Artificial Intelligence Review (2024) 57:233
https://doi.org/10.1007/s10462-024-10835-x

An effective multi-modal adaptive contextual feature information fusion method for Chinese long text classification

Yangshuyi Xu[1] · Guangzhong Liu[1] · Lin Zhang[1] · Xiang Shen[1] · Sizhe Luo[1]

Published online: 6 August 2024

Abstract
Chinese long text classification plays a vital role in Natural Language Processing. Compared to Chinese short texts, Chinese long texts contain more complex semantic feature information. Furthermore, the distribution of these semantic features is uneven due to the varying lengths of the texts. Current research on Chinese long text classification models primarily focuses on enhancing text semantic features and representing Chinese long texts as graph-structured data. Nonetheless, these methods are still susceptible to noise information and tend to overlook the deep semantic information in long texts. To address the above challenges, this study proposes a novel and effective method called MACFM, which introduces a deep feature information mining method and an adaptive modal feature information fusion strategy to learn the semantic features of Chinese long texts thoroughly. First, we present the DCAM module to capture complex semantic features in Chinese long texts, allowing the model to learn detailed high-level representation features. Then, we explore the relationships between word vectors and text graphs, enabling the model to capture abundant semantic information and text positional information from the graph. Finally, we develop the AMFM module to effectively combine different modal feature representations and eliminate the unrelated noise information. The experimental results on five Chinese long text datasets show that our method significantly improves the accuracy of Chinese long text classification tasks. Furthermore, the generalization experiments on five English datasets and the visualized results demonstrate the effectiveness and interpretability of the MACFM model.

Keywords Chinese long text classification · Graph convolutional network · Modal interaction · Noise information filtering · Adaptive modal feature fusion

1 Introduction

The significance of text in advancing humanity and civilization is evident, as it serves as a fundamental medium for documenting human history. In the contemporary era of vast data, it is crucial to manage and classify the immense amount of textual information effectively (Li et al. 2022; Duarte and Berton 2023). As artificial intelligence rapidly evolves, Chinese

Extended author information available on the last page of the article

text classification technology has made consistent progress and is extensively applied in various fields, including healthcare (Fernandes et al. 2023), digital libraries (Jiang et al. 2022), sentiment analysis (Gautam et al. 2022), etc.

The main goal of the Chinese long text classification task is to use computers to identify and comprehend the characteristics of Chinese long texts. The task also aims to explore the underlying connections between the semantic features of texts and their classes in depth. This task requires computers to accurately capture and fully comprehend contextual information and unevenly distributed semantic information from long texts. Based on the text length, Chinese long texts contain more abundant and intricate semantic features than Chinese short texts. Additionally, the uneven distribution of these semantic features in Chinese long texts is more pronounced. The semantic feature information at different positions in long texts is often difficult to capture and comprehend effectively, directly impacts the final classification accuracy of the model. As a result, extracting rich, complex, and unevenly distributed semantic feature information from long Chinese texts and then deeply mining their high-level semantic features is a significant challenge faced by Chinese long text classification tasks.

The attention mechanism (Niu et al. 2021) is a technique that mimics the human ability to focus on specific aspects of objects and assign varying levels of importance to different inputs. Its purpose is to enhance feature selection, thereby improving model predictions' accuracy. The development of the attention mechanism has dramatically influenced the field of Natural Language Processing (NLP), particularly with the introduction of the Transformer model (Vaswani et al. 2017) and BERT model (Devlin et al. 2019).These models have been widely applied in various visual and multi-modal tasks (Liu et al. 2023; Xu et al. 2023b; Dosovitskiy et al. 2021) as researchers continue to explore their potential. Regarding long text classification tasks in Chinese, Deng et al. (2021) proposed the ABLG-CNN model, which combines the attention mechanism with Bi-LSTM, Convolutional Neural Network (CNN), and gated mechanism. This model effectively filters unnecessary information while accurately capturing the context and characteristics of local phrases in long texts. Similarly, Yang et al. (2023) introduced the feature-enhanced text-inception model designed explicitly for Chinese long text classification. This model identifies and processes the inherent feature information in long Chinese texts. Additionally, Chen et al. (2022) presented the Local Feature Convolution Network (LFCN), which builds upon BERT and is tailored for Chinese long texts. This network can extract local features, such as key phrases, and efficiently integrate them from long Chinese texts. The Graph Convolutional Network (GCN) (Bhatti et al. 2023) is a graph network model that applies convolution operations from traditional data (such as images and texts) to graph-structured data. It leverages the relationships between graph nodes to learn more complex text features in text classification tasks. Yao et al. (2019) introduced the TextGCN model, which utilizes GCN to capture the connections between words, words and documents, and documents in the graph; Liu et al. (2020) developed an approach to represent Chinese long texts as graph-structured using the concept interaction graphs and successfully applied GCN for classification; To address resource consumption during graph construction, Huang et al. (2019) established globally shared graph parameters for texts based on the TextGCN model (Yao et al. 2019); Additionally, Liu et al. (2020) proposed a method to construct multi-graph representations, including semantic graphs and context-related information graphs, achieving favorable results in text classification tasks; Furthermore, Yang et al. (2022) proposed to introduce contrastive learning and adaptive enhancement strategies into graph-based text classification models, resulting in improved classification performance and reducing resource consumption. Zhang et al. (2023) developed a new ProbSparse self-attention Transformer

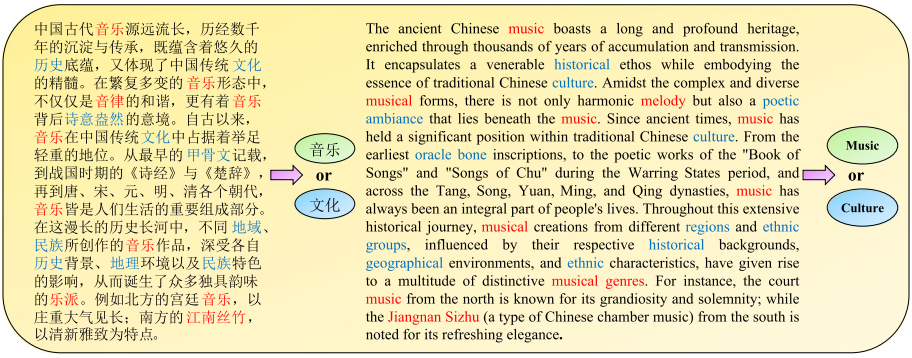

Fig. 1 Chinese long text classification task examples. The category of this long text should be music; red and blue represent musical and cultural characteristics, respectively. (Color figure online)

model named SpaL Transformer, with KL divergence and adding long short-term memory (LSTM) to positional encoding to focus the overall attention. The cognitively inspired multi-granularity model incorporating label information (LIMG) model (Gao et al. 2024) is proposed to obtain tag semantics from the abstract and extract multi-granularity features, enhancing text representation and stimulating cognitive systems to understand complex information in the abstract.

The Chinese long text classification model introduced above mainly researches contextual semantics and local feature representation, text feature enhancement, and using GCN to process graph-structured text data. These methods can capture and understand semantic features of Chinese long texts to a certain extent. However, they still can not thoroughly learn the rich and complex semantic feature information in Chinese long texts, resulting in a lack of in-depth understanding of the semantic features. Moreover, the above methods still consider less irrelevant information in Chinese long texts, causing the model's performance to capture text feature information easily affected by redundant information such as noise information. In real-life scenarios, Chinese long text classification models will encounter more complex situations, such as more irrelevant noise information in the text, longer text length, more complex semantic features, uneven distribution, etc. Therefore, Chinese long text classification models must have better noise information filtering capabilities, long-distance dependency capture capabilities, and semantic feature extraction and mining capabilities. As depicted in Fig. 1, the Chinese long text contains additional semantic elements beyond the core semantic features of the topic, which presents a challenge for the model to capture the full range of feature information effectively. This issue can adversely affect the model's depth of understanding regarding critical feature information in the text.

This paper considers that combining multi-modal fine-grained fusion of Chinese long text features and the adaptive feature fusion mechanism can further improve the overall performance of the Chinese long text classification model. Based on the distribution characteristics of semantic features of Chinese long texts and the problem that GCN tends to ignore contextual feature information in Chinese long text classification tasks, inspired by related research (Dosovitskiy et al. 2021; Woo et al. 2018; Lin et al. 2021; Arevalo et al. 2020; Dhingra et al. 2017), this article suggests a new and effective Multi-modal Adaptive Contextual Feature Information Fusion Method (MACFM) for Chinese long text classification. This method can effectively capture the representations of context features and graph

node features in Chinese long texts and then adaptively fuse the two feature representations. Specifically, in terms of text feature representation, we first introduce the RoBERTa model (Liu et al. 2019) to initially extract text features of Chinese long texts, then the Bi-GRU network is used to fully capture and learn the dependency relationships in contextual features, and finally we propose the Deep Contextual Feature Attention Module (DCAM) to perform fine-grained deep mining of keyword features and critical context feature information; In the graph node feature representation part, we effectively optimize the size of the graph structure by combining the method of limiting the dictionary size during the graph construction process, the word vector features extracted by the RoBERTa model are used as the initial feature matrix at the same time, so that the GCN can effectively learn the text semantic features and positional information, and then an effective layer residual unit is introduced, which can make the GCN model to conduct in-depth learning and understanding of the rich feature information in the graph nodes; In terms of modal fusion, by proposing an Adaptive Modal Feature Information Fusion Module (AMFM), it can effectively integrate multi-modal semantic features of texts while improving the interaction ability between modalities and the robustness of model.

