IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 June 2024, accepted 12 July 2024, date of publication 16 July 2024, date of current version 24 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3429150

APPLIED RESEARCH

A Hybrid Neural Network Model for Sentiment Analysis of Financial Texts Using Topic Extraction, Pre-Trained Model, and Enhanced Attention Mechanism Methods

GANGLONG DUAN, SHUNFEI YAN, AND MENG ZHANG
School of Economics and Management, Xi'an University of Technology, Xi'an 710054, China
Corresponding author: Shunfei Yan (1154620052@qq.com)

ABSTRACT In the financial field, texts such as news and commentaries, as carriers of public opinion, have the function of reflecting investor sentiment, influencing investment decisions and market trends, and extracting positive or negative sentiment from them in a timely manner is very important to the investment decisions of fintech companies and investors. However, financial sentiment analysis is challenging due to issues such as unclear sentiment polarity of financial texts, high context-dependency, and highly specialised and specific expressions of the linguistic features of financial texts. In order to overcome these challenges, we design a hybrid topic feature and pre-trained model financial text sentiment analysis model, and we design the method of improved attention mechanism to optimise the model iteratively, the improved attention mechanism reduces the noise caused by the improper allocation of attention to a single word through adaptive attention threshold, attention weight masking mechanism, so as to capture more contextual information and avoid the loss of information, thus improving the stability of the model. This improves the stability of the model. The introduction of thematic features enables the model to capture potential semantic structures and long-distance word associations in the text to enhance the understanding of text context. The experimental results show that the method has an improvement of 2.05%-7.27% in F1 value compared with the baseline method, which is suitable for financial text sentiment analysis.

INDEX TERMS Financial text analytics, sentiment analysis, progressive attention mechanism, FinBERT.

I. INTRODUCTION

Investor sentiment has received much attention as one of the key elements in the field of financial text research. The public is highly sensitive to the release of economic data, policy adjustments, and emergencies, which often lead to the formation of public sentiment and extensive discussion. In the financial field, as a carrier of public opinion, texts such as news and commentaries have the function of reflecting investor sentiment and influencing investment decisions and market trends, and irrational investor sentiment may lead to the deviation of financial asset prices from their fundamental values [1]. The market is not a place where the price of a financial asset can be deviated from its fundamental value. Therefore, how to be able to quickly and accurately mine the emotions in financial texts has become a hot research topic in recent years.

Sentiment analysis plays a crucial role in analysing, interpreting and extracting insights from financial data. For FinTechs, sentiment analysis has demonstrated a profound and transformative impact primarily on market sentiment monitoring and risk management [2]. Based on the theory of

The associate editor coordinating the review of this manuscript and approving it for publication was Shovan Barma.

behavioural finance, especially the in-depth study of investor sentiment, fintech companies use sentiment analysis to accurately distill and quantify the collective psychological state of market participants from multiple sources of information, such as social media, news reports, analysts' comments, and even corporate financial reports, so as to enable them to acutely identify potential risks in the fast-changing market environment, and to provide early warning of systemic risks. Through continuous tracking and analysis of market sentiment, fintech companies can build dynamic risk management frameworks, flexibly adjust risk management strategies according to real-time changes in market sentiment, and achieve optimal allocation of investment portfolios. At the same time, sentiment analysis technology is integrated into the decision support system, combining fundamental and technical analysis and macroeconomic data to provide comprehensive and accurate risk assessment and investment advice for fintech companies, significantly improving the efficiency and accuracy of risk management [3].

Another application of sentiment analysis in finance is in predicting market volatility, with academic research widely confirming that there is a significant positive correlation between market sentiment and stock prices, trading volume and market volatility [4]. Through sentiment analysis techniques, investors are able to monitor fluctuations in market sentiment in real time and anticipate price movements that may be triggered by changes in market sentiment in the short term. For example, a surge in negative sentiment on social media often signals a pullback or increased selling pressure, while a rise in positive sentiment may signal upward momentum. In addition, sentiment analysis can assist investors in identifying extremes in market sentiment, so-called ''sentiment bubbles'' or ''sentiment freezes'', which are potential signals of market reversals. By comparing historical data, investors can use sentiment analysis to determine whether current market sentiment is deviating from the normal range, so as to warn of potential market overreaction, take appropriate risk avoidance measures, or look for opportunities to invest against the trend [5].

In the face of the huge amount of financial text data emerging every day, how to extract valuable information from it has become a challenging research task for both academia and industry [6], [7] Sentiment analysis in finance is currently facing two main challenges. Currently, sentiment analysis in the financial domain faces two main challenges: first, the expression of sentiment in financial texts is often not as direct and explicit as texts in other domains. These texts may contain a large number of complex sentiment expressions, and even the same sentence or paragraph may contain both positive and negative sentiments, e.g., a report on a company's quarterly results may point to revenue growth (positive sentiment), but at the same time it may also mention rising costs or increased competition in the market (negative sentiment). In addition, some financial texts use implicit or indirect ways of expressing sentiment, e.g., ''Despite market challenges, the company achieved its goals'', where ''challenges'' denotes potentially negative sentiment, while ''achieved its goals'' is positive. Sentiment analysis in financial texts needs to be highly context-dependent, as the same word or phrase may express different sentiments in different contexts. For example, the word ''growth'' is a positive sentiment in the context of ''corporate profit growth'', but may be a negative sentiment in the context of ''debt growth''. Models need to be able to understand and differentiate between these different contexts in order to make accurate sentiment judgements. Secondly, there are many technical terms and specific expressions in the financial domain, which may not be accurately handled in a generic sentiment analysis model. For example, ''bull market'', ''bear market'', ''shorting'', etc. have specific sentiment meanings, which are likely to misjudge their sentiment tendencies if the model is not specially trained. Commonly used language in financial texts may contain complex syntactic structures and rhetorical devices, e.g. ''solid growth'' conveys a more positive sentiment than ''growth'' alone, while ''potential risks'' may need to be contextualised to determine its emotional polarity.

