Journal of Computers Vol. 36 No. 3, June 2025, pp. 55-67
https://doi.org/10.63367/199115992025063603005

Financial Risk Prediction Model Based on Bert and Attention Mechanism Fusion

Jun Qian* and Yubao Wu

School of information technology, Nanjing Police University, Nanjing, 210023, China
qianj@nfpc.edu.cn, eice_wu@126.com

Received 28 March 2024; Revised 26 April 2024; Accepted 26 April 2024

Abstract. The advent of the big data era has led to an explosion in the volume of text generated by social media platforms, containing rich emotional information. The utilization of computers to accurately categorize the sentiment of extensive volumes of public financial commentary aids in evaluating the fundamentals of current financial expectations. Financial reviews, as short texts, are characterized by sparsity and high-dimensional features. The interpretation of their semantics is diverse and highly context-dependent, making text sentiment classification tasks more challenging. Therefore, it is necessary to mine the features of short texts from multiple perspectives and learn the hidden emotional information. In this paper, we propose a fnancial risk prediction model Bert-HG based on the fusion of Bert and attention mechanism, in which Bert is used to train word vector representations, and textual features are extracted by the dual information processing channels—the Semantic Information Channel Module (SAIC) and the Structural Information Channel Module (SCIC). The SAIC Module utilizes hierarchical attention mechanisms to extract text semantic information, while the SCIC Module utilizes graph attention mechanisms to pay attention to sentence-level dependencies within the documents and extract text structural information. The obtained text vectors are concatenated and input into a fully connected layer, after which they are classified into different sentiments by the softmax function. The experimental results show the proposed model outperforms other baseline models.

Keywords: sentiment analysis, short text classification, attention mechanism, natural language processing, deep learning

1 Introduction

The complexity and uncertainty of financial markets have increased the difficulty of financial analysis tasks, involving extensive data processing and information mining. In financial market analysis, financial risk analysis can be used to predict the future price trends of financial products, which has attracted much attention as a subtopic of financial market analysis.

With the rapid development of Internet technology, the widespread application of Web 2.0 and social media has significantly increased the volume of User-generated content (UGC), Facilitating investors to generate and share various financial reviews on social media platforms, thereby aiding other users to make the best decisions [1]. For instance, when the discussion and sentiment of a certain stock on social media start to significantly increase, it can be inferred that the attention and sentiment towards this stock are rising, which might have a positive or negative impact on its stock price. Studies have shown that public sentiment and financial sentiment play an important role in investment decisions [2]. Moreover, social media messages have been confirmed as a useful resource for detecting sentiments in the field of natural language processing [3]. However, capturing financial sentiment is not easy, as traders often face influences from various factors such as news, earnings reports, and the general perception of companies, as well as the views of other traders. Recently, machine learning has been extensively applied in financial analysis, and many researchers are exploring how to extract key features from financial news [4] and financial social networks [5] to understand user financial sentiment, and improve the accuracy of financial analysis by evaluating the financial sentiment of a large number of users.

Regarding financial-related messages on social media, they are generally classified into three categories: corporate financial reports, financial news reports, and financial reviews on financial social networks. Financial reviews, as short texts, exhibit characteristics such as oral language, noisy features, limited text features, and lack of prominent themes. Moreover, short texts inherently possess characteristics of being large-scale, sparse,

* Corresponding Author

and irregular, which makes it challenging to extract the features of financial short text reviews and perform accurate sentiment classification. Financial reviews often encompass multiple topics and various emotions. For example, “Today’s weather is great, but the stock market is terrible”, the text contains both weather and stock market topics, as well as both positive and negative emotions. If semantic modeling directly analyzes mixed texts that contain multiple emotional tendencies, it might increase the difficulty of emotional inclination judgments. Financial reviews generally contain a few practically significant words or phrases that are vital for classification results. Accurately extracting the key words and phrases that significantly affect the sentiment classification results can better represent the semantic information of the text, thereby improving the accuracy of text sentiment classification. For instance, “Although the stock market has been volatile recently, even to the level of being extremely poor, I still believe in this stock and think it will go up”. Although the text expresses a bullish sentiment towards the stock market, the presence of negative evaluations like “extremely poor” may lead the classifier to misclassify the sentence into a negative category. Clearly, the keywords “stock” and “go up” and the key phrase “I still believe in this stock and think it will go up” contain positive emotions that play a more significant role in the classification results, warranting a higher weight. Intuitively, considering the semantic and structural relationships within the sentence can improve the accuracy of text sentiment classification.

To address these challenges and fill the research gap, we propose a financial risk prediction model, Bert-HG, based on Bert and attention mechanism fusion. To enhance the capability of word vector representation, we use the pre-trained model Bert instead of traditional Word2Vec for word vector encoding during text vectorization, and add a layer of average pooling in the Bert layer to extract the semantic information of each sentence. The model adopts an information processing dual-channel module, in which the semantic information channel module considers contextual information and balances the importance of words and sentences in different contexts through the hierarchical attention network, thereby more accurately extracting the semantic information of the text. The structure information channel module utilizes graph attention networks to extract hierarchical structure features from input text. The main contributions of this article are as follows:

(1) The addition of an average pooling layer to Bert, extracting semantic information from each sentence, constructing sentence-text graphs, and utilizing graph attention networks to extract hierarchical structure features from input text.