In summary, the contributions of this paper are as follows:

(1) A deep contextual feature attention module is proposed, which can effectively thoroughly learn and understand the unevenly distributed feature semantic information in Chinese long texts, capture the keywords and critical contextual information, and at the same time filter out some irrelevant noise information, thereby effectively enhancing modal interactive learning capabilities and improving the fusion efficacy of modal features in the feature fusion stage;
(2) When constructing a graph on a Chinese long text dataset, the more robust Positive Pointwise Mutual Information (PPMI) method is used to build the relationship measurement between words in the text. At the same time, the word frequency statistical method is used to limit the size of the text dictionary, which can effectively optimize the size of the graph structure of the texts and save computing overhead;
(3) Introducing an effective layer residual unit in GCN can effectively compensate for the problem that shallow GCN cannot thoroughly learn the information of high-order neighbor nodes, making GCN learn text feature representations more effectively. This unit can also avoid the gradient disappearance problem when the number of layers of GCN is deepened;
(4) The proposed adaptive modal feature information fusion module for modal fusion aims to merge feature information from both texts and graph nodes adaptively. This module enables the model to extract high-level semantic features, effectively remove irrelevant information, and ultimately enhance the model's ability to comprehend the complex semantics in Chinese long texts deeply;
(5) The MACFM model has demonstrated favorable results on the Chinese long text datasets Iflytek, INews, THUCNews, SogouCS, and Fudan, achieving accuracy rates of 70.73%, 89.87%, 98.22%, 96.89%, and 98.13%, respectively. Moreover, the MACFM model's validity and interpretability are further supported by conducting generalization experiments on five English datasets, performing MACFM module ablation experiments, and analyzing the results through visualization.

The remaining parts of this paper are organized as follows: Sect. 2 provides an overview of recent research that is directly relevant to our study; Sect. 3 defines the research

problem of this article; Sect. 4 gives detailed insights into the MACFM method we propose, including each component of the model and the associated techniques; Sect. 5 presents the experimental configuration, all relevant experiments and visualization results to demonstrate the effectiveness of our model; Sect. 6 gives a brief discussion of our model; Sect. 7 concludes our study and provides an outlook on potential avenues for future exploration.

2 Related works

2.1 Chinese long text classification

The Chinese long text classification task aims to accurately predict the category label of a given text, understanding the feature information through feature extraction and learning. Due to the complex semantic features and uneven distribution in Chinese long texts compared to Chinese short texts, this task requires the model to capture long-distance dependencies and have strong feature information mining capability (Zhang et al. 2022). In early Chinese long text classification tasks, machine learning methods were utilized (Kramer 2011; Suthaharan and Suthaharan 2016; Song and Ying 2015), requiring significant time and effort to construct feature engineering on text data. Moreover, these methods often used discrete forms of text representation, causing issues such as insufficient semantic mining of long texts and limited understanding of contextual information, thereby affecting the model's performance. Deep learning methods, particularly those incorporating the attention mechanism, have been extensively employed in Chinese long text classification research. The attention mechanism is highly effective in capturing feature information and filtering noise in text classification tasks, enhancing the model's comprehension of text feature information. In a pioneering study (Yang et al. 2016), the hierarchical attention mechanism was applied to text classification tasks, focusing on extracting word and sentence features in the text separately. Another study (Guo et al. 2018) proposed a hybrid attention mechanism for fine-grained extraction of text semantic features. Additionally, a Chinese text classification method that combines feature enhancement with attention mechanisms to identify key features of texts was suggested in a separate study (Xie et al. 2020).

Recently, the development of pre-training techniques has significantly enhanced text classification models. These models can extract and assimilate a broader range of text feature information more efficiently. This progress has effectively addressed the irregular distribution of text semantic features, leading to notable improvements in the results of NLP tasks. With the continuous development of related research on pre-training models, more efficient or lightweight pre-training models continue to emerge based on the Transformer (Vaswani et al. 2017), such as BERT (Devlin et al. 2019), GPT-3 (Brown et al. 2020), XLNet (Yang et al. 2019), RoBERTa (Liu et al. 2019), etc. Relevant researchers have provided Chinese versions of some pre-trained models to facilitate the development of related tasks in the field of Chinese NLP (Cui et al. 2021). These pre-training models necessitate extensive data, storage capacity, and costly hardware resources throughout the pre-training process to achieve positive outcomes. Consequently, pre-trained models in current text classification research are typically adapted to specific downstream tasks, involving fine-tuning and combining with other deep learning models for performance evaluation (Zhang 2023).

2.2 Graph convolutional network

The Graph Neural Network (GNN) model is a deep learning model that represents given data as nodes in a graph structure and then represents the relationship between nodes through the graph's edges (directed or undirected). By treating these nodes and edges as neurons and connections in the neural network, the GNN model performs feature extraction and learning in information aggregation and transfer.

The GCN was proposed by Kipf and Welling (2016), essentially a convolutional neural network that directly utilizes the structural information of graph-structured data. The convolution operation is performed in a grid structure with specific rules for traditional convolutional neural networks. However, for graph-structured data with irregular connections between nodes, the standard convolution operation loses its effect, so it needs to be redefined for the graph structure. In the GCN model, it is necessary to collect the feature information in each node and all their neighbor nodes and then achieve the purpose of convolution operation through operations such as aggregation and splicing.

In text classification tasks, the GCN model represents the text as an undirected graph, the word vectors in the text are represented as node feature information, and the edge connections between nodes represent the order in which these words appear in sentences(Pham et al. 2023). The GCN model can fully use the feature information relationships between nodes, thereby learning more complex text features and achieving better text classification task results. For Chinese text classification tasks, Jing et al. (2021) proposed an attention mechanism combined with a GCN method to classify Chinese geographical texts. In the ST-Text-GCN model (Cui et al. 2022), the author designed a self-training method to add keywords to the training set without introducing external knowledge. This method adds word confidence to the edge weight calculation of graph-structured data, which can effectively reduce classification errors caused by word ambiguity.

2.3 Attention mechanism

Initially introduced in the Computer Vision (CV) domain for extracting features from crucial image areas, the attention mechanism has since been established as a conventional Natural Language Processing (NLP) technique. The initial implementation of the attention mechanism in Neural Machine Translation (NMT) tasks was successfully improved and implemented by Bahdanau et al. (2014). Subsequently, the attention mechanism has been increasingly utilized in different studies in Natural Language Processing (NLP) and multimodal tasks, resulting in remarkable outcomes.

To improve the precision and accuracy of text classification models, the attention mechanism is utilized to assist in identifying and capturing semantic features effectively. Luong et al. (2015) expanded the attention mechanism based on the model in research (Bahdanau et al. 2014) and proposed two efficient and concise attention structures: global attention and local attention; Zhou et al. (2016) proposed a model that combines the Bi-LSTM network with the attention mechanism for semantic relationship classification tasks, enabling the extraction of important feature information from sentences; Yin et al. (2016) integrated the attention mechanism into the TextCNN (Kim 2014) to model sentence pairs in text, effectively representing the dependencies; From the perspective of mathematical explanation and cognitive intuition, Du et al. (2018) proposed a method that combines recurrent neural networks with CNN-based attention models, which effectively capture significant

features in sentences; In view of the high dimensionality and sparsity of text data and the complexity of natural language semantics, Liu and Guo (2019) introduced an AC-BiLSTM model that combines Bi-LSTM, attention mechanism, and convolutional layer. This model can capture both local features of text phrases and global sentence semantic features; Liang et al. (2021) developed a spatial view attention convolutional neural network known as SVA-CNN. By integrating multi-view representation learning, heterogeneous attention mechanisms, and convolution operations, this model can automatically extract and weigh multiple granular and fine-grained representations of text semantic features; Cheng et al. (2022) proposed a method that combines capsule network (Manoharan 2021) with hierarchical attention mechanism (Yang et al. 2016) for text classification. This approach focuses on refining the relationship between local features and global features of the text, obtaining detailed text feature information; Liu et al. (2022) presented a co-attention network model CNLE with label embedding, which enables the model to effectively mine the critical features by co-encoding texts and labels into mutually focused representations.

2.4 Modal fusion mechanism

In multi-modal tasks, effectively integrating processed features from multiple modalities is a crucial indicator for assessing the model's performance (Nagrani et al. 2021). Early machine learning techniques traditionally tended to process each modality's feature data independently, carrying out alignment and other processes on these features before combining them for multi-modal tasks. Due to the varying expression forms across different modalities and the distinct perspectives and approaches used in analyzing subjects, multimodal tasks inherently exhibit characteristics of both redundancy and complementarity. Hence, the essential factor in acquiring in-depth modal fusion feature information lies in the sensible and efficient integration of multi-modal feature information.

In text classification tasks, with the continuous development of related research on graph neural networks, some researchers consider traditional text feature representation and graph-structured text feature representation as single-modal features of text and conduct related research on multi-modal fusion. It can be known from the Literature (Lin et al. 2021) that the combination of the conductive method with GCN and BERT (Devlin et al. 2019) has been proposed for text classification. This approach aims to improve the representation capability of graph node features and facilitate the comprehensive learning of the dense relationship between text feature representation and graph node feature representation in texts. Based on the research (Lin et al. 2021), Dong et al. (2022) proposes to combine cross-entropy and hinge loss in the model training stage to effectively improve the ability of the model to learn contextual semantics and structural information in Chinese texts; Xu et al. (2023a) designed a model that integrates GCN and Bi-GRU networks to fully capture and learn complex semantic relationships and spatial feature information in texts.