In order to solve the above problems, this paper proposes a hybrid neural network-based LFBP model (including the Topic analysis layer, Finbert layer, BiGRU layer, and Progressive attention mechanism layer) for sentiment analysis of financial texts. The model incorporates the advantages of multiple neural network models and can capture various features in the text more effectively. Specifically, the FinBERT pre-training model can learn the linguistic features of financial texts well, the bidirectional gated loop unit can better capture the local and temporal features before and after a long text, and the improved attention mechanism can more comprehensively take into account all the important information of the sentence after multiple rounds of iterative loops, thus improving the accuracy and robustness of the text sentiment classification. The proposal of this model brings new ideas and methods to the development of the field of financial text sentiment analysis, and provides new ideas and practical basis for research in related fields.

Specifically, our research makes major contributions in three areas:

(1) The standard attention mechanism is susceptible to strong sentiment words, causing the model to ignore low-frequency words or phrases in the context that may have important sentiment colors. We propose an improved training method for the attention mechanism, which can consider all the important information of a sentence more comprehensively through the strategy of dynamically adjusting thresholds set by text length and word frequency and the masking technique, thus improving the accuracy and robustness of financial text sentiment classification. We propose that this improved attention mechanism method can be applied to other domains of sentiment analysis by changing the training corpus, which provides a new idea and practical basis for research in related fields.

(2) A LFBP-based sentiment analysis model is proposed for sentiment analysis of financial texts and the model is compared with standard approaches. The experimental results show that our constructed model exhibits significant effectiveness on several benchmark datasets.

(3) The complexity of financial texts is often reflected in their jargon, industry-specific content, and diversity of contexts. We have innovatively designed a hybrid thematic analysis module to help understand the context of a text and extract key thematic information from it. In this way, the model is made to focus not only on the emotional polarity of the text, but to understand the financial information expressed in the text at a deeper level.

This paper is organised as follows. In Section II, we review the related literature. Section III describes the dataset used in this paper and explains the overall architecture of the LFBP financial text sentiment analysis model and the role of each module. We demonstrate the effectiveness of our approach in Section IV, where we show that it outperforms other state-of-the-art models. Specifically, we provide a discussion of evaluating the model's performance in a sentiment classification task on a financial dataset and validate it against the effectiveness of the various modules of the model proposed in this paper. Finally, Section V concludes with concluding remarks and suggestions for future research directions.

II. RELATED WORKS

In this section, we provide a brief introduction to financial texts and their most popular sentiment analysis methods. Based on the core idea behind these methods, we classify them into three categories: lexicon and rule-based methods, machine learning-based methods, and deep learning-based methods. In the next subsections, we also discuss the classical evolution of topic models and pre-trained models. Finally, we give the evaluation metrics used in this paper.

A. FINANCIAL TEXTS

Financial data can be broadly categorised into three types: image-based, numerical and text-based. Text-based data include financial commentaries, research reports, current news and company announcements. From an academic perspective, financial text can be defined as a kind of information carrier that is specifically related to the financial field, and its content revolves around financial markets, financial institutions, financial instruments, economic indicators, financial data, regulatory policies, and related economic activities. Such texts not only carry historical data and current status of financial markets, but also contain forecasts and analyses of future market dynamics [8]. The main characteristics of financial texts include: specialisation, real-time, data-intensity, diversity and predictability. In academic research, financial text analytics is an important research area that involves the use of quantitative methods and natural language processing techniques to extract useful information from texts in order to analyse the behaviour of financial markets, predict changes in economic indicators, assess the financial position of a company, and so on. Research in this area not only helps to understand the workings of financial markets, but also has important practical applications for investors to formulate strategies, governments to formulate policies, and financial regulators to carry out supervision [9]. This paper focuses on the study of financial text information. This paper focuses on the investor sentiment embedded in financial text information to explain investor behaviour in the market. Sentiment analysis of such financial text data is necessary because investors are affected by factors such as the level of investment knowledge and attention, as well as psychological cognitive biases such as overconfidence, the follow-the-leader effect, and cognitive bias, which may produce irrational biases in their expectations of the future and lead to the emergence of abnormal investment behaviours.

B. SENTIMENT ANALYSIS OF FINANCIAL TEXTS

Text sentiment analysis methods in finance can be categorised into dictionary-based methods, machine learning-based methods [10], [11]. Among them, machine learning methods can be further divided into traditional machine learning and deep learning [12], [13]. Deep learning is developed on the basis of traditional machine learning, which has developed rapidly in recent years and has gradually become a research field different from traditional machine learning. A detailed comparison of the three sentiment analysis methods is shown in Table 1.