(2) We propose a novel sentiment classification prediction model called Bert-HG. By combining with hierarchical attention networks and graph attention networks, it can fully extract the semantic information and structural information of the text. Thus, Bert-HG can effectively improve the performance of financial text sentiment classification.

(3) We validate the proposed method using the financial social network StockTwits’ financial information dataset. Our proposed model outperforms other baselines in experimental results.

The paper is organized as follows. Section 2 introduces the related work; Section 3 describes the proposed financial risk prediction model Bert-HG based on the fusion of Bert and attention mechanism; Section 4 provides detailed experimental results and analysis. Finally, in Section 5, we conclude our work and extend the work in the future.

2 Related Work

Text sentiment analysis is a research field aimed at analyzing subjective information in texts, such as opinions, attitudes, emotions, etc. [6]. Financial sentiment analysis is an important research area in finance, with a focus on analyzing financial news articles [7], using financial social networks similar to Twitter and StockTwits to analyze financial mood [8], garnering recent attention from researchers. Financial text sentiment analysis methods can be divided into three categories: lexicon-based, machine learning-based, and deep learning-based.

2.1 Financial Text Sentiment Analysis

The method based on emotional lexicons is a traditional sentiment analysis technique with the core idea of leveraging pre-built emotion lexicons to infer the emotional inclination of a text by calculating the number and polarity of sentiment words in the text. Although this approach offers the advantage of simplicity and ease of understanding, its accuracy is largely limited by the quality and scope of the emotion dictionary used. Building high-quality emotion dictionaries requires significant human and material resources, and they often fail to fully cover the various expressions of emotions and contexts in various languages. This approach is not well-suited for emerging words, as these words may not be included in existing emotion dictionaries. Therefore, when dealing with complex and diverse contexts and emotional expressions, methods based on emotion dictionaries may exhibit limitations [9].

From the 1960s to the 2010s, text classification models using traditional machine learning algorithms dominated, such as Naive Bayes [10], K-nearest neighbors [11], and Support Vector Machines [12]. The methods offer simplicity and high accuracy, making them suitable for text sentiment analysis [13]. However, they heavily depend on domain knowledge and become unstable with more training samples or a domain switch.

Since the early 2010s, the accumulation of data samples in the field of financial texts has made it possible to use deep learning for financial information sentiment analysis. Deep learning methods overcome the limitations of conventional machine learning methods which rely heavily on manually designed rules and features, enabling the formation of high-dimensional embeddings through learning, which is beneficial for enhancing the generalization of the model and thus more efficiently carrying out text mining. However, given the complexity of natural language structures, current deep learning models typically only address certain issues while overlooking others. The emergence of large language models in NLP has had a significant impact, with more researchers paying attention to the applications of these pre-trained models. Li et al. [14] used pre-trained word embeddings and BiLSTM for sentiment recognition, capturing semantic and sentiment information. However, their model didn't consider the varying importance of words and subsentences within the text. Tang et al. [15] in used pre-trained word embeddings, combining TextCNN and BiLSTM for sentiment classification. However, BiLSTM still struggles to capture all the dependencies of each preceding word. To enhance the expression of emotional tendency, some studies leverage language resources or external knowledge to enrich the information input of the model, thereby improving the understanding and inference capabilities of the sentiment analysis model for emotional expression. Su et al. [16] extracted semantic information using topic model fusion BiLSTM and CNN, enhancing the accuracy of short text classification tasks. In Li et al. [17], graph neural network and LSTM were combined to extract semantic and syntactic features of sentences and classify the sentiment of Weibo Comments. Current text modeling methods often treat the entire text as a whole [18], ignoring the semantic structural relationships within the text.

2.2 Text Feature Representation

The core of text sentiment analysis lies in text vector representation and feature extraction. Given the vast number financial information words, common text encoding methods may result in enormous representation sizes and a large number of training parameters. Hence, introducing word embedding techniques is necessary to reduce the dimensionality of word representation vectors. Word2Vec [19] is a neural network language model that considers contextual semantic information and can avoid dimensionality issues compared to traditional models. However, since Word2Vec is a static model, it cannot solve the problem of polysemy. In 2014, Cho et al. and Sutskever et al. proposed a fully recurrent neural network-based encoder-decoder model for sequence-to-sequence tasks [20], but the RNN structure has disadvantages such as inefficiency and lack of parallelizability. To address this issue, Transformer [21] employs Self-Attention mechanisms to replace the recurrent neural network's sequence structure, effectively solving the long-term dependency issues in natural language processing. Bert, proposed by Devlin et al. [22] in 2018, is a word vector pre-training model that utilizes a bidirectional Transformer encoder structure for encoding. Bert employs only the encoder portion of the Transformer, utilizing masking language models and next sentence prediction for word vector training. Bert has strong scalability, providing generalized and highly effective pre-trained models for NLP tasks.

3 Methods

Fig. 1 shows the overall framework of the Bert-HG model. This model is primarily comprised of three main components: the word vector encoding module based on Bert, the information processing dual channel module, and the text sentiment prediction module. From section 3.1 to section 3.3, we will elaborate on the implementation methods of these modules.

3.1 Word Vector Encoding Module

The model structure of Bert is shown in Fig. 2, where E_1, E_2, …, E_n represents the input sequence of Bert, *Trm* represents the Transformer Encoder model, and T_1, T_2, …, T_n represents the output word vector sequence of Bert.