The existing research on multi-modal fusion in text classification tasks often overlooks irrelevant information within both high and low-level semantic feature information. Motivated by the findings in the study (Dosovitskiy et al. 2021; Arevalo et al. 2020), we developed the Adaptive Modal Feature Information Fusion Module (AMFM). This innovative module proficiently merges two distinct modal features: the representation of text features and the representation of text's graph node features. It is designed to enhance the model's adaptive control over how these two modal features influence the ultimate categorization of

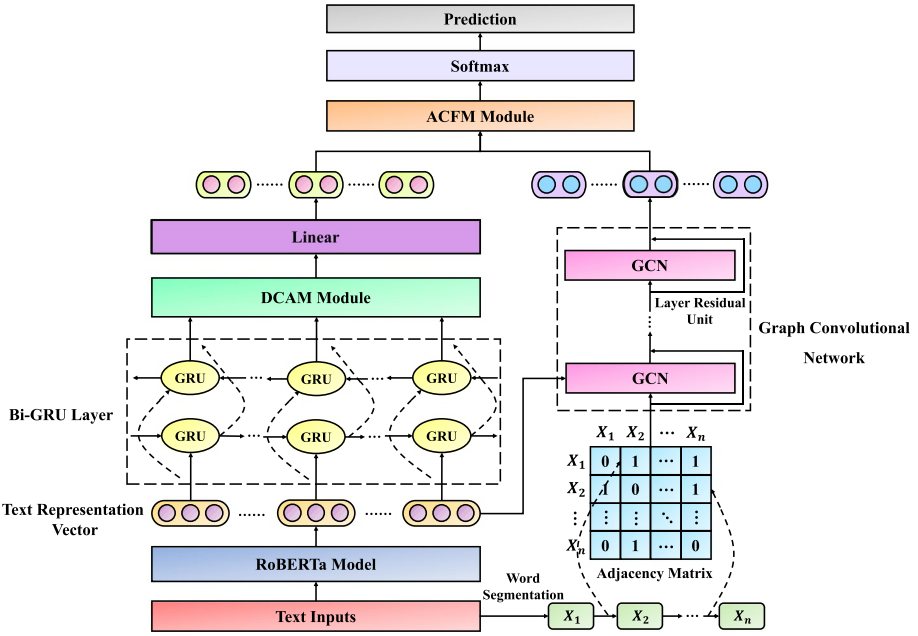

Fig. 2 The overall framework of the MACFM model proposed in this paper

texts. Additionally, this module plays a significant role in improving the overall accuracy and precision of the model when performing text classification tasks.

3 Problem formulation

For the given Chinese long text corpus $C = \{c_1, c_2, c_3, \cdots, c_m\}$, the corresponding text category labels are denoted as $L \in \mathbb{R}^{m \times n}$, where m represents the total number of texts in the corpus, and n denotes the total number of categories within the corpus C.

Our objective is first to embed the corpus C into word vectors and construct the corresponding text graph $G = (V, E)$, to acquire the word embedding features T and graph node features G. Here, V represents the set of word and document nodes contained within graph G, and E denotes the set of edges in graph G between words as well as between words and documents. Subsequently, by employing the MACFM model, we delve into the deep feature relationships of T and G, thereby obtaining refined feature representations $\bar{T}$ and $\bar{G}$. After that, we utilize adaptive context-aware multimodal feature fusion method to effectively and thoroughly merge these two modal feature representations, resulting in the fused feature f for training the model. Finally, we use the trained model M to predict the categories of unlabeled Chinese long texts, as shown in Eq. (1):

$$M(\bar{T}, \bar{G}) \rightarrow \hat{Y}, \tag{1}$$

where $\hat{Y}$ represents the model's prediction result of the text category.

4 Methods

This section will provide a detailed introduction to the MACFM model proposed in this paper. Figure 2 illustrates the overall framework of our model.

The MACFM model comprises three modules: text feature representation module, graph node feature representation module, modal feature fusion and class prediction module. Section 4.1 first introduces the text feature representation module of the model. In Sect. 4.2, the model's graph node feature representation module will be presented. Section 4.3 introduces the model's modal feature fusion and class prediction module.

4.1 Text feature representation

4.1.1 Word embedding layer

Inspired by the literature (Dai et al. 2021), to separate Chinese long texts and convert them into word vectors, we utilize RoBERTa in the word embedding layer. Extracting and representing text semantic features through RoBERTa can solve the problem of polysemy in Chinese words and obtain rich text semantic and grammatical feature information.

After the input text T goes through the input layer of RoBERTa, the sentence vector of the text is recorded as $S \in \mathbb{R}^{d_S}$. This vector represents the final hidden state of the [CLS] token in the text, where d_S is the dimension size of the sentence vector; The corresponding word vector of the i-th word in T is denoted as $W_i \in \mathbb{R}^{d_W}$, where d_W represents the dimension size of the word vector.

4.1.2 Bi-GRU layer

After receiving the text features generated by the word embedding layer, we utilize the Bi-GRU network to capture distant dependencies within the text. In this layer, the text representation vectors $T_1, T_2, \ldots, T_t$ outputted by the word embedding layer are first fed into the forward GRU network, to obtain the forward hidden layer's text semantic feature $\overrightarrow{h}_i$; Similarly, by inputting the text representation vectors $T_1, T_2, \cdots, T_t$ into the backward GRU network, we obtain the backward hidden layer's text semantic feature $\overleftarrow{h}_i$. At the i-th moment, we combine the forward output $\overrightarrow{h}_i$ and the backward output $\overleftarrow{h}_i$ to acquire the hidden layer's text semantic feature h_i, as indicated in Eq. (2):

$$h_i = \left[\overrightarrow{h}_i \oplus \overleftarrow{h}_i\right] \tag{2}$$

4.1.3 DCAM module

The structure of the Deep Contextual Feature Attention Module (DCAM) is shown in Fig. 3.

In the DCAM module, the critical features T in the context features captured by the Bi-GRU network are first learned and filtered through one-dimensional adaptive maximum

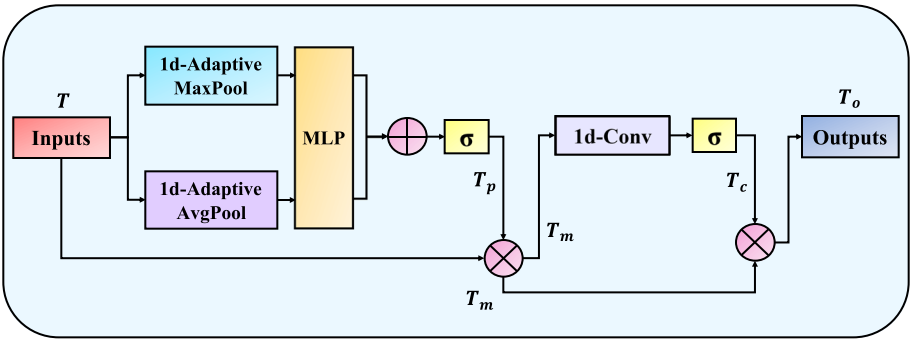

Fig. 3 The structure of the deep contextual feature attention module

pooling and one-dimensional adaptive average pooling, respectively. Then, the adaptive pooling attention feature T_p is obtained. The process is shown in Eq. (3):

$$\begin{aligned} T_p &= \sigma\left(\mathrm{MLP}\left(\mathrm{AvgPool}_1 D(T)\right) + \mathrm{MLP}\left(\mathrm{MaxPool}_1 D(T)\right)\right) \\ &= \sigma\left(W_1\left(W_0\left(T^p_{\mathrm{avg}}\right)\right) + W_1\left(W_0\left(T^p_{\mathrm{max}}\right)\right)\right), \end{aligned} \tag{3}$$

where $\sigma(\cdot)$ is the sigmoid activation function; MLP represents the multi-layer perceptron; W_0 and W_1 are weight matrices, used to linearly transform the input in MLP.

Multiply the text input feature T and the adaptive pooling attention feature T_p to obtain the weighted pooling attention feature T_m, as shown in Eq. (4):

$$T_m = T \times T_p \tag{4}$$

After performing a one-dimensional convolution operation on the weighted pooled attention feature T_m, the keyword semantic features in the context key features can be fully captured, and the convolutional attention feature T_c is obtained. The process is as shown in Eq. (5):

$$T_c = \sigma\left(f^{1\times 3}\left(T_m\right)\right), \tag{5}$$

where $\sigma(\cdot)$ is the sigmoid activation function; $f^{1\times 3}$ represents using a convolution kernel with a size of 1×3 to perform a 1d convolution operation.

After obtaining the convolutional attention feature T_c, multiply T_c by the weighted pooling attention feature T_m to obtain the output feature representation T_o, as shown in Eq. (6):

$$T_o = T_m \times T_c \tag{6}$$

The DCAM module can effectively extract deeper and fine-grained features from the Chinese long text context features extracted by the Bi-GRU network layer, thereby effectively capturing the keyword features and essential context feature information.

4.2 Graph node feature representation

4.2.1 Graph construction

This paper explores the creation of heterogeneous graphs using words and documents in text as nodes, focusing on the relationships between nodes in graph-structured data of texts.

We employ the TF-IDF technique for establishing the connection between words and documents, effectively measuring the significance of a particular word within a specific document. The frequency of a word appearing in a document directly corresponds to its importance for that document. In contrast, the frequency of a word occurring in all documents inversely affects its relevance to a particular document. With the TF-IDF approach, we can accurately capture the correlation between words and documents, leading to improved construction of graph-structured data.

To establish connections between words, we utilize the Positive Pointwise Mutual Information (PPMI) technique (Luo et al. 2020). The overall PMI (Pointwise Mutual Information) approach (Salle and Villavicencio 2022) quantifies the semantic correlation and resemblance between two words by evaluating the ratio of the likelihood that both words appear within the same context simultaneously to the likelihood that the two words appear in different contexts. Equation (7) displays the computation process:

$$\mathrm{PMI}(x, y) = \log_2 \frac{p(x, y)}{p(x)p(y)}, \tag{7}$$

where $p(x, y)$ represents the probability that vocabulary x and vocabulary y appear in a context window at the same time; $p(x)$ and $p(y)$ present the probability of occurrence of the context window containing x and y, respectively.

The problem with the PMI method is that if two words never appear together or alone in the same context, the value of PMI will be 0 or a negative value, where a negative value means no correlation between the two words. This does not mean that the two words are unrelated in actual situations. The PPMI method only considers the situation where two words appear in the same context and can reassign the PMI value to 0 when it is negative, thereby effectively avoiding the problems and defects in the PMI method. Equation (8) illustrates the procedure for calculating the PPMI method:

$$\mathrm{PPMI}(x, y) = \max\left\{\log_2 \frac{p(x, y)}{p(x)p(y)}, 0\right\} \tag{8}$$

The definition of the edge weight relationship between any two nodes i and j in the graph is as shown in Eq. (9):

$$A_{i,j} = \begin{cases} \mathrm{PPMI}(i, j), & i, j \text{ are words and } i \neq j \\ \text{TF-IDF}(i, j), & i \text{ is document, } j \text{ is word} \\ 1, & i = j \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

4.2.2 Graph convolutional network

In the graph $G = (V, E)$, V denotes the collection of nodes in the graph G, while E represents the collection of edges in the graph G. In GCN, the aggregation is accomplished by applying convolution operations to the feature data of neighboring nodes belonging to any given node in the graph G. This process enables the node's characteristics to be updated while effectively capturing the structural information contained within the graph and preserving the translation invariance of neighbor nodes' features. As a result, all nodes can extensively learn the nodes' high-level attributes and the interdependency between each node and its higher-order neighboring nodes.