Dictionary-based approach is to match the identified words in the document with the words in the sentiment dictionary, and then determine the text sentiment polarity based on the semantic direction of the words and phrases or the calculated attitudinal tendency value. Currently, the more authoritative and widely used sentiment dictionaries or corpora are HarvardIV-4 Sentiment Dictionary [14] The Loughran-McDonald Financial Sentiment Dictionary [6], [15], [16], SentiWordNet [17], SenticNet [18]. Price et al. [19] found that topic-specific or contextually relevant dictionaries were better used than general-purpose dictionaries. Caporin and Poli [20] constructed a specialised database of news sentiment metrics based on news from two mainstream financial media outlets in the US and for the S&P 100 index constituents. The shortcomings of dictionary-based methods include: ignoring the order and grammatical structure of word occurrence, resulting in bias or even deviation from the understanding of the text content; word weighting scheme varies from person to person and is sensitive to contextual content, and the dictionary cannot be automatically adapted to reflect the changing contexts and semantics; manually constructing the dictionary is time-consuming and laborious, and it is mostly used to formulate the trade-offs and classifications for specific domains, which is applicable to a small range of domains and poorly disseminated. Nevertheless, dictionary-based methods are widely used in financial text sentiment analysis because they are easy to understand and simple to operate. Especially for short texts with weak

TABLE 1. Comparison of three methods of sentiment analysis.

Comparative dimension	Dictionary-based approach	Machine learning based approach	Deep learning based approach
Basic principle	Lexical matching	Feature representation, performance evaluation, model optimisation	Neural Networks, Model Optimisation
Difficulty of use	Simple	relatively simple	intricate
Advantages	Easy to understand and simple to operate	Can be directly applied to mature models, convenient and efficient, saving time and effort; more accurate capture of text semantics; can handle text big data	Feature extraction and selection without excessive human intervention; suitable for large datasets
Drawbacks	Prone to losing and misinterpreting part of the textual information; time-consuming and labour-intensive manual construction of dictionaries; high lexicon-specificity	Model goodness is highly dependent on the quality and quantity of annotations in the training dataset	Involves a lot of computer knowledge, which makes it difficult to get started quickly; long training time
Scope	Shorter texts with weak contextual links	Text with relatively small amounts of data	Data volume of large text, targeted training for different text characteristics

contextual links, the accuracy of dictionary-based methods is comparable to that of machine-learning-based methods in classifying text sentiment [21].

The shortcomings of the sentiment lexicon approach have prompted some scholars to develop statistical techniques based on traditional machine learning that can learn complex features from data and improve the process of sentiment analysis. Traditional machine learning has been used earlier in sentiment analysis, and through literature combing, the more popular traditional machine learning models are Support Vector Machine (SVM) and Naive Bayes (NB) [11], different machine learning models perform differently in classifying financial texts in terms of sentiment polarity.

In the field of financial text sentiment analysis, the deep learning models that are used more often are Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) [22] The RNN that has attracted a lot of attention is the Transformer, which further improves the overall classification performance by performing Sec2Sec transformations using encoders and decoders. For example, Devlin et al. [23] demonstrated that BERT exhibits unique advantages in all 11 NLP tasks, while Zhao et al. [24] RoBERTa model outperforms the BERT model on sentiment analysis tasks by a narrow margin. Mishev et al. [25] compared the sentiment classification effectiveness of various Transformer family models including BERT, RoBERTa, BART, etc. based on two manually labelled financial news datasets, and the results showed that the BART model was optimal. Deep learning models [26], [27], [28], [29], [30], [31], [32], [33] are more complex and perform better than traditional machine learning models, in the following ways: in traditional machine learning methods, the process of feature extraction is very time-consuming, but deep learning is able to automatically create the features needed for the classification process, and the extraction and selection of features require little human intervention; for large datasets, traditional machine learning cannot be performed, while deep learning models can be trained to be better; deep learning's multi-layer architecture is more effective. better; deep learning's multi-layer architectural model can capture non-linear relationships and even more complex features in the data very well [34]. The Common datasets in the financial domain today are the Financial Sentiment Analysis Challenge Dataset (FiQATask1) [35], SemEval2017 Task5 [36], FinancialPhraseBank [37], StockSen [38], SEntFiN [39] But BERT, RoBERTa, ALBERT [40] ERNIE [41] and many other Transformer family models, including BERT, RoBERTa, ALBERTa, ERNIE, etc., have not been trained for the financial domain, so there are some compatibility problems when performing financial domain text sentiment analysis, such as: financial domain-specific linguistic features, unclear sentiment polarity, context-dependent, and contextual relevance. Therefore, this study adopts a novel model architecture of FinBERT, a pre-trained model for the financial domain, a hybrid neural network and an improved attention mechanism. The FinBERT pre-training model can learn the linguistic features of financial texts well, the bidirectional gated loop unit can better capture the local and temporal features before and after the long text, and the improved attention mechanism can more comprehensively consider all the important information of the sentence after multiple rounds of loop iterations, so as to improve the accuracy and robustness of the text sentiment classification.