Suppose a text S contains n sentences, denoted as $S = \{sen_1, sen_2, \ldots, sen_i, \ldots, sen_n\}$, where sen_i represents the *i-th* sentence. Each sentence contains at most t words, so the *i-th* sentence sen_i can be represented by $sen_i = \{w_{i1}, w_{i2}, \ldots, w_{it}\}$, where w_{it} represents the *t-th* word of the sen_i. Using Bert to encode the text S results in the hidden layer representation of the text S:

$$Bert(S) = \left[b_{[cls]}, bw_{11}, bw_{12}, \cdots, bw_{it}, \cdots, bw_{nt}, b_{[sep]}\right] \in R^{(nt+2)\times d}. \quad \textbf{(1)}$$

Where d represents the dimension of the hidden layer vector, $[CLS]$ corresponds to the word vector of the first word in the input text, $[SEP]$ corresponds to the word vector of the last word in the input text, and it serves to divide different sentences.

As shown in Fig. 1, the semantic representation bs_i of the *i-th* sentence in the text S is obtained using mean pooling:

$$bs_i = MeanPooling\left(\left[bw_{i1}, bw_{i2}, \cdots, bw_{it}\right]\right) \in R^{i\times d}. \quad \textbf{(2)}$$

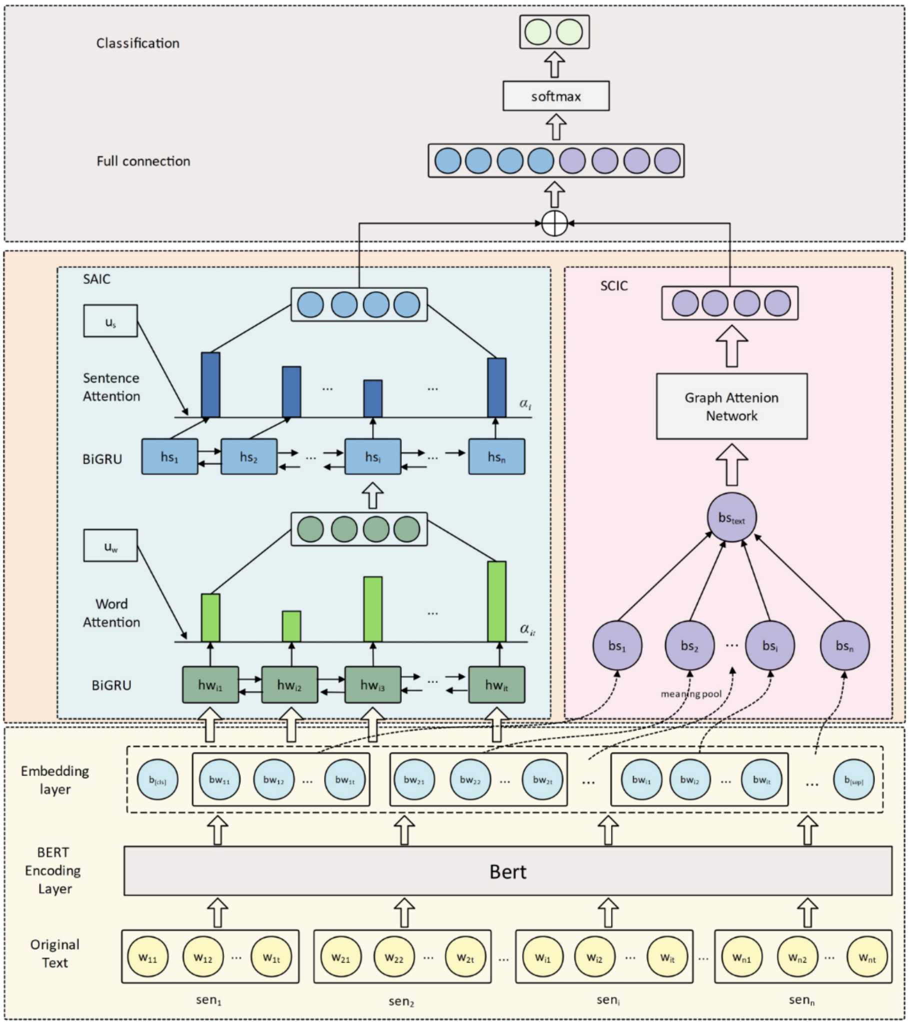

Fig. 1. The overall framework of the Bert-HG model

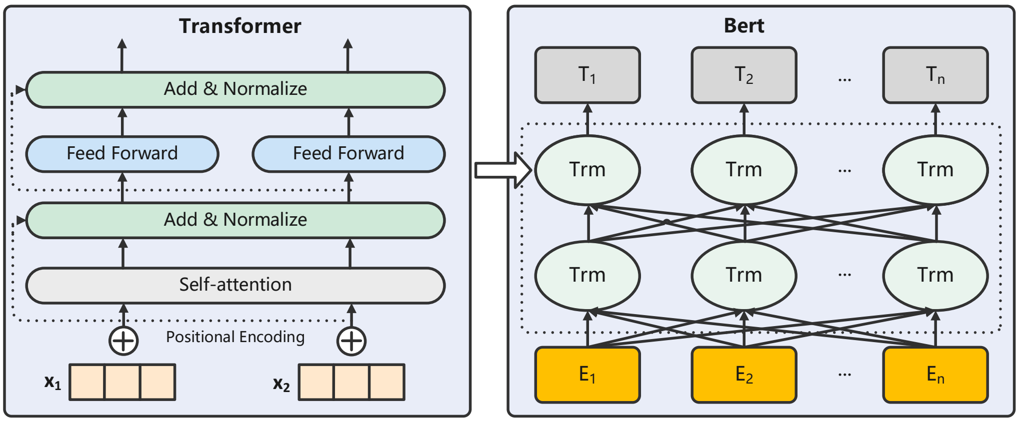

Fig. 2. The structure diagram of Bert

3.2 Information Processing Dual Channel Module

SAIC Module. In terms of text sentiment classification tasks, the contributions of words and sentences are not equal, but rather influenced by their contextual relationships. Therefore, semantic information channels adopt a hierarchical attention network to fully extract the features of words and subsentences within documents [23]. The hierarchical attention network combines the word-level and subsentence-level attention mechanisms by assigning different attention weights to different words and subsentences to extract words and subsentences that are key to the classification outcome. The extraction of text semantic features based on HAN is divided into four steps:

Word Encoder. 3.1 section obtains the vector $\boldsymbol{bw}_{i1}$, $\boldsymbol{bw}_{i2}$, $\boldsymbol{bw}_{it}$ from the word embedding module in the BiGRU input, which is utilized for deep feature extraction. The BiGRU consists of two parts - forward GRU and reverse GRU. This module uses the BiGRU to aggregate information from both the forward and reverse directions of the word $\boldsymbol{w}_{it}$ to obtain its semantic representation, enabling better capture of the semantic features of the word in context. The calculation process of the semantic representation of the word $\boldsymbol{w}_{it}$ is illustrated in Equations (3) to (5):

$$\overrightarrow{hw_{it}} = \overrightarrow{GRU}\left(bw_{it}\right) . \tag{3}$$

$$\overleftarrow{hw_{it}} = \overleftarrow{GRU}\left(bw_{it}\right) . \tag{4}$$

$$hw_{it} = \left[\overrightarrow{hw_{it}}, \overleftarrow{hw_{it}}\right] . \tag{5}$$