In constructing graphs for Chinese long texts, we utilize the word vector feature representations derived from RoBERTa as the rows of the initial feature matrix for the graph. The number of rows in the feature matrix matches the number of nodes in the graph adjacency matrix, and the number of columns in the matrix corresponds to the dimension of the word vector obtained from RoBERTa.

Employing the graph construction approach, the initial feature matrix X may incorporate abundant features related to location information and text semantics. Concurrently, it can enhance the model's generalizability to a certain degree. Regarding the initial feature matrix $X \in R^{n \times d}$, n denotes the count of nodes in the adjacency matrix, d signifies the graph feature embedding dimension for each node.

The information transfer between layers in GCN is shown in Eq. (10):

$$H^{(l+1)} = \rho\left(D^{-\frac{1}{2}}\tilde{A}D^{-\frac{1}{2}}H^{(l)}W^{(l)}\right), \tag{10}$$

where $H^{(l)}$ represents the node feature matrix of the l-th layer, when $l = 0$, $H^{(1)} = X$, which is the initial feature matrix; The weight matrix of the l-th layer is denoted as $W^{(l)} \in R^{d \times n}$, which is used to linearly transform the node features; $\rho(\cdot)$ represents the nonlinear activation function; $\tilde{A} \in R^{n \times n}$ represents the graph adjacency matrix with self-connection, $\tilde{A} = A + I$, A represents the graph adjacency matrix, I is the identity matrix; D is the symmetric matrix of $\tilde{A}$.

4.2.3 Layer residual unit

Currently, the number of layers in GCN generally does not exceed three. The reason is that deepening the number of layers in GCN may cause the model to suffer from the gradient disappearance problem and the deep degradation problem of over-smoothing node information. The shallow GCN cannot thoroughly learn the high-level semantic information in high-order neighbor nodes, so the model's performance will be subject to certain limitations.

To address the above problems, we introduce an effective layer residual unit, and its calculation process is shown in Eq. (11):

$$H^{(l+1)} = \rho\left(D^{-\frac{1}{2}}\tilde{A}D^{-\frac{1}{2}}\left(H^{(l)} + X\right)W^{(l)}\right) \tag{11}$$

By incorporating the output initial feature matrix X from the previous layer of GCN, the layer residual unit ensures that the model retains sufficient information. This significantly

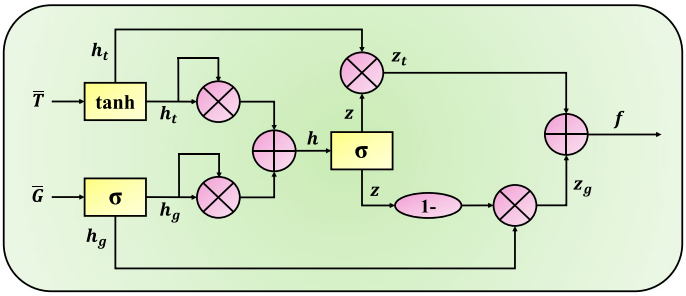

Fig. 4 The structure of the adaptive modal feature information fusion module

improves the GCN's results and performance when the number of convolution layers is increased.

4.3 Modal feature fusion and category prediction

To obtain more effective modal fusion features for making final predictions, it is necessary to filter further the text feature representation $\bar{T}$ and graph node feature representation $\bar{G}$, because these feature representations may still contain noise information. We have developed the Adaptive Modal Feature Information Fusion Module (AMFM) to address this issue, as depicted in Fig. 4, drawing inspiration from previous research (Dosovitskiy et al. 2021; Arevalo et al. 2020). This module improves the capability of the feature subspace to represent semantics, allowing for the deep adaptive fusion of feature representations from diverse modalities. By controlling the contribution ratio of the two modalities to the output feature information, the AMFM effectively filters the multi-modal fusion mechanism to remove irrelevant and redundant information. Consequently, the model can learn and comprehend more high-level semantic fusion features.

The AMFM module performs adaptive feature fusion calculation on text feature representation $\bar{T}$ and graph node feature representation $\bar{G}$ to obtain the fusion feature representation f. The calculation process is as follows:

$$h_t = \tanh(\bar{T}) \tag{12}$$

$$h_g = \text{sigmoid}(\bar{G}) \tag{13}$$

$$h = \left(h_t \times h_t\right) + \left(h_g \times h_g\right) \tag{14}$$

$$z = \text{sigmoid}(h) \tag{15}$$

$$z_t = z \times h_t \tag{16}$$

$$z_g = (1 - z) \times h_g \tag{17}$$

$$f = z_t + z_g \tag{18}$$

In the above calculation process, the initial step involves performing deep-level and fine-grained information filtering on the text feature representation $\bar{T}$, the graph node feature representation $\bar{G}$, and the weighted sum feature h of them through Eqs. (12)–(14). Through Eqs. (15)–(17), the weight z used to dynamically adjust the contribution of h_t and h_g, text adaptive features z_t, and image adaptive features z_g can be obtained so that the inflow of feature information can be adaptively controlled, thereby achieving full integration of text feature representation and graph node feature representation.

Once the AMFM module generates the modal feature fusion representation f, it is fed into the fully connected layer. This step maps the feature vector to the sample label space through feature space transformation.

Finally, the prediction of the text category is determined using the Softmax function, which is illustrated in Eq. (19):

$$P(Text = N \mid f) = \frac{\exp\left(W_N^T f + b_N\right)}{\sum_{n=1}^{N} \exp\left(W_n^T f + b_n\right)}, \tag{19}$$

where W_N^T, W_n^T are the weight matrices, b_N, b_n represent the bias vector, N represents the total number of text category labels.

5 Experiments

5.1 Datasets

Iflytek[1]: Consists of 119 text categories and 30,812 data. The average text length is 533 characters. The dataset is further separated into a training set with 24,649 data, a validation set with 3081 data, and a test set with 3082 data.
INews[1]: Consists of 3 text categories and 7355 data. The average text length is 1251 characters. The dataset is further separated into a training set with 5355 data, a validation set with 1000 data, and a test set with 1000 data.
THUCNews[1]: Consists of 14 text categories and 41,796 data. The average text length is 921 characters. The dataset is further separated into a training set with 33,436 data, a validation set with 4180 data, and a test set with 4180 data.
SogouCS[2]: Released in 2013 by Sogou Lab. This experiment used a subset of the original dataset due to its large size, and categories with small samples were excluded. This dataset consists of 9 text categories and 92,103 data. The average text length is 621 characters. The dataset is further separated into a training set with 73,677 data, a validation set with 9211 data, and a test set with 9215 data.
Fudan[3]: Consists of 20 text categories and 19,636 data. The average text length is 8334 characters. The dataset is further separated into a training set with 9804 data and a test set with 9832 data.

[1] https://github.com/ChineseGLUE/ChineseGLUE.

[2] http://www.sogou.com/labs/resource/cs.php.

[3] http://www.nlpir.org/wordpress/download/tc-corpus-answer.rar.

Table 1 General hyperparameter settings for Chinese long text datasets

Dataset	Input sequence length	Batch size	Epoch	BERT Learning rate	GCN learning rate
Iflytek	128	64	50	2e−5	1e−3
INews	128	64	50	2e−5	1e−3
THUCNews	128	32	50	2e−5	1e−3
SogouCS	128	32	50	2e−5	1e−3
Fudan	128	32	50	2e−5	1e−3

Table 2 Graph convolutional network and DCAM module hyperparameter settings

Module	Parameter	Value
Graph convolutional network	Convolutional layers	5
	Hidden layer dimensions	256
	Dropout	0.1
DCAM module	Convolution kernel size	3

5.1.1 Data preprocess

Text preprocessing is required since both the SogouCS and Fudan datasets are original Chinese corpora containing many meaningless sentences and symbols. First, we use regular expressions to filter noise information in the dataset, such as particular characters, punctuation marks, and emoticons. Then, we utilize the Chinese stop words list compiled by the Harbin Institute of Technology to remove stop words that appear frequently but have no practical meaning in the dataset, such as modal particles and structural particles.

After the above steps, punctuation marks were removed from the remaining three preprocessed Chinese long text datasets to minimize the impact of irrelevant information on the construction process of graph-structured data.

5.1.2 Graph construction details

While creating the text graph, the window size is defined as 20, the word frequency threshold is defined as 15, and the dimension size of the initial feature matrix is set to 768, which matches the dimension of the word vectors obtained from RoBERTa. For more information about the window size and word frequency threshold experiments, please refer to Sect. 5.4.2.

By implementing a word frequency threshold to filter sparse words and uncommon words, the size of the text dictionary can be effectively controlled, thereby optimizing the size of the graph structure, reducing computational complexity, saving storage resources, and retaining important feature information.

5.2 Implementation details

According to the average text length of the dataset and the maximum length limit of RoBERTa for processing texts, we set Chinese texts with a text length of more than 512 words as the distinguishing conditions between long text and short text.

The experiments conducted in this article utilize the RoBERTa-zh-base[4] pre-training model. All experiments are carried out on the RTX3090 graphics card using the PyTorch deep learning framework. Considering the recommendations mentioned in the study (Lin et al. 2021), athe experimental environment, hardware configuration, and the relevant ablation experiments in Sect. 5.4.2, the general hyperparameter settings for the five long text datasets are shown in Table 1.

In Table 1, the input sequence length of the five Chinese long text datasets is uniformly set to 128 due to hardware constraints. Due to the large amount of data and long text length in the THUCNews, SogouCS, and Fudan datasets, the training batch size is set to 32. Furthermore, this article uses the Adam optimizer and cross-entropy loss function.

Considering the suggestions from related research (Lin et al. 2021) and the experimental study on model parameter ablation in Sect. 5.4.1, the hyperparameter settings in the GCN and DCAM module are shown in Table 2.