C. TOPIC MODELS

A topic model (TM) is a generative probabilistic model for automatically extracting implicit semantic topics from unstructured data, commonly used in large-scale corpora and discrete data modelling. The model understands the documents in the corpus as a distribution of specific implicit topics, and thus abstract topics can be discovered according

to the implicit semantic features and represented in the form of word lists. The most popular topic model is the one developed by Blei et al. [42] The latent Dirichlet allocation model (LDA) proposed by Blei et al. has been used in many fields such as text categorisation, anomaly detection, recommender systems, text summarisation, viewpoint extraction, lexical summarisation, sentiment analysis, information retrieval, etc. [43], [44]. It has been widely used and rapidly developed. However, since the topics are in the form of word lists, it usually causes some obstacles for users to understand them correctly. Especially when the user lacks the relevant background knowledge of the subject area [45], their understanding of the topic may be fragmented, one-sided and inaccurate. In the financial domain, themes often cover important topics such as stocks, market trends, and company performance. By incorporating this thematic information into a sentiment analysis model, we are able to understand the meaning of the text in a more comprehensive way. Sentiment analysis is no longer just a single categorisation of sentiment, but is closely related to the topics covered in the text, allowing for an increase in the dimensionality of the comprehensive information. The complexity of financial texts is often reflected in their jargon, industry-specific content, and diversity of contexts. Topic feature extraction techniques can help understand the context of a text and extract key topic information from it. These thematic information is the core content contained in financial texts, such as stock market dynamics, company financial performance and so on. Combining thematic features with a sentiment analysis model, through which the sentiment analysis model does not only focus on the emotional polarity of the text, but also understands the financial information expressed in the text at a deeper level, thus improving the accuracy and reliability of the analysis results.

D. EVALUATION METRICS

In this study, we use the confusion matrix to validate the effectiveness of the LFBP sentiment analysis model. We used metrics such as accuracy and F1 score to assess the accuracy of the model and to compare its performance with other models. In particular, the elements of the confusion matrix include true positives (TP, predicted positive and actually positive), false positives (FP, predicted positive but actually negative), false negatives (FN, predicted negative but actually positive), and true negatives (TN, predicted negative and actually negative). In addition, we defined the actual value, predicted value and the mean of the values.

$$F_1 = 2PR \big/ (P + R) \tag{1}$$

$$A = (TP + TN) \big/ (TP + TN + FP + FN) \tag{2}$$

III. MATERIALS AND METHODOLOGY

In this section, there are two subsections. In the first subsection, information about the dataset used in this paper is presented. In the second subsection, the overall architectural design of the LFBP model proposed in this paper is firstly presented, followed by the presentation of the model's topic analysis layer, FinBERT layer, BIGRU module, and the iterative approach to optimise the model training with the improved attention mechanism proposed in this paper.

A. MATERIALS

Five financial text sentiment analysis datasets were used in this study, namely the Financial Sentiment Analysis Challenge dataset (FiQATask1), SemEval 2017 Task5, FinancialPhraseBank, StockSen, and SEntFiN. where the first two datasets were used to train the models and the last three datasets were used to evaluate the model performance.

The Financial Sentiment Analysis Challenge dataset (FiQATask1) consists of two types of data: financial news headlines and financial tweets, both of which come with manually labelled target entities and sentiment scores. The financial news headlines dataset contains a total of 529 annotated headline samples, while the financial tweets dataset contains 774 annotated post samples. The dataset contains texts related to finance, economy and markets, such as news articles, social media posts, financial reports and analysts' comments.

SemEval 2017 Task 5 is a fine-grained FSA for news headlines and tweets. The training dataset consists of 1,142 financial news headlines and 1,694 posts with target entities and corresponding sentiment scores as shown below. The test dataset consists of 491 financial news headlines and 794 posts. The task is to extract and detect the target and its corresponding sentiment score.

The FinancialPhraseBank dataset contains 4,846 English-language sentences from randomly selected financial news texts from the LexisNexis database, which were annotated and processed by 16 experts with backgrounds in finance and business. The dataset contains sentences extracted from news and other finance-related sources that were manually labelled with sentiment polarity, including positive, negative and neutral. The dataset was designed with the specificity and complexity of the financial domain in mind, and thus contains specialised terms and expressions related to market dynamics, company performance, economic indicators, etc.

The SEntiFin dataset contains 10,753 annotated news headlines for entities and their associated financial sentiments. Of these, there are 2,847 news headlines with more than two entities, including 1,233 news headlines with conflicting emotions. News headlines with multiple entities contributed 6,500 entity sentiment annotations with an average of 2.3 entities per headline. The average word length of sentences across the dataset was 9.91 words, and sentences with multiple entities tended to have a higher average word length of 10.39 words. SEntFiN identified 14,404 entities and their sentiments, of which 35.23%, 26.48%, and 38.29% were positive, negative, and neutral sentiments, respectively, indicating a low class imbalance.

The StockTwits sentiment (StockSen) dataset is derived from the StockTwits platform, a social networking service for

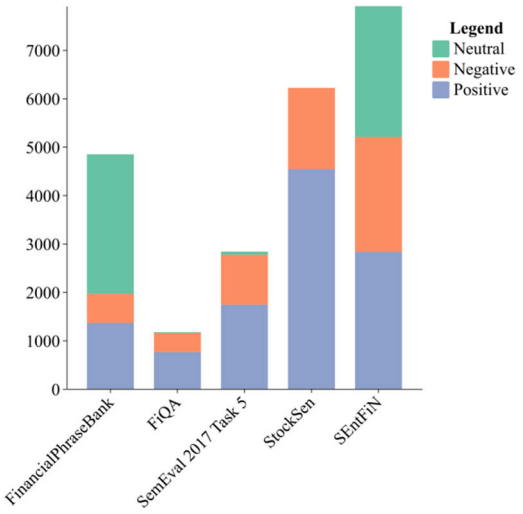

FIGURE 1. The structure of the dataset.