Where $\overrightarrow{hw_{it}}$ and $\overleftarrow{hw_{it}}$ represent the hidden layer states of the forward GRU and reverse GRU respectively. hw_{it} encompasses information from both the forward and reverse directions of the word w_{it}.

Word Attention. The attention mechanism utilizes the attention mechanism to highlight words that are crucial for sentiment classification and identify the relationships between all words in the sequence and adjust their representations accordingly based on their importance. In Equation (6), the hidden layer vector $\boldsymbol{hw}_{it}$ is further represented by the one-layer MLP unit $\boldsymbol{u}_{it}$.

$$u_{it} = \tanh\left(W_w hw_{it} + b_w\right) . \tag{6}$$

Where Ww is the weight matrix, and b_w is the corresponding bias vector.

The attention mechanism assigns different weights to the semantic encoding of word embeddings based on their importance. In Equation (7), the importance of a word is measured by the similarity of the contextual word u_w of u_{it}. The normalized importance weight α_{it} is obtained by applying the softmax function.

$$\alpha_{it} = \frac{\exp\left(u_{it}^{\top} u_w\right)}{\sum_t \exp\left(u_{it}^{\top} u_w\right)} . \quad \textbf{(7)}$$

For the *i-th* sentence of the given text, its vector s_i is represented by the weighted sum of the word embedding vectors, and the result is inputted into the Sentence Encoder layer for further processing, and the result is inputted into the Sentence Encoder layer for further processing.

$$s_i = \sum_t \alpha_{it} hw_{it} . \quad \textbf{(8)}$$

Where the weights α_{it} reflect the importance of each word in the subsentence.

Sentence Encoder. After obtaining the vector of the subsentence $\boldsymbol{s_i}$, the subsentence is also encoded using BIGRU. The calculation process of the semantic representation of the subsentence $\boldsymbol{s_i}$ is shown in Equations (9-11):

$$\overrightarrow{hs_i} = \overrightarrow{GRU}\left(s_i\right) . \quad \textbf{(9)}$$

$$\overleftarrow{hs_i} = \overleftarrow{GRU}\left(s_i\right) . \quad \textbf{(10)}$$

$$hs_i = \left[\overrightarrow{hs_i}, \overleftarrow{hs_i}\right] . \quad \textbf{(11)}$$

Sentence Attention. In Equation (12), the hidden layer vector $\boldsymbol{hs_i}$ is further represented by $\boldsymbol{u_i}$ using a single-layer MLP.

$$u_i = \tanh\left(W_s hs_i + b_s\right) . \quad \textbf{(12)}$$

Where W_s is the weight matrix, and b_s is the corresponding bias vector.

The attention mechanism assigns different weights to the semantic encoding of the subsentence vectors. In Equation (13), the importance of a word is measured by the similarity of the contextual word u_s of u_i. The normalized importance weight α_i is obtained by applying the softmax function.

$$\alpha_i = \frac{\exp\left(u_i^{\top} u_s\right)}{\sum_i \exp\left(u_i^{\top} u_s\right)} . \quad \textbf{(13)}$$

The weights α_i of each subsentence reflect the importance of the subsentence in the text.

