uppsala, Sweden
18. Angeli G, Tibshirani J, Wu J, Manning CD (2014) Combining distant and partial supervision for relation extraction. In: Moschitti A, Pang B, Daelemans W (eds) Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Doha, Qatar, pp 1556–1567. https://doi.org/10.3115/v1/D14-1164
19. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
20. Guo, Z., Zhang, Y., Lu, W.: Attention guided graph convolutional networks for relation extraction. In: Korhonen, A., Traum, D., Màrquez, L. (eds.) Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 241–251. Association for Computational Linguistics, Florence, Italy (2019). https://doi.org/10.18653/v1/P19-1024 . https://aclanthology.org/P19-1024
21. Zhang Y, Qi P, Manning CD (2018) Graph convolution over pruned dependency trees improves relation extraction. In: Riloff E, Chiang D, Hockenmaier J, Tsujii J (eds) Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Brussels, Belgium, pp 2205–2215. https://doi.org/10.18653/v1/D18-1244
22. Tian Y, Chen G, Song Y, Wan X (2021) Dependency-driven relation extraction with attentive graph convolutional networks. In: Zong C, Xia F, Li W, Navigli R (eds.) Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 4458–4471. Association for Computational Linguistics, Online. https://doi.org/10.18653/v1/2021.acl-long.344 . https://aclanthology.org/2021.acl-long.344
23. Mandya A, Bollegala D, Coenen F (2020) Graph convolution over multiple dependency sub-graphs for relation extraction. In: Scott D, Bel N, Zong C (eds) Proceedings of the 28th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Barcelona, Spain, pp 6424–6435. https://doi.org/10.18653/v1/2020.coling-main.565 (**Online**)
24. Yu B, Mengge X, Zhang Z, Liu T, Yubin W, Wang B (2020) Learning to prune dependency trees with rethinking for neural relation extraction. In: Scott D, Bel N, Zong C (eds) Proceedings of the 28th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Barcelona, Spain, pp 3842–3852. https://doi.org/10.18653/v1/2020.coling-main.341 (**Online**)
25. Li D, Lei Z-L, Song B-Y, Ji W-T, Kou Y (2022) Neural attentional relation extraction with dual dependency trees. J Comput Sci Technol 37(6):1369–1381. https://doi.org/10.1007/s11390-022-2420-2
26. Zhang D, Liu Z, Jia W, Wu F, Liu H, Tan J (2024) Dual attention graph convolutional network for relation extraction. IEEE Trans Knowl Data Eng 36(2):530–543. https://doi.org/10.1109/TKDE.2023.3289879
27. Long J, Liu L, Fei H, Xiang Y, Li H, Huang W, Yang L (2022) Contextual semantic-guided entity-centric gcn for relation extraction. Mathematics 10(8):1344. https://doi.org/10.3390/math10081344
28. Dong Y, Xu X (2023) Weighted-dependency with attention-based graph convolutional network for relation extraction. Neural Process Lett 55(9):12121–12142. https://doi.org/10.1007/s11063-023-11412-z
29. Li N, Wang Y, Liu T (2024) Dependency-position relation graph convolutional network with hierarchical attention mechanism for relation extraction. J Supercomput 80(13):18954–18976. https://doi.org/10.1007/s11227-024-06204-8
30. Müller-Eberstein M, Goot R, Plank B (2022) Spectral Probing. arXiv:2210.11860
31. Sun K, Zhang R, Mao Y, Mensah S, Liu X (2020) Relation extraction with convolutional network over learnable syntax-transport graph. Proc AAAI Conference Artifi Intell 34(05):8928–8935. https://doi.org/10.1609/aaai.v34i05.6423

32. Niu H, Wang M, Xiong Y, Yang B, Jia X, Guo Z (2024) Linking adaptive structure induction and neuron filtering: A spectral perspective for aspect-based sentiment analysis. In Calzolari N, Kan M-Y, Hoste V, Lenci A, Sakti S, Xue N (eds.) Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pp. 10584–10597. ELRA and ICCL, Torino, Italia. https://aclanthology.org/2024.lrec-main.926
33. Kayal S, Tsatsaronis G (2019) EigenSent: Spectral sentence embeddings using higher-order dynamic mode decomposition. In: Korhonen A, Traum D, Màrquez L (eds) Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Florence, Italy, pp 4536–4546. https://doi.org/10.18653/v1/P19-1445