5.3 Performance comparison

In this section, the Chinese long text classification model MACFM proposed in this paper is compared with SOTA models on Iflytek, INews, THUCNews, SogouCS, and Fudan datasets to verify the performance advantages of the MACFM model in Chinese long text classification tasks.

Section 5.3.3 uses the MACFM model to conduct generalization experiments on five English datasets to prove its scalability and interpretability further.

5.3.1 Comparison models

(1) **RoBERTa** (Liu et al. 2019): A BERT variant model using different pre-training methods, which is more robust and optimized than the BERT model.
(2) **TextGCN** (Yao et al. 2019): The GCN model performs an aggregation operation on the feature information of all nodes and their neighbor nodes in the adjacency matrix of the text graph.
(3) **GAT** (Velikovi et al. 2018): The GAT model uses a multi-head attention mechanism to assign corresponding weights to all nodes in the adjacency matrix, and the information aggregation result of the nodes is used as the weighted sum of attention weights.
(4) **SGC** (Zhu and Koniusz 2020): The SGC model reduces unnecessary complexity and redundant calculations in the GCN model by continuously eliminating nonlinearity and folding the weight matrix between consecutive layers.
(5) **HyperGAT** (Ding et al. 2020): The HyperGAT model defines document-level hypergraphs and designs hypergraph attention networks for graph nodes and edges, respectively.

[4] https://github.com/brightmart/roberta_zh.

Table 3 Experimental results comparing the performance of the MACFM model on Chinese datasets

Model	Dataset									
	Iflytek		INews		THUCNews		SogouCS		Fudan	
	Acc	F_1	Acc	F_1	Acc	F_1	Acc	F_1	Acc	F_1
RoBERTa	63.82	46.54	84.80	79.96	95.57	94.83	92.18	84.49	94.61	62.14
TextGCN	66.15	41.55	87.44	80.72	96.38	95.59	94.64	86.37	96.77	80.85
GAT	65.28	36.27	86.92	79.55	96.19	95.94	94.49	85.75	96.00	75.24
SGC	66.34	39.65	87.81	80.77	95.62	94.59	95.26	88.87	96.72	80.19
HyperGAT	65.80	38.07	87.33	81.56	95.67	94.74	95.27	87.48	96.41	78.93
RoBERTaGCN	68.44	47.64	89.03	84.15	97.38	96.21	95.66	91.35	97.71	88.51
RoBERTaGAT	67.28	43.53	88.29	82.41	97.28	96.18	95.22	88.91	97.28	84.79
GFN	66.76	43.99	87.58	81.51	96.51	95.15	95.91	90.42	96.94	84.12
GRTE	68.35	48.15	88.17	82.96	97.69	96.95	96.09	91.77	97.53	89.75
DCAT-RBG	69.13	50.92	**90.20**	**85.77**	97.06	95.99	96.75	92.59	97.77	90.60
ContGCN-RB	67.58	47.47	89.59	84.68	97.83	97.25	96.47	92.04	97.16	89.47
MACFM-GDAL (Ours)	**70.73**	**52.50**	89.87	85.56	**98.22**	**97.66**	**96.89**	**92.77**	**98.13**	**91.22**

The bold values represent the best accuracy and F1 score of our proposed MACFM model and other comparative models on five Chinese datasets. In each column of the table, we bold the highest value so that readers can more intuitively see the performance comparison of our proposed model with the comparative models on different datasets

(6) **RoBERTaGCN** (Lin et al. 2021): The variant model of BertGCN(Lin et al. 2021) model, replace BERT with RoBERTa to combine GCN.
(7) **RoBERTaGAT** (Lin et al. 2021): Similar to RoBERTaGCN, except GCN is replaced with GAT.
(8) **GFN** (Dai et al. 2022): The GFN model supports efficient reasoning of documents and can better capture structural information by integrating different views of the text graph.
(9) **GRTE** (Aras et al. 2024): Combines GNN with large-scale pre-trained models, which can retrieve global and contextual information in documents and generate word embeddings for inductive reasoning.
(10) **DCAT-RBG** (Dong et al. 2023): Capturing the logical semantics of text through transductive learning and graph structures, then using a dual-channel attention network to achieve complementarity between semantics to improve understanding deficits. "RBG" presents using the RoBERTaGCN model.
(11) **ContGCN-RB** (Wu et al. 2023): Combines an occurrence memory module with a self-supervised contrastive learning objective to update the model dynamically. "RB" presents using the RoBERTa model.

5.3.2 The results of performance comparison with SOTA methods on Chinese datasets

The experimental results of the performance comparison between the MACFM model and the above text classification SOTA models on five Chinese long text datasets are shown in Table 3.

Table 4 Experimental results comparing the performance of the MACFM model on English datasets

Model	Dataset				
	20ng	R8	R52	Ohsumed	MR
CNN-rand	77.54	94.59	85.84	44.87	75.68
CNN-non-static	82.67	96.23	88.07	59.39	78.47
LSTM	67.23	94.50	86.67	42.31	75.50
Bi-LSTM	75.03	96.64	91.45	50.34	78.54
fastText	79.68	96.34	92.90	58.19	75.34
fastText(bigrams)	79.96	94.85	91.04	56.08	76.36
SWEM	85.45	95.58	93.18	63.67	77.28
LEAM	82.15	93.55	92.07	59.37	77.40
RoBERTa	83.80	97.80	96.20	70.70	89.40
TextGCN	86.30	97.10	93.60	68.40	76.70
RoBERTaGCN	89.50	98.20	96.10	72.80	89.70
RoBERTaGAT	86.50	98.00	96.10	71.20	89.20
SGC	88.60	97.40	94.20	68.80	76.20
HyperGAT	86.78	98.20	95.25	70.24	78.59
GFN	87.01	98.22	95.31	70.20	78.04
GRTE	**90.64**	98.72	96.25	71.70	89.73
DCAT-RBG	89.30	**98.80**	97.00	**74.90**	90.20
ContGCN-RB	90.10	98.60	97.00	73.40	91.30
MACFM-GDAL (Ours)	**90.64**	98.73	**97.38**	74.65	**92.28**

The bold values represent the best accuracy of the MACFM model proposed by us and other comparative models in experiments on five English datasets. In the numerical values of each column in the table, we bold the highest value so that readers can more intuitively see the performance comparison of our proposed model with the comparative models on different datasets

From the experimental results in Table 3, the Chinese long text classification model MACFM proposed in this paper can achieve the best classification results on four Chinese long text classification datasets except for the INews dataset.

The RoBERTaGCN learns text features by combining GCN with RoBERTa, but a manual threshold setting is required to adjust the ratio of the two features. GRTE model can effectively retrieve global and contextual information in text and generate word embeddings for inductive reasoning, improving the model's reasoning ability for text semantics. DCAT-RBG model realizes the semantic complementarity between RoBERTa and GCN through a dual-channel attention network to enhance the model's understanding of text semantic features. GFN integrates different views of text graphs to better capture structural information. ContGCN-RB utilizes general semantic knowledge in the large Wikipedia corpus by combining event memory modules with self-supervised contrastive learning objectives.

Compared with the above models, the MACFM model can more effectively focus on text semantic features of complex Chinese long texts by introducing the DCAM and AMFM modules. In addition, the MACFM model can also adaptively fuse text features at a fine-grained level to improve text classification performance. The advantage of the MACFM model lies in learning and understanding more complex long text features, and its

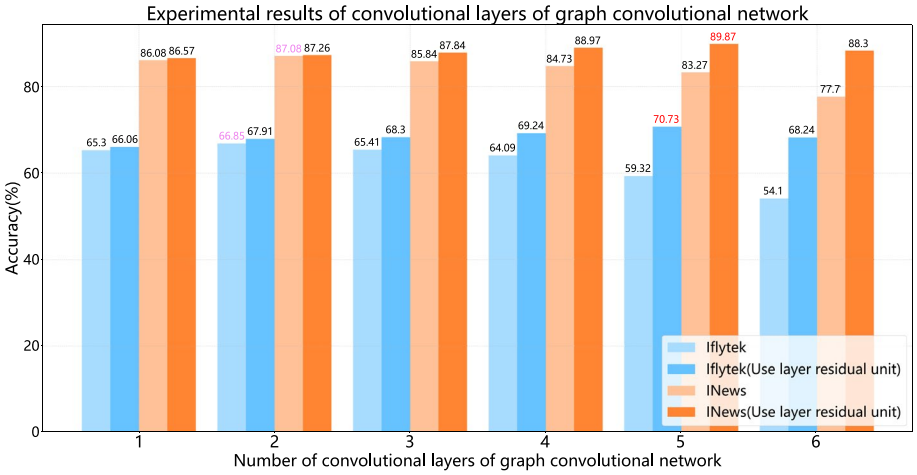

Fig. 5 Experimental results of convolutional layers of graph convolution network

performance advantage on the Ifytek dataset is relatively significant. In addition to positive and negative category labels, the INews dataset also has a category of neutral labels. We speculate that MACFM performs slightly worse than DCAT-RBG on this dataset because the dual-channel attention mechanism can more accurately understand the semantic features of texts in neutral categories and reduce the probability of misclassifying texts as positive or negative.

Overall, the MACFM model can significantly improve the accuracy indicators of the four Chinese datasets and show competitive results on the INews dataset. It indicates that the multi-modal fine-grained fusion strategy of Chinese long text features and the adaptive modal feature fusion mechanism proposed in this article are effective.

5.3.3 The results of generalization experiments on English datasets

To prove the generalization of the MACFM model proposed in this paper, the experimental performance comparison results of test accuracy with SOTA models on five English datasets are shown in Table 4.

SWEM, fastText, and LEAM are word embedding models based on pretraining, which can initially capture potential semantic features in text; CNN, LSTM, and Bi-LSTM are sequence-based methods capable of capturing text features from local continuous word sequences. For the rest of the comparison models, please refer to Sect. 5.3.1.