In Equation (14), the vector representation hs'_{text} of the text S is obtained by weighted summation of the subsentence embedding vectors.

$$hs'_{text} = \sum_t \alpha_i hs_i . \quad \textbf{(14)}$$

SCIC Module. Graph Neural Networks use message passing to extract and mine features and patterns from graph structure data [24]. In graph structure, the nodes adjacent to each other are usually assigned the same weight, but in fact, the importance of adjacent nodes varies to some extent. To better capture the complex relationships between nodes, this paper utilizes a Graph Attention Network (GAT) [25] to map the structural features of subsentences in a given text, allowing the current node to pay more attention to important nodes. As shown in Fig. 1, a text structure network graph $G = (V, E)$ is constructed, where V is the set of node, E is the set of edges. The network graph G contains subsentence nodes and text nodes, and directed connections are established between them. Suppose the number of subsentences in the text is n. If the number of subsentences is less than n, 0 is used to fill in. The semantic information encoding bs_i of the *i-th* subsentence in the text S is obtained by Equation (2), and it is used as the initial representation of the subsentence node. The text node is randomly initialized. In the GAT, we introduce multi-head attention to enhance the model's capabilities and stabilize the training process. Considering there are k attention heads, the calculation process is as follows:

For the *k-th* head, its attention coefficient is calculated as shown in Equation (15):

$$e_{ij}^{(k)} = \text{LeakyReLU}\left(w_k^T \cdot \left[W_1^{(k)} bs_i \,||\, W_2^{(k)} bs_j \right] \right) . \quad \textbf{(15)}$$

Where $w_k^T \in \mathbb{R}^{2d_w \times 1}$ represents the parameter vector of the attention head k, $||$ represents the connecting operation on the vector, and $W_1^{(k)}$, $W_2^{(k)} \in \mathbb{R}^{d_w \times d_w}$ represents the weight matrix corresponding to the *k-th* attention head.

The attention coefficients of each attention head are normalized using the softmax function, as shown in Equation (16):

$$\alpha_{ij}^{(k)} = softmax\left(e_{ij}^{(k)}\right) = \frac{\exp\left(e_{ij}^{(k)}\right)}{\sum_{k=1}^{K} \sum_{r \in \mathcal{N}_i} \exp\left(e_{ir}^{(k)}\right)} . \quad \textbf{(16)}$$

Where $\mathcal{N}_i$ represents the set of neighbor nodes of node i .

The features of the node i's neighbor nodes are weighted by the attention coefficients for each attention head, and a new representation of the node i is obtained, as shown in Equation (17):

$$bs_i^{'(k)} = \tanh\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij}^{(k)} W_3^{(k)} bs_j \right) . \quad \textbf{(17)}$$

where $W_3^{(k)}$ represents the weight matrix of the *k-th* attention head.

The representations of the node features obtained from each attention head are concatenated, resulting in the final node feature representation bs_i' , as shown in Equation (18):

$$bs_i^{'} = \text{Concat}\left(bs_i^{'(1)}, bs_i^{'(2)}, \ldots, bs_i^{'(k)} \right) \quad \textbf{18)}$$

After extracting structural features using GAT, the final hierarchical structural feature bs'_{text} is obtained.

3.3 Text Sentiment Prediction Module

The text sentiment prediction module integrates the outputs of the semantic information channel and the structural information channel, connecting the two types of text representations, obtaining $S_{text} = hs'_{text} \| bs'_{text}$ as the final representation, and S_{text} is linearly connected and reduced to the classification dimension. The softmax function is used to predict the final classification probability, as shown in Equation (19):

$$\hat{y} = softmax\left(W_y S_{text} + b_y \right) . \quad \textbf{(19)}$$

Where W_y represents the weight matrix, and b_y represents the bias term.

The model is trained using the cross-entropy loss function, as shown in Equation (20):

$$Loss = -\sum_{i=1}^{m} y_i \log \hat{y}_i \ . \tag{20}$$

Where m represents the number of samples, y_i represents the true label of the i-th sample, and $\hat{y}_i$ represents the predicted label of the i-th sample.

4 Experimental

In this section, we assess the effectiveness of the Bert-HG model through extensive experiments conducted on datasets.

4.1 Experimental Setup

Experimental Data. StockTwits is a financial social network platform designed specifically for individuals interested in trading and investing [26]. It is similar to Twitter, but StockTwits focuses more on financial and economic discussions. Users on StockTwits share their trading experiences and opinions on various stocks, and their tweets are limited to 140 characters or less. Users can label their tweets as bullish, bearish, or neutral to express their financial emotions, as shown in Fig. 3, where the user's tweet about Tesla stock is labeled as Bullish. Research has shown that sentiment analysis of StockTwits tweets can be a good predictor of important financial information, such as future asset prices.

Fig. 3. Example of StockTwits tweet

Similar to the literature [27], we selected the five most prominent stocks from various industries, namely Apple Inc. (AAPL), Facebook, Tesla Inc. (TSLA), U.S. Bank (BAC), and Nvidia (NVDA), and the total number of relevant tweets was 102,408. Each tweet included the user ID, tweet ID, tweet content, number of followers, time stamp, and other properties of record retention. After preprocessing the raw dataset and removing special characters, we cleaned the tweets without emotion labels in the original dataset. The final dataset contained 94,388 tweets with emotion labels, with the user information including the number of identified tweets from October 2021 to February 2022, with the tweet content and emotion labels retained. The data set is shown in Table 1.

Table 1. Examples of dataset

Tweets	Emotion label
i think the market is going to rip faces next week, i feel a wave of optimism coming.	Bullish
futures will be green tomorrow!	Bullish
all smart money is out, anticipate more selling overnight.	Bearish
our president may be investigating for voter fraud and espionage, if this doesn't bring the market down nothing will.	Bullish

The data set is divided into 80% training data, 10% validation data, and 10% test data. The statistical information of the dataset is shown in Table 2.