TABLE 2. Statistical information on the dataset.

Dataset	Positive	Negative	Neutral	total	Purpose
FinancialPhraseBank	1363	604	2879	4846	*Evaluation*
FiQA	760	399	14	1173	*Training*
SemEval 2017 Task 5	1739	1032	65	2836	*Training*
StockSen	4542	1676	-	6218	*Evaluation*
SEntFiN	2832	2373	2701	7906	*Evaluation*

stock and financial market participants that allows users to share their opinions, analyses, and investment strategies on specific stocks. Users can post "Twits" (similar to Twitter tweets) on the platform, which can be labelled as Bullish, Bearish or Neutral.The StockSen dataset consists of 55,171 texts accessed from the StockTwits platform accessed from 55,171 entries dated 2019-06-06 to 2019-08-26. After filtering out entries without self-labelled sentiment, we obtained 20,675 financial tweets (labelled as positive or negative), of which a total of 6,218 tweets (4,542 positive and 1,676 negative) were randomly selected for testing. We manually checked the self-labels to confirm that they were of high quality. The statistics of the dataset are shown in Figure 1 and Table 2.

B. METHODOLOGY

1) LFBP MODEL ARCHITECTURE DESIGN

The structure of the financial text sentiment analysis model based on LFBP is depicted in Figure 1, where the input is preprocessed financial text data. The model contains the following key components: topic feature analysis layer, FinBERT layer, feature learning layer and Softmax layer. Specifically, the preprocessed financial text data is first input to the topic feature analysis layer and FinBERT pre-training model, and then the topic layer data is spliced with the preprocessed data, which is then integrated to obtain the semantic feature F. Next, the obtained temporal feature F is feature extracted by the bi-directional GRU layer, and the improved post-attention mechanism is used to extract the temporal feature F through the dynamic threshold adjustment and attention weight masking mechanism, which is able to identify and filter the words that have misleading influence on sentiment classification, and ensure that the model focuses on other important contextual information, and after several iterations of training, through the dropout layer and the fully-connected layer, the final feature vector H is obtained. Finally, the obtained feature vector H is processed through the Softmax layer, so as to obtain the classification results of the sentiment tendency.

2) THEMATIC ANALYSIS LAYER

Topic feature extraction is to extract topic information from text and represent it as $n \times 768$ topic vectors by using LT model. This method can effectively use the probability distribution property of LDA model to mine the implicit topic information of the text and transform it into vector representation, so as to enhance the expression ability of text features. At the same time, the adaptive extraction of theme information is realised, and it is spliced with other features in FinBERT layer, so as to enhance the fusion ability of text features. The extraction of theme features relies on the fusion algorithm of LDA and K-means. Among them, LDA is a three-layer Bayesian statistical model that assigns a topic to each document in the dataset in the form of a probability distribution, and it is capable of mining the implicit topic information of the text. Its principle is to assume that each document consists of multiple topics, each topic consists of multiple words, the model iteratively updates the distribution of words under each topic and the distribution of topics under each document, and finally obtains the probability that each word belongs to a different topic and the probability that each document contains a different topic. Therefore, in order to make LDA output the topic vectors of documents, this paper makes the following improvements to LDA: expand LDA for clustering into a topic model that can generate document vectors, and no longer use the clustering results of the LDA model; modify the Dirichlet distribution of the "document-topic" to take the topic as the dimension, and the probability value as the value on the dimension. We modify the Dirichlet distribution of "document-topic" to take the topic as the dimension and the probability value as the value on the dimension, so as to get the vector representation of the document on different topic dimensions; we combine the LDA topic model and the K-means algorithm to enhance the extraction capability of the model on the topic; and we splice the topic vectors of the document into a matrix of size $n \times 768$.

The outermost layer of the improved LT model is the document collection layer, followed by the document layer and the word layer. Among them, a single circle represents a latent variable, a rectangle indicates repeated sampling (the letter