According to the findings in Table 4, the final row displays the test results of the MACFM model introduced in this article. The experimental results indicate that compared to other advanced models for text classification, the MACFM model achieves the highest accuracy on three English datasets: 20ng, R52, and MR. The results on the 20ng dataset are tied with the GRTE model. Although MACFM did not achieve the best performance results on the R8 and Ohsumed datasets, the gap with DCAT-RBG was small.

The competitive results of MACFM provide evidence for the effectiveness of the multi-modal fine-grained fusion strategy and the adaptive control feature fusion mechanism in English texts. In addition, the generalization and interpretability of the MACFM model have been further demonstrated.

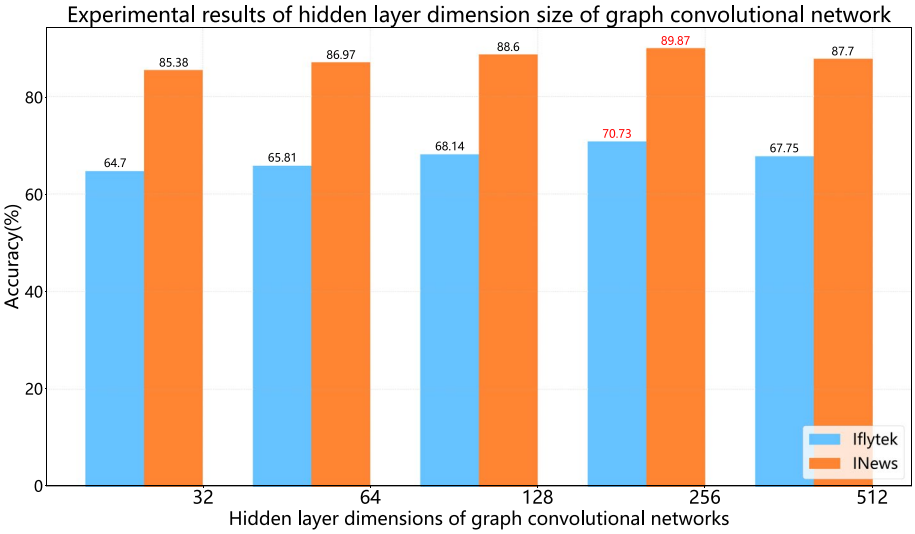

Fig. 6 Experimental results of hidden layer dimension size of graph convolutional network

Table 5 Experimental results of convolution kernel size parameters of DCAM module

Dataset	Convolution kernel size	Accuracy (%)	Precision (%)	Recall (%)	(%)
Iflytek	1	67.53	63.63	39.92	42.93
	2	69.40	65.54	42.99	47.23
	3	**70.73**	**74.60**	**48.14**	**52.50**
	4	68.11	65.96	41.90	43.03
	5	66.45	63.50	38.77	42.39
INews	1	87.78	84.51	79.57	81.42
	2	88.26	85.30	80.05	82.05
	3	**89.87**	**88.04**	**83.76**	**85.56**
	4	87.94	84.63	80.88	82.40
	5	86.75	84.33	76.23	78.70

The bold values represent the optimal convolution kernel size and its performance obtained by the DCAM module in our proposed MACFM model on two Chinese datasets. In each column of the table, we bold the highest value so that readers can more intuitively see the performance comparison of the DCAM module using different convolution kernel sizes on the two datasets

5.4 Ablation studies

In this section, we mainly explore the GCN and DCAM module in the MACFM model and the relevant parameter selection of graph construction. Additionally, through ablation experiments of MACFM variant models, the effectiveness and interpretability of our model are proved.

Table 6 Experimental results of word frequency threshold parameters

Dataset	Word frequency	Accuracy (%)	Precision (%)	Recall (%)	F_1 (%)	Dictionary size	Graph nodes
Iflytek	None	68.61	66.55	41.76	42.54	92009	122821
	5	59.58	29.51	27.91	26.88	29152	59964
	10	67.37	65.34	39.37	42.94	18050	48862
	1 5	**70.73**	**74.60**	**48.14**	**52.50**	13653	44465
	20	68.76	61.29	41.29	44.71	**11162**	**41974**
INews	None	88.60	84.86	82.63	83.64	168327	175682
	5	81.10	83.57	60.36	58.28	46659	54014
	10	87.38	81.97	82.94	82.41	27665	35020
	15	**89.87**	**88.04**	**83.76**	**85.56**	19837	27192
	20	88.02	83.57	82.23	82.84	**15621**	**22976**

The bold values indicate the best word frequency threshold parameters and their performance when our proposed MACFM model constructs graph structure data on two different Chinese datasets. In each column of the table, we bold the highest value so that readers can more intuitively see the performance comparison of the MACFM model using different word frequency threshold parameters when constructing graph structure data on two datasets

5.4.1 MACFM parameter ablation

This section mainly discusses the selection of relevant parameters of the MACFM model (using all modules).

(1) Considering that the hidden layer dimension size of GCN is limited to 256, the selection of the convolutional layers in GCN (with layer residual units) is explored on the Iflytek and INews datasets, respectively. The results are shown in Fig. 5.

Based on the results of the experiment shown in Fig. 5, it is observed that the model performs the best when the GCN has 2 convolutional layers without the layer residual unit. Increasing the convolutional layers leads to a decline in the model's performance, as the deep graph convolution network struggles to learn the node characteristics of the graph effectively. However, when the residual unit of the graph convolution network layer is added, the model's overall performance remains stable even with more than 2 convolutional layers in the GCN. Specifically, the model performs the best with 5 convolutional layers. Therefore, the number of convolutional layers in the GCN is set to 5.

(2) Under the condition that the number of convolutional layers in GCN is limited to 5, the selection of hidden layer dimension size in GCN is explored on the Iflytek and INews datasets, respectively. The results are shown in Fig. 6.

Based on the results in Fig. 6, when the number of convolutional layers is set to 5, the model performance shows an increasing trend before the hidden layer dimension size increases to 256. Consequently, GCN's hidden layer dimension size is set to 256.

(3) When the relevant parameters and modules in GCN are fixed, the selection of convolution kernel size in the DCAM module is explored on the Iflytek and INews datasets, respectively. The results are shown in Table 5.

From the experimental results in Table 5, with the relevant parameters and modules in GCN fixed, when the convolution kernel size in the DCAM module is set to 3, the MACFM model can achieve the best results on both datasets simultaneously. Therefore, the convolution kernel size is set to 3.

Table 7 Experimental results of window size parameters

Dataset	Window size	Accuracy (%)	Precision (%)	Recall (%)	F_1 (%)
Iflytek	10	67.03	48.53	41.14	42.05
	2 0	**70.73**	**74.60**	**48.14**	**52.50**
	30	67.79	50.55	43.64	44.82
	40	64.36	46.71	38.39	39.98
INews	10	87.67	85.38	78.41	80.88
	2 0	**89.87**	**88.04**	**83.76**	**85.56**
	30	87.28	83.06	80.27	81.49
	40	86.49	81.88	78.02	79.56

The bold values indicate the best window size parameters and their performance when our proposed MACFM model constructs graph structure data on two different Chinese datasets. In the numerical values of each column in the table, we bold the highest value so that readers can more intuitively see the performance comparison of the MACFM model using different window size parameters when constructing graph structure data on two datasets

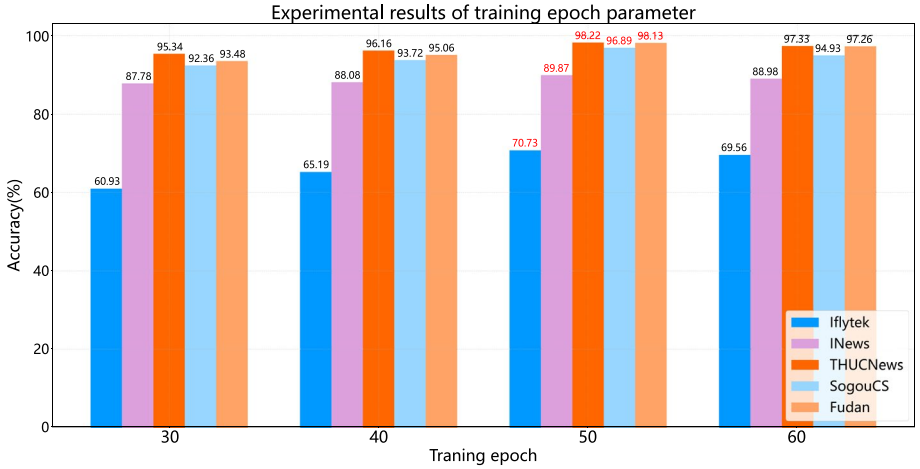

Fig. 7 Experimental results of training epoch parameter

5.4.2 Dataset parameter ablation

This section mainly uses the MACFM model to conduct experiments on the relevant parameters of graph construction and the training epoch.

(1) Set the window size to 20 and explore the word frequency threshold parameters on Iflytek and INews datasets. The experimental results are shown in Table 6.

Based on the results shown in Table 6, the graph's size can be effectively reduced by increasing the word frequency thresholds when building graphs for Chinese long text datasets. This reduction results in fewer unnecessary nodes and edges in the graph, ultimately optimizing the graph's size and reducing computational complexity. A significant decrease in the model's performance occurs when the word frequency threshold

Table 8 Experimental results of MACFM variant model on Iflytek and INews datasets

Dataset	Model	Accuracy(%)	Precision(%)	Recall(%)	F_1(%)
Iflytek	TextGCN	66.15	65.77	38.07	41.55
	MACFM-GL	66.82	63.58	39.75	43.05
	MACFM-DL	67.82	65.03	40.17	43.59
	MACFM-AL	68.33	71.03	43.80	48.06
	MACFM-GDL	69.11	67.72	43.16	47.07
	MACFM-GAL	70.12	66.44	45.14	45.39
	MACFM-GDAL	**70.73**	**74.60**	**48.14**	**52.50**
INews	TextGCN	86.50	85.07	75.08	77.82
	MACFM-GL	87.44	83.43	79.48	81.10
	MACFM-DL	87.80	85.31	79.11	81.40
	MACFM-AL	88.15	84.39	80.57	82.11
	MACFM-GDL	88.20	84.47	81.59	82.81
	MACFM-GAL	89.31	86.34	**84.19**	85.13
	MACFM-GDAL	**89.87**	**88.04**	83.76	**85.56**

The bold values indicate the best performance of our proposed MACFM model when using different module combinations on two different Chinese datasets. In the numerical values of each column in the table, we bold the highest value so that readers can more intuitively see the performance comparison of different module combination variants of the MACFM model on the two datasets

is set to 5. The probable cause is the heightened presence of noise information contained in the dictionary under this threshold. As the threshold increases, the model's performance gradually improves. The MACFM model achieves the best outcomes on the Iflytek and INews datasets when the word frequency threshold is set to 15. Consequently, the word frequency threshold for graph construction on the datasets is set to 15.