Table 2. Statistics on dataset

Dataset	Training	Validation	Test	Total
Bullish samples	33814	4096	4109	42019
Bearish samples	41698	5342	5329	52369
Total	75512	9438	9438	94388
Average length	74.35	74.14	75.70	-
Median	67	67	69	-

The relationship between the length and quantity of text in the dataset is presented in Fig. 4, and it is observed that most of the comments in the dataset are short texts, with texts less than 100 characters making up about 70% of the total, and texts between 100 and 150 characters making up about 25% of the total, and texts longer than 150 characters making up about 5% of the total.

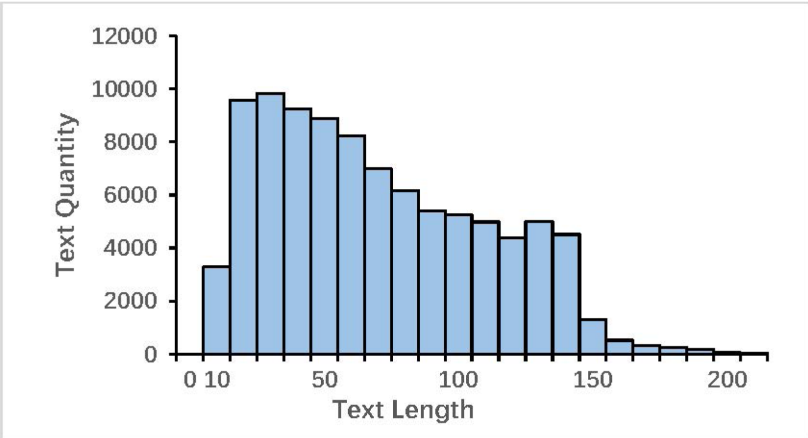

Fig. 4. The relationship between the length and quantity of text

Evaluation Metrics. This study uses Precision, Recall, F1-score, and Accuracy to evaluate the effectiveness of a text sentiment classification model. In the experiment, an emotional label of Bullish indicates a positive sentiment, while an emotional label of Bearish indicates a negative sentiment. The calculation formulas for each metric are shown in Equations (21)-(24):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad . \tag{21}$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad . \tag{22}$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad . \tag{23}$$

$$\text{F1-Score} = 2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \quad . \tag{24}$$

Where *TP*, *TN*, *FP*, and *FN* respectively represent True Positive, True Negative, False Positive, and False Negative.

Parameter Settings. In our model architecture, we utilized the bert-base-uncased pre-trained language model with a hidden layer dimension of 768. The learning rate is set to 3.0e-06, and the optimizer is AdamW. Dropout is set to 0.3 after fully connected layers. The training epoch is set to 1200 and the batch size is 32. To validate the effectiveness of the model proposed in this paper, we conducted a comparative experiment against the baseline model.

4.2 Model Comparison

To validate the proposed model's efficacy, we partitioned the baseline model into two distinct groups for comparative experiments. The first group of experiments compared baseline models using word2vec and Bert to assess the impact of different word vectors on classification outcomes. The second group set entailed ablation experiments, removing certain modules from the model to assess their impact on overall performance.

Group 1: Deep Learning Models Based on Word2Vec and Bert Embeddings

In the first group of experiments, we compared the proposed model against the following current classic and advanced baseline models in the field of financial text classification.

CNN: CNN is a type of deep neural network with a convolutional structure, originally applied to image processing. Numerous studies have proven that CNN can effectively improve the accuracy of text sentiment classification [28].

RCNN: RCNN is a type of RNN and CNN combined model for text classification [29].

BiGRU: BiGRU consists of two parts, forward GRU and reverse GRU, which utilize BiGRU to extract features from both the forward and backward directions.

BiGRU+Attention: By combining BiGRU with an attention mechanism, we can effectively capture the contextual features of the sequence.

CNN+BiGRU+Attention: By utilizing CNN to capture static features and BiGRU to capture contextual features, the attention mechanism is applied to highlight important words. This model has been proven to improve text sentiment classification performance [30].

Group 2: Ablation Experiments

To validate the effectiveness of each module in the model, we observed the effects of removing the SAIC module and SCIC module from the model.

w/o_SAIC: By removing the SAIC module, we utilize GAT to extract text structural characteristics for classification prediction.

w/o_SCIC: By removing the SCIC module, we utilize HAN to extract semantic features from the text, and make classification predictions.

Without SAIC and SCIC: The SAIC module and SCIC module are simultaneously removed, and BERT is utilized to exclusively extract the <cls> tag from the text sequence in order to provide sentence-level semantic embedding features for classification prediction.

4.3 Experimental Results and Analysis

Comparative Experiments. This section compares the classification results of two baseline models. From Table 3, the Bert-based model performs overall better than the word2vec based deep learning model for short text classification, indicating Bert word vector embeddings are effective. Additionally, the CNN and RCNN-based models perform better than BiGRU, suggesting BiGRU does not fully utilize long-distance feature extraction when classifying short texts. The CNN model can converge faster. Additionally, the performance of the CNN+BiGRU+Attention model outperforms the other baseline models, indicating that the introduction of the attention mechanism has a positive impact on the classification results. By comparing with the baseline model, our proposed model Bert-HG achieves the best performance.