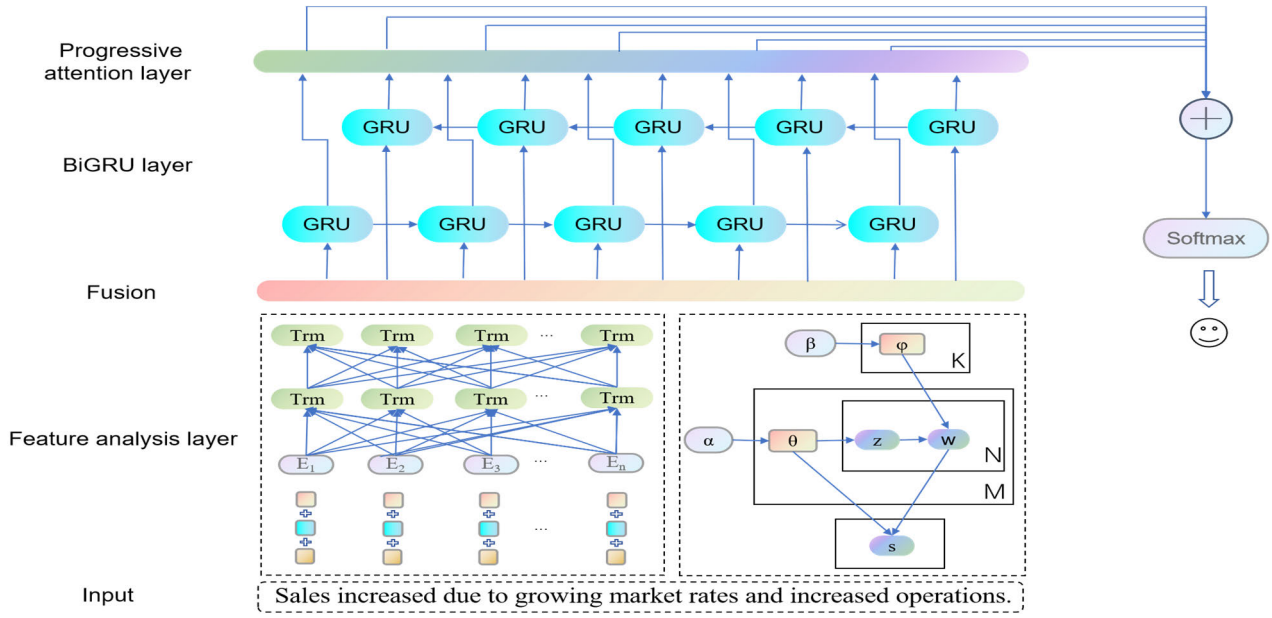

FIGURE 2. Structure of financial text sentiment analysis model based on LFBP.

in the lower right corner is the number of repeated sampling), and an arrow indicates the dependence of conditional probability between two variables. The topic distribution of the document and the word distribution of the topic are θ and φ, respectively, which obey the a priori Dirichlet distribution with corpus level parameters α and β. The document collection is D, the sentence in each document is d, the number of sentence is i, the number of Chinese characters in each sentence is N, and the number of Chinese characters is n. The implied topic share assigned to the word in each document is z, the topic word in the text is w, each word has a potential topic, and the number of topics is N, and the number of characters in the text is n. The implicit topic share assigned to the word in each document is z, and the topic word in the text is w, each word has a potential topic, and the topic number of the topic is n. The topic share assigned to the word in each document is z. a potential topic,the number of topics is K,and the topic distribution of its document is shown in equation (3) as follows.

$$P(w, z, \theta | \alpha, \beta) = P(\theta | \alpha) \prod_{n=1}^{N} P(z_n | \theta) P(w_n | z_n, \beta) \tag{3}$$

The basic logic of the K-means clustering algorithm is to first define the number of centres of mass C, and then objects are assigned to the nearest centre of mass in a continuous loop. At each step, C new centres of mass need to be recalculated, and then the objects are reassigned until no more changes are made. In the improved algorithm of this paper, the centre of mass is the subject, so C = K. In order to improve the accuracy of LDA in extracting the subject information, this paper integrates K-means clustering algorithm into LDA algorithm, and θ is used as the input of the K-means clustering

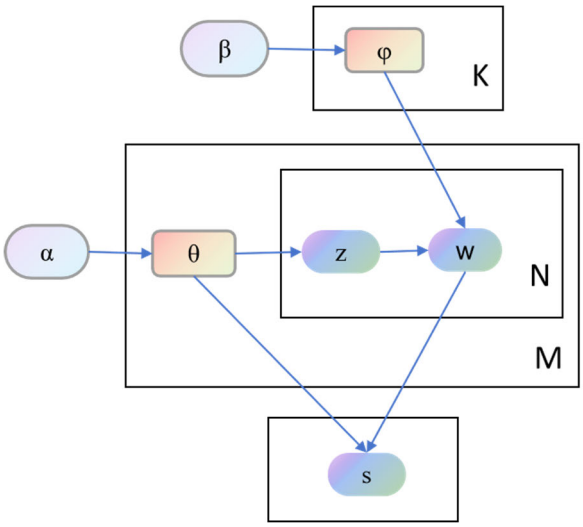

FIGURE 3. The structure of LT topic analysis layer.

algorithm. The specific structure of the LT model is shown in Figure 3.

The model parameters are iteratively updated by the Gibbs sampling algorithm, and when the model converges, we can obtain the topic distribution of each document and normalise it so that it sums to 1. In this way, each document can be represented by a K-dimensional probability vector, where K is the number of topics, and each element of the vector corresponds to the probability of a topic. This probability number is a multi-dimensional vector, where each dimension corresponds to a topic. The PCA algorithm is used to downscale the high-dimensional data, and the K-dimensional topic vectors of each document are projected into a low-dimensional space to

TABLE 3. The extraction method of the theme feature vector.