(2) Set the word frequency threshold to 15 and explore the window size parameters on Iflytek and INews datasets. The experimental results are shown in Table 7.

Based on the results obtained from Table 7, it can be observed that the MACFM model's classification performance on Chinese long text datasets is dependent on the window size during graph construction. Specifically, when the window size is set to 15, the MACFM model demonstrates optimal performance on the Iflytek and INews datasets. Therefore, the window size is set to 15.

(3) The training epochs of the MACFM model are explored separately on five Chinese long text datasets. The experimental results are shown in Fig. 7.

From the experimental results in Fig. 7, when the training epoch is 50, the MACFM model can achieve the best results on five Chinese long text datasets. When the training epoch is 60, the model performance begins to decline, and the reason may be that the model training time is too long, resulting in overfitting. Therefore, the training epoch is set to 50.

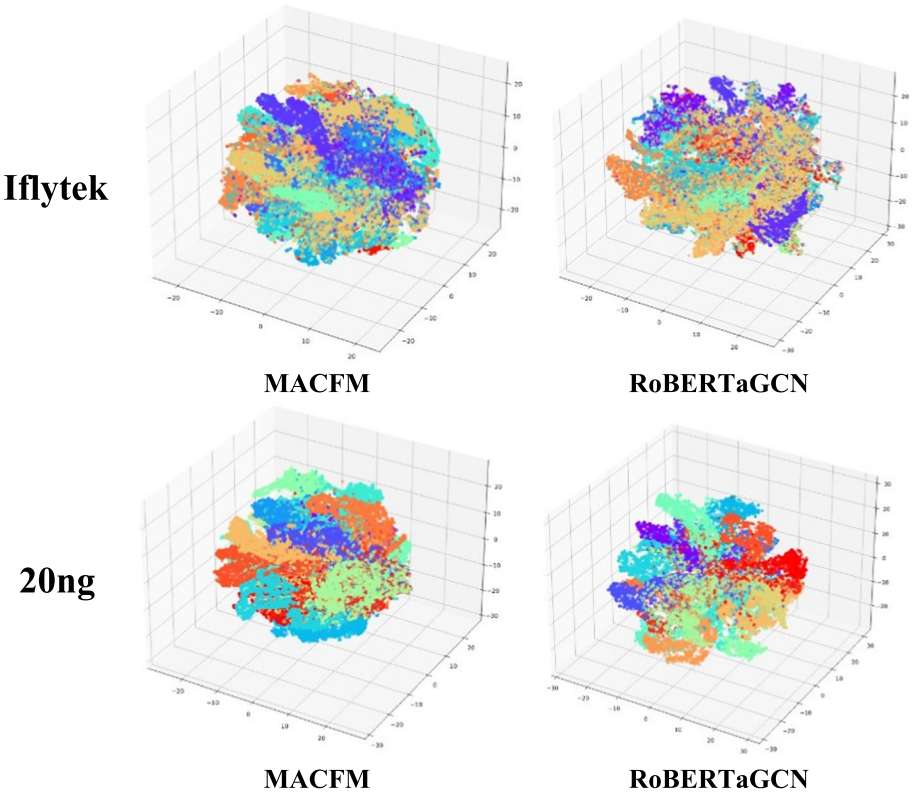

Fig. 8 The t-SNE visualization results

5.4.3 MACFM variants

Table 8 displays the findings from the performance comparison of various MAGM model variants on Iflytek and INews datasets.

The MACFM model uses the Bi-GRU network, DCAM module, AMFM module, and layer residual unit to fully mine, understand, and fuse the complex and unevenly distributed feature semantic information in Chinese long texts. Table 8 sets the convolutional layers in GCN to 5, and the layer residual unit is used in GCN, represented with 'L'. 'GL' represents using the combination of the Bi-GRU network and layer residual unit, 'DL' means using the combination of the DCAM module and layer residual unit. 'AL', 'GDL', 'GAL' and 'GDAL' are all analogous.

Based on the findings presented in Table 8, when the MACFM model utilizes the Bi-GRU network, DCAM module, and AMFM module, it demonstrates significant performance enhancements on the Iflytek and INews datasets compared to the baseline model.

The experiments in Table 8 prove that the Bi-GRU network can effectively learn long-distance dependencies in Chinese long texts, capture and extract contextual feature information, and provide fine-grained text feature semantic information for the DCAM and AMFM modules. The DCAM module can effectively focus on and capture

the contextual feature information in Chinese long texts. At the same time, it can filter irrelevant information to obtain practical high-level text semantic features. The AMFM module can effectively perform a deep adaptive fusion of text and graph node feature representation and fully filter irrelevant information in the modal fusion process.

5.5 Visualization

To provide additional evidence of the MACFM model's superior performance and interpretability in long text classification tasks, we employ the t-SNE technique (Van der Maaten and Hinton 2008) to compare the model's results visually, as shown in Fig. 8.

Figure 8 illustrates the visualization results of the MACFM model and the baseline comparison model RoBERTaGCN on the test set of the Chinese dataset Iflytek and the English dataset 20ng. The colors of the nodes in the figure signify the category labels of the datasets.

Compared with the RoBERTaGCN model, the MACFM model optimizes the graph structure, deeply mines text semantics through the DCAM module, and adaptively fuses different text modal features through the AMFM module. These improvements effectively solve the problem of the RoBERTaGCN model's insufficient understanding of text semantics and the need to set the fusion scale parameters manually.

The visualization results in Fig. 8 show that the MACFM model can learn more differentiated text semantic representations, making the distance between data of the same category closer, and the overall data distribution shows a more aggregated trend. This reveals that our model has more robust representation capabilities, leading to better performance in long text classification.

6 Discussion

Distinct from current research on Chinese text classification methods that focus on single text features and shallow fusion of text features, we emphasize investigating effective ways to fuse different textual feature representations inspired by multimodal task-related research. This approach enables the model to deeply understand the contextual feature information within complex and long Chinese texts. Our proposed MACFM model can effectively fuse fine-grained text and graph node features from Chinese long texts. As evidenced by the experimental results in Sect. 5, the MACFM model achieves satisfactory classification performance for Chinese long texts and demonstrates good generalizability and interpretability on English datasets.

Indeed, our current research has certain limitations. The performance of the MACFM model on the INews, R8, and Ohsumed datasets is slightly weaker than the DCAT-RBG model, which provides us with an excellent perspective and ideas for future long text classification research. For instance, before fusing the word embedding and graph node features, a more in-depth study of inter-modal interaction learning could further enhance the model's efficiency in learning and understanding textual language. We can deeply explore the advancement of the dual-channel mechanism of the DCAT-RBG model. At the same time, we can further explore complementary methods between text semantic features to continue improving the model's ability to understand

the semantics of long texts. Additionally, subsequent work could explore the adaptive feature fusion method in greater depth, such as integrating optimization techniques from operations research, like branch and bound, for adaptive fusion parameter learning. This could better integrate different modal text features and enhance the semantic representation capability of the fused features.

7 Conclusion

This study aims to enhance the performance of Chinese long text classification tasks by introducing several improvements. First, we present a deep contextual feature attention module to improve the model's ability to capture important contextual and keyword features. Next, we optimize the size of the graph structure by introducing text word vector features and layer residual units to enhance the semantic feature mining capability of the Graph Convolutional Network. Additionally, an adaptive modal feature information fusion mechanism is integrated into the multi-modal fusion process to improve the model's overall feature fusion ability. Experiments conducted on Chinese and English datasets validate the interpretability and effectiveness of the proposed MACFM model. The results demonstrate that the MACFM model successfully addresses the challenges posed by uneven semantic distribution and complex semantic information in Chinese long texts, as well as the issue of the Graph Convolutional Network overlooking contextual information.

Enhancing the model's ability to capture fine-grained semantic features in text questions is a significant obstacle for specific multi-modal tasks like Visual Question Answering (VQA). This is crucial for improving the model's learning capabilities in intra- and inter-modal interactions. Hence, we believe that the findings presented in this paper can be applied to multi-modal tasks. In future work, we will investigate how to effectively integrate the proposed methods of extracting feature information and filtering noise information into multi-modal tasks better to establish the relationship between features of different modalities.

Author contributions Yangshuyi Xu: Conceptualization, methodology, writing—original draft. Guangzhong Liu: Supervision, review and editing. Lin Zhang: Conceptualization, review. Xiang Shen: Resources, editing. Sizhe Luo: Software, editing.

Data availability Data is provided within the manuscript or supplementary information files.

Declarations

Competing interests The authors declare no competing interests.