Table 3. Overall model comparison results

Category	Model	Class	Precision	Recall	F1-Score	Accuracy
Word2Vector+baseline	CNN	Neg	0.7436	0.73448	0.7391	0.7742
		Pos	0.7972	0.80482	0.8009	
	RCNN	Neg	0.7392	0.75882	0.7489	0.7785
		Pos	0.8101	0.79367	0.8018	
	BiGRU	Neg	0.7361	0.73205	0.7340	0.7691
		Pos	0.7942	0.79766	0.7959	
	BiGRU+Attention	Neg	0.7364	0.75638	0.7462	0.7761
		Pos	0.8081	0.79129	0.7996	
	CNN+BiGRU +Attention	Neg	0.7423	0.76782	0.7548	0.7829
		Pos	0.8161	0.79452	0.8051	
Bert+baseline	CNN	Neg	0.7965	0.7762	0.7862	0.8174
		Pos	0.8325	0.8487	0.8405	
	RCNN	Neg	0.7979	0.7892	0.7935	0.8212
		Pos	0.8388	0.8459	0.8423	
	BiGRU	Neg	0.7961	0.7758	0.7858	0.8169
		Pos	0.8322	0.8483	0.8402	
	BiGRU +Attention	Neg	0.7988	0.7824	0.7905	0.8195
		Pos	0.8348	0.8481	0.8414	
	CNN+BiGRU+Attention	Neg	0.8021	0.7897	0.7958	0.8236
		Pos	0.8397	0.8497	0.8447	
Ours	Bert-HG	Neg	**0.8036**	**0.8109**	**0.8072**	**0.8314**
		Pos	**0.8531**	**0.8472**	**0.8501**	

Ablation Experiments. We conducted an ablation study to validate the effectiveness of different modules in Bert-HG, as shown in Table 4.

Table 4. Results of ablation experiment

Model	Class	Precision	Recall	F1-Score	Accuracy
w/o_SAIC	Neg	0.7952	0.7994	0.7973	0.8231
	Pos	0.8447	0.8413	0.8430	
w/o_SCIC	Neg	0.7982	0.8067	0.8024	0.8271
	Pos	0.8497	0.8427	0.8462	
w/o_SAIC&SCIC(Bert)	Neg	0.7986	0.7743	0.7863	0.8168
	Pos	0.8300	0.8495	0.8396	
Bert-HG	Neg	**0.8036**	**0.8109**	**0.8072**	**0.8314**
	Pos	**0.8531**	**0.8472**	**0.8501**	

The efficacy of the SAIC Module. Table 4 shows that removing the SAIC module has a significant negative effect on the results, suggesting that the hierarchical attention network plays a crucial role in extracting text semantic information. However, its performance is still higher than that of the Bert model, confirming the effectiveness of graph networks in text processing.

The efficacy of the SCIC Module. Through Table 4, it can be observed that the removal of the SCIC module has resulted in a decrease in model performance, yet the performance is still higher than that of the Bert-based model classification. This demonstrates that the introduction of text structural features can enhance model performance, and the structural information channel module is effective within the model.

The efficacy of the information processing dual-channel module. By simultaneously removing the SAIC module and the SCIC module, as shown in Table 4, the model with no information processing dual-channel module performs significantly worse than in the first two cases. This validates the efficacy of the information processing dual-channel module.

5 Conclusion

Financial review short texts exhibit features such as colloquialism and high noise levels. They typically consist of a few significant words, which may encompass various topics and emotions. This leads to relatively poor performance in financial review text sentiment classification. This study proposes a method, Bert-HG, based on the fusion of BERT and attention mechanism for financial risk prediction. Experimental results show Bert-HG outperforms other baseline models. Current methods only consider the financial review text itself, ignoring social influence of users. Clearly, the bullish and bearish opinions of users with more followers than ordinary users are more valuable for reference. In future, we will consider social influence of users on tweets, further improving prediction accuracy.

Acknowledgement

This paper is sponsored by 2024 Outstanding young backbone teacher of Jiangsu Universities "Qinglan Project" and Technical research plan of the Ministry of public security Research on dark network investigation and forensic technology based on operation behavior (2020SJYJC27).

References

[1] B.M. Barber, T. Odean, The internet and the investor, Journal of Economic Perspectives 15(1)(2001) 41-54. https://doi.org/10.1257/jep.15.1.41

[2] H. Chen, L. Shan, C. H. Wang, Investment Sentiment in Finance Market, in: Proc. 2021 3rd International Conference on Economic Management and Cultural Industry (ICEMCI 2021), 2021. https://doi.org/10.2991/assehr.k.211209.541

[3] Y. Ge, J. Qiu, Z. Liu, W. Gu, L. Xu, Beyond negative and positive: Exploring the effects of emotions in social media during the stock market crash, Information processing & management 57(4)(2020) 102218. https://doi.org/10.1016/j.ipm.2020.102218

[4] S.Y. Li, J.H. Kong, News Sentiment and the Risk of a Stock Price Crash Risk: Based on Financial Dictionary Combined Bert-DCA, Discrete Dynamics in Nature and Society 2022(2022) 8305947. https://doi.org/10.1155/2022/8305947

[5] S. Yıldırım, D. Jothimani, C. Kavaklioğlu, A. Başar, Deep learning approaches for sentiment analysis on financial microblog dataset, in: Proc. 2019 IEEE International Conference on Big Data (Big Data), 2019. https://doi.org/10.1109/BigData47090.2019.9006056