Algorithm Extraction of subject vectors
Input:Document D
Parameters: "document-topic" distribution θ, number of topics K
Output:Subject vector s
M←LDA(D,K)
centroids,labels←K-means(M,K)
topic_vectors←[]
for i in range(K):
indices←[j for j in range(len(labels)) if labels[j]= =i] and
rows ←[θ[j] for j in indices] and mean←average(rows) and
topic_vectors. append(mean)
s←matrix(topic_vectors)
return s

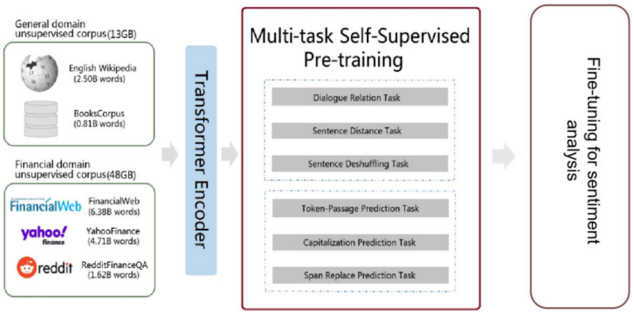

FIGURE 4. The pre-training process of Finbert.

obtain a two-dimensional matrix s. s is a topic embedding of n × 768, which represents the topic information of the text, and is used to assist the encoder to learn the topic information of the text. The extraction method of the theme feature vector is shown in the following Table 3.

3) FinBERT LAYER

FinBERT was created by Liu et al. [34] A pre-trained model based on BERT was proposed and trained using a large general-purpose financial corpus. FinBERT first trains the language model on the TRC2-financial corpus, and then uses its weights to initialise the financial commentary sentiment analysis model, as shown in the following Figure 4.

In order to extract the semantic features of financial text more effectively, the pre-processed financial text data are firstly embedded using FinBERT pre-training model, i.e. token embedding, Segmentation embedding and Position embedding.

(1) Token embedding: turning words into fixed dimensional vectors;

(2) Segmentation embedding: aids Finbert in distinguishing the vector representation of two sentences in a sentence pair;

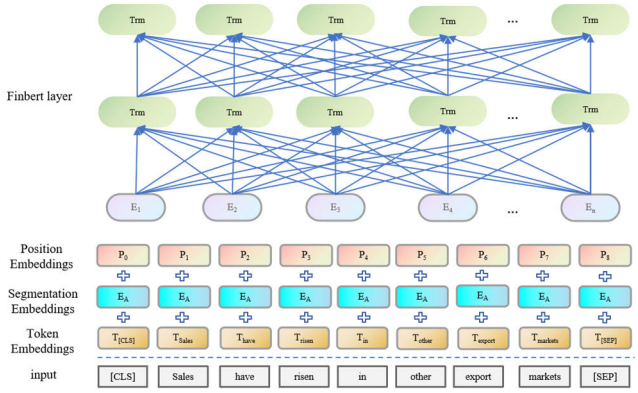

FIGURE 5. The structure of FinBERT layer.

(3) Position embedding: assists Finbert in learning the sequential properties of the input.

Finally, as is shown in equation (4) as follows, the sum of the 3 vectors is used as the model input, which can retain more valid information and affective tendencies for the subsequent tasks, as shown in the following Figure 5

$$E_n = T_n + S_n + P_n \tag{4}$$

where E_n represents financial text data embedding; T_n represents token embedding; S_n is the Segmentation embedding; P_n is the Position embedding.

Subsequently, the financial text embedding E is dimensionally transformed into a tensor of size of the tensor, where is the sample size selected for one training, and is the maximum length of the financial text. Then, the financial text data embedding E is used as an input to the FinBERT pre-training model. The coding layer consists of 12 Transformer layers stacked on top of each other, which aims to learn the deep semantic features of the text. Each Transformer layer contains a multi-head self-attention mechanism and a fully-connected feed-forward neural network. Each Transformer layer contains a multi-head self-attention mechanism and a fully-connected feedforward neural network. The multi-head self-attention sublayer can calculate the correlation between any two words and reduce the distance to 1. In order to solve the problem of memory bias in the process of information transfer, the sequences are sequentially fed into residual connection and layer normalisation. The residual linkage can make the model easier to learn the constant mapping, thus reducing the training difficulty. The inputs to the model are then summed with the outputs of the previous layer. Layer normalisation can make the model pay more attention to the difference information, improve the expressive ability of the model, and solve the problem of gradient explosion in the process of information transfer.

The text data is first transformed into sequence vectors through the embedding layer, and then summed with the position vectors to get the input to the Transformer encoder. The input sequence is mapped into key matrix (K,keys), value matrix (V,value) and query matrix (Q,query) through

three linear transformations respectively into the multi-head attention sub-layer, and the correlation weight between each word/phrase and other words/phrases is calculated. The number of heads of the multi-head attention mechanism is h. Each Transformer encoder in this model contains 12 multi-head attention sublayers. The weight matrix is. The output of the multi-head attention layer is shown in equation (5) as follows.

$$MultiHead\,(Q, K, V) = Concat\,(head_1, \ldots, head_h)\,W^M \tag{5}$$

The model uses Softmax function to calculate the weight coefficients of one word/phrase to other words/phrases, and uses the syntactic structure and semantic information learnt from the coding layer to classify the textual sentiment. Using as a reconciliation factor to stabilise the training gradient of the module, the purpose is to prevent the inner product from being too large, to assist the model to capture the relevance of the data, and to solve the problem of long-distance dependence of traditional neural networks. In the model, the output of the self-attention sublayer is shown in equation (6).

$$Attention\,(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{6}$$

The output of the first AddNorm layer is used as the input to the feed-forward neural network (FFN), and the output of the FFN is passed through the next AddNorm layer. nx is the number of layers in the stack, and Finbert used 12 layers in total. the Transformer encoder network extracts text features in parallel, and then maps the text features to the sample markup space through the fully-connected layers, obtaining the corresponding representation vector T. The specific structure of the Transformer encoder in the encoding layer is shown in Fig. The Transformer encoder network can extract text features in parallel, and then map the text features to the sample markup space through the fully connected layers, and get the corresponding representation vector T. The specific structure of the Transformer encoder with coding layer is shown in Figure 6.