References

Aras AC, Alikasifoglu T, Koç A (2024) Graph receptive transformer encoder for text classification. IEEE Trans Signal Inf Process Netw 10:347–359

Arevalo J, Solorio T, Montes-y Gomez M et al (2020) Gated multimodal networks. Neural Comput Appl 32:10209–10228

Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. arXiv preprint. arXiv:1409.0473

Bhatti UA, Tang H, Wu G et al (2023) Deep learning with graph convolutional networks: an overview and latest applications in computational intelligence. Int J Intell Syst 2023:1–28

Brown T, Mann B, Ryder N et al (2020) Language models are few-shot learners. Adv Neural Inf Process Syst 33:1877–1901

Chen X, Cong P, Lv S (2022) A long-text classification method of chinese news based on Bert and CNN. IEEE Access 10:34046–34057

Cheng Y, Zou H, Sun H et al (2022) HSAN-capsule: a novel text classification model. Neurocomputing 489:521–533

Cui Y, Che W, Liu T et al (2021) Pre-training with whole word masking for Chinese Bert. IEEE/ACM Trans Audio Speech Lang Process 29:3504–3514

Cui H, Wang G, Li Y et al (2022) Self-training method based on GCN for semi-supervised short text classification. Inf Sci 611:18–29

Dai J, Yan H, Sun T et al (2021) Does syntax matter? A strong baseline for aspect-based sentiment analysis with roberta. arXiv preprint. arXiv:2104.04986

Dai Y, Shou L, Gong M et al (2022) Graph fusion network for text classification. Knowl Based Syst 236:107659

Deng J, Cheng L, Wang Z (2021) Attention-based bilstm fused CNN with gating mechanism model for Chinese long text classification. Comput Speech Lang 68:101182

Devlin J, Chang MW, Lee K et al (2019) Bert: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American Chapter of the Association for Computational Linguistics: human language technologies, vol 1 (long and short papers), pp 4171–4186

Dhingra B, Liu H, Yang Z et al (2017) Gated-attention readers for text comprehension. In: Proceedings of the 55th annual meeting of the Association for Computational Linguistics, vol 1: long papers, pp 1832–1846

Ding K, Wang J, Li J et al (2020) Be more with less: hypergraph attention networks for inductive text classification. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP), pp 4927–4936

Dong K, Liu Y, Xu F et al (2023) DCAT: combining multisemantic dual-channel attention fusion for text classification. IEEE Intell Syst 38(4):10–19. https://doi.org/10.1109/MIS.2023.3268228

Dong Y, Yang Z, Cao H (2022) A text classification model based on GCN and BIGRU fusion. In: Proceedings of the 8th international conference on computing and artificial intelligence, pp 318–322

Dosovitskiy A, Beyer L, Kolesnikov A et al (2021) An image is worth 16 × 16 words: transformers for image recognition at scale. In: International conference on learning representations

Du J, Gui L, Xu R et al (2018) A convolutional attention model for text classification. In: 9th CCF International conference on natural language processing and Chinese computing, NLPCC 2017, Dalian, China, 8–12 November 2017, proceedings, vol 6. Springer, Cham, pp 183–195

Duarte JM, Berton L (2023) A review of semi-supervised learning for text classification. Artif Intell Rev 56(9):9401–9469

Fernandes MB, Valizadeh N, Alabsi HS et al (2023) Classification of neurologic outcomes from medical notes using natural language processing. Expert Syst Appl 214:119171

Gao L, Liu Y, Zhu J et al (2024) A cognitively inspired multi-granularity model incorporating label information for complex long text classification. Cogn Comput 16(2):740–755

Gautam M, Sahai N, Yadav AS et al (2022) Sentiment analysis about covid-19 vaccine on twitter data: understanding public opinion. In: 2022 6th International conference on intelligent computing and control systems (ICICCS). IEEE, pp 1487–1493

Guo L, Zhang D, Wang L et al (2018) CRAN: a hybrid CNN-RNN attention-based model for text classification. In: Conceptual modeling: 37th international conference, ER 2018, Xi'an, China, 22–25 October 2018, proceedings, vol 37. Springer, Cham, pp 571–585

Huang L, Ma D, Li S et al (2019) Text level graph neural network for text classification. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pp 3444–3450

Jiang M, D'Souza J, Auer S et al (2022) Evaluating BERT-based scientific relation classifiers for scholarly knowledge graph construction on digital library collections. Int J Digit Libr 23(2):197–215
Jing W, Song X, Di D et al (2021) geoGAT: Graph model based on attention mechanism for geographic text classification. Trans Asian Low-Resour Lang Inf Process 20(5):1–18
Kim Y (2014) Convolutional neural networks for sentence classification. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1746–1751
Kipf TN, Welling M (2016) Semi-supervised classification with graph convolutional networks. arXiv preprint. arXiv:1609.02907
Kramer O (2011) Dimensionality reduction by unsupervised k-nearest neighbor regression. In: 2011 10th international conference on machine learning and applications and workshops. IEEE, pp 275–278
Li Q, Peng H, Li J et al (2022) A survey on text classification: from traditional to deep learning. ACM Trans Intell Syst Technol (TIST) (TIST) 13(2):1–41
Liang Y, Li H, Guo B et al (2021) Fusion of heterogeneous attention mechanisms in multi-view convolutional neural network for text classification. Inf Sci 548:295–312
Lin Y, Meng Y, Sun X et al (2021) BertGCN: transductive text classification by combining GNN and BERT. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp 1456–1462
Liu G, Guo J (2019) Bidirectional lstm with attention mechanism and convolutional layer for text classification. Neurocomputing 337:325–338
Liu M, Liu L, Cao J et al (2022) Co-attention network with label embedding for text classification. Neurocomputing 471:61–69
Liu Y, Zhang Y, Wang Y et al (2023) A survey of visual transformers. IEEE Trans Neural Netw Learn Syst 35(6):7478–7498
Liu Y, Ott M, Goyal N et al (2019) Roberta: a robustly optimized bert pretraining approach. arXiv preprint. arXiv:1907.11692
Liu X, You X, Zhang X et al (2020) Tensor graph convolutional networks for text classification. In: Proceedings of the AAAI conference on artificial intelligence, pp 8409–8416
Luo X, Liu Z, Shang M et al (2020) Highly-accurate community detection via pointwise mutual information-incorporated symmetric non-negative matrix factorization. IEEE Trans Netw Sci Eng 8(1):463–476
Luong MT, Pham H, Manning CD (2015) Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 conference on empirical methods in natural language processing, pp 1412–1421
Manoharan DJS (2021) Capsule network algorithm for performance optimization of text classification. J Soft Comput Paradigm 3(1):1–9
Nagrani A, Yang S, Arnab A et al (2021) Attention bottlenecks for multimodal fusion. Adv Neural Inf Process Syst 34:14200–14213
Niu Z, Zhong G, Yu H (2021) A review on the attention mechanism of deep learning. Neurocomputing 452:48–62
Pham P, Nguyen LT, Pedrycz W et al (2023) Deep learning, graph-based text representation and classification: a survey, perspectives and challenges. Artif Intell Rev 56(6):4893–4927
Salle A, Villavicencio A (2022) Understanding the effects of negative (and positive) pointwise mutual information on word vectors. J Exp Theor Artif Intell 35(8):1161–1199
Song YY, Ying L (2015) Decision tree methods: applications for classification and prediction. Shanghai Arch Psychiatry 27(2):130
Suthaharan S, Suthaharan S (2016) Support vector machine. In: Machine learning models and algorithms for big data classification: thinking with examples for effective learning. Springer, Boston, pp 207–235
Van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9:2579–2605
Vaswani A, Shazeer N, Parmar N et al (2017) Attention is all you need. In: Advances in neural information processing systems, vol 30: annual conference on neural information processing systems, Long Beach, 4–9 December 2017
Velikovi P, Cucurull G, Casanova A et al (2018) Graph attention networks. In: International conference on learning representations
Woo S, Park J, Lee JY, et al (2018) Cbam: Convolutional block attention module. In: Proceedings of the European conference on computer vision (ECCV), pp 3–19
Wu T, Liu Q, Cao Y et al (2023) Continual graph convolutional network for text classification. In: Proceedings of the AAAI conference on artificial intelligence, pp 13754–13762
Xie J, Hou Y, Wang Y et al (2020) Chinese text classification based on attention mechanism and feature-enhanced fusion neural network. Computing 102:683–700
Xu X, Chang Y, An J et al (2023a) Chinese text classification by combining Chinese-BERTology-wwm and GCN. PeerJ Comput Sci 9:e1544

Xu Z, Gu J, Liu M et al (2023b) A question-guided multi-hop reasoning graph network for visual question answering. Inf Process Manag 60(2):103207
Yang Z, Yang D, Dyer C et al (2016) Hierarchical attention networks for document classification. In: Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies, pp 1480–1489
Yang Z, Dai Z, Yang Y et al (2019) Xlnet: generalized autoregressive pretraining for language understanding. In: Advances in neural information processing systems, vol 32
Yang Y, Miao R, Wang Y et al (2022) Contrastive graph convolutional networks with adaptive augmentation for text classification. Inf Process Manag 59(4):102946
Yang G, Jiayu Y, Dongdong X et al (2023) Feature-enhanced text-inception model for Chinese long text classification. Sci Rep 13(1):2087
Yao L, Mao C, Luo Y (2019) Graph convolutional networks for text classification. In: Proceedings of the AAAI conference on artificial intelligence, pp 7370–7377
Yin W, Schütze H, Xiang B et al (2016) ABCNN: attention-based convolutional neural network for modeling sentence pairs. Trans Assoc Comput Ling 4:259–272
Zhang W (2023) Research on chinese news text classification based on ernie model. In: Proceedings of the world conference on intelligent and 3-D technologies (WCI3DT 2022) methods, algorithms and applications. Springer, pp 89–100
Zhang C, Guo R, Ma X et al (2022) W-TextCNN: a textcnn model with weighted word embeddings for Chinese address pattern classification. Comput Environ Urban Syst 95:101819
Zhang S, Ye J, Wang Q (2023) Spa-l transformer: Sparse-self attention model of long short-term memory positional encoding based on long text classification. In: 2023 26th International conference on computer supported cooperative work in design (CSCWD). IEEE, pp 618–623
Zhou P, Shi W, Tian J et al (2016) Attention-based bidirectional long short-term memory networks for relation classification. In: Proceedings of the 54th annual meeting of the association for computational linguistics (vol 2: short papers), pp 207–212
Zhu H, Koniusz P (2020) Simple spectral graph convolution. In: International conference on learning representations

Authors and Affiliations

Yangshuyi Xu[1] · Guangzhong Liu[1] · Lin Zhang[1] · Xiang Shen[1] · Sizhe Luo[1]

✉ Guangzhong Liu
gzhliu@shmtu.edu.cn

Yangshuyi Xu
xuyangshuyi@163.com

Lin Zhang
linzhang@shmtu.edu.cn

Xiang Shen
shenxiang1107@163.com

Sizhe Luo
luosizhe@stu.shmtu.edu.cn

[1] College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China