[6] W. Medhat, A. Hassan, H. Korashy, Sentiment analysis algorithms and applications: A survey, Ain Shams Engineering Journal 5(4)(2014) 1093-1113. https://doi.org/10.1016/j.asej.2014.04.011

[7] X. Man, T. Luo, J.W. Lin, Financial sentiment analysis (fsa): A survey, in: Proc. 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS), 2019. https://doi.org/10.1109/ICPHYS.2019.8780312

[8] B.N. Huang, J.K. Wei, Research on deep learning-based financial risk prediction, Scientific programming 2021(2021) 6913427. https://doi.org/10.1155/2021/6913427

[9] C.-C. Chen, H.-H. Huang, H.-H. Chen, NTUSD-Fin: a market sentiment dictionary for financial social media data applications, in: Proc. of the 1st financial narrative processing workshop (FNP 2018), 2018.

[10] M.E. Maron, Automatic indexing: an experimental inquiry, Journal of the ACM (JACM) 8(3)(1961) 404-417. https://doi.org/10.1145/321075.321084

[11] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Transactions on Information Theory 13(1)(1967) 21-27. https://doi.org/10.1109/TIT.1967.1053964

[12] T. Joachims, Text categorization with support vector machines: Learning with many relevant features, in: Proc. European conference on machine learning, 1998. https://doi.org/10.17877/DE290R-5097

[13] Q. Li, H. Peng, J.X. Li, C.Y. Xia, R.Y. Yang, L.C. Sun, P.S. Yu, L.F. He, A Survey on Text Classification: From Shallow to Deep Learning, ArXiv:2008.00364. https://doi.org/10.48550/arXiv.2008.00364

[14] X.L. Li, Y.Y. Lei, S.W. Ji, Bert-and BiLSTM-based sentiment analysis of online Chinese buzzwords, Future Internet 14(11)(2022) 332. https://doi.org/10.3390/fi14110332

[15] H.Q. Tang, R.R. Zhang, Research on Chinese Microblog Sentiment Classification Based on TextCNN-BiLSTM Model, Journal of Information Processing Systems 19(6)(2023) 842-857. https://doi.org/10.3745/JIPS.04.0299

[16] X.S. Su, H.F. Song, Y.F. Wang, M. Wang, A short text topic classification method based on feature expansion and bi-directional neural network, in: Proc. 2022 International Conference on Artificial Intelligence, Information Processing and Cloud Computing (AIIPCC), 2022. https://doi.org/10.1109/AIIPCC57291.2022.00089

[17] Y. Li, N.F. Li, Sentiment analysis of Weibo comments based on graph neural network, IEEE Access 10(2022) 23497-23510. https://doi.org/10.1109/ACCESS.2022.3154107

[18] B.X. Xue, C. Zhu, X. Wang, W.J. Zhu, The study on the text classification based on graph convolutional network and BiLSTM, Applied Sciences 12(16)(2022) 8273. https://doi.org/10.3390/app12168273

[19] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, arXiv:1310.4546. https://doi.org/10.48550/arXiv.1310.4546

[20] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, arXiv:1409.3215. https://doi.org/10.48550/arXiv.1409.3215

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, arXiv:1706.03762. https://doi.org/10.48550/arXiv.1706.03762

[22] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, ArXiv:1810.04805. https://doi.org/10.48550/arXiv.1810.04805

[23] Z.C. Yang, D.Y. Yang, C. Dyer, X.D. He, A. Smola, E. Hovy, Hierarchical attention networks for document classification, in: Proc. of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies, 2016. https://doi.org/10.18653/v1/N16-1174

[24] J.J. Zhang, Z.Y. Cui, H.J. Park, G. Noh, BHGAttN: A Feature-Enhanced Hierarchical Graph Attention Network for Sentiment Analysis, Entropy 24(11)(2022) 1691. https://doi.org/10.3390/e24111691.

[25] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, ArXiv:1710.10903. https://arxiv.org/abs/1710.10903.

[26] S. Sohangir, D.D. Wang, A. Pomeranets, T.M. Khoshgoftaar, Big Data: Deep Learning for financial sentiment analysis, Journal of Big Data 5(3)(2018) 1-25. https://doi.org/10.1186/s40537-017-0111-6

[27] R. Batra, S.M. Daudpota, Integrating StockTwits with sentiment analysis for better prediction of stock price movement, in: Proc. 2018 international conference on computing, mathematics and engineering technologies (ICoMET), 2018. https://doi.org/10.1109/ICOMET.2018.8346382

[28] H. Kim, Y.S. Jeong, Sentiment classification using convolutional neural networks, Applied Sciences 9(11)(2019) 2347. https://doi.org/10.3390/app9112347

[29] S.W. Lai, L.H. Xu, K. Liu, J. Zhao, Recurrent convolutional neural networks for text classification, in: Proc. of the AAAI conference on artificial intelligence, 2015. https://doi.org/10.1609/aaai.v29i1.9513

[30] T. Bao, N. Ren, R. Luo, B.J. Wang, G.Y. Shen, T. Guo, A Bert-based hybrid short text classification model incorporating CNN and attention-based BiGRU, Journal of Organizational and End User Computing (JOEUC) 33(6)(2021) 1-21. https://doi.org/10.4018/JOEUC.294580