By summing the elements of the vector representation output from the embedding layer, a two-dimensional matrix E of shape n × 768 is obtained as the input representation of the coding layer; E is the weighted vector of word embedding, position embedding, and segment embedding of the ith token; N is the length of the token sequence, with the maximum not exceeding 512; and this is the input representation passed to the encoder layer based on the Transformer. The vector E is mapped to the hidden layer dimension through the decomposition embedding parameterisation mechanism, and is used as the input of the first encoder; the output of each encoder is used as the input of the next encoder; the output of the last encoder is the representation vector T of each word/phrase in the text after integrating the full text semantic information.

4) BiGRU LAYER

Gated Recurrent Unit (GRU) is a more effective variant of the LSTM model, using two ''gates'': r_t reset gate and z_t update

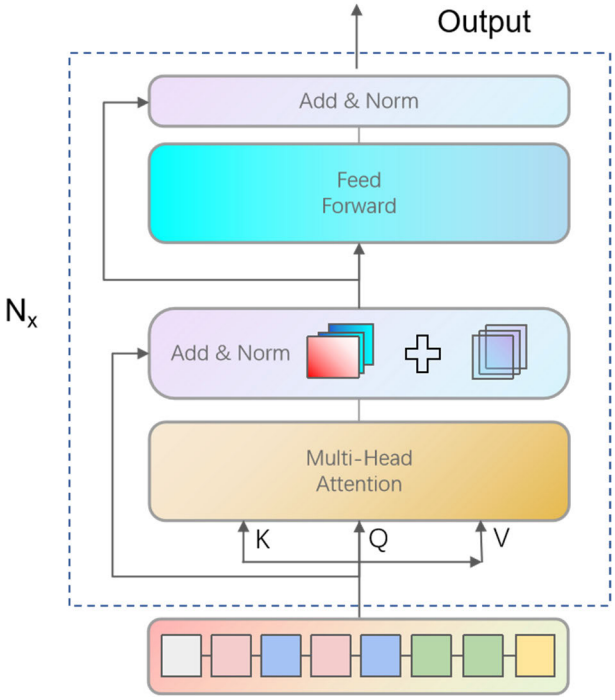

FIGURE 6. The structure of coding layer module.

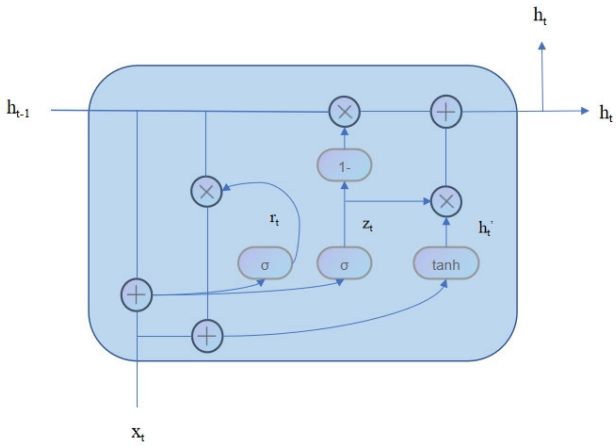

FIGURE 7. The structure of the BiGRU module.

gate to memorise the information, using x_t to represent the input data, ht to be the output of the GRU unit, and h_t to be the candidate hidden state, as shown in in equation (7)-(10) and Figure 7.

$$z_t = \sigma\,(W_z x_t + U_Z h_{t-1} + b_z) \tag{7}$$
$$r_t = \sigma\,(W_r x_t + U_r h_{t-1} + b_r) \tag{8}$$
$$h_t = (1 - Z_t) \odot h_{t-1} + Z_t \odot \tilde{h}_t \tag{9}$$
$$\tilde{h}_t = tanh\,(W_h x_t + U_h\,(r_t \odot h_{t-1}) + b_h) \tag{10}$$

The topic vectors from the topic analysis module and the semantic vectors obtained from the FinBERT pre-trained model are input into the BiGRU model. In recurrent neural networks, the state transfer is usually from forward to

backward, however, the current output state is not only related to the previous state, but also related to the subsequent state, therefore, this paper uses the bidirectional GRU model to obtain the text information from the forward and backward direction at the same time. BiGRU consists of two GRUs superimposed on each other backward and forward, and the output is determined by the state of the two GRUs together, so that the final output information is as follows. $h_t^{(i)} = \left[\overrightarrow{h_t^{(1)}}, \overleftarrow{h_t^{(1)}}\right]$ BiGRU consists of two GRUs superimposed on each other, and their states determine the output. Here $h_t^{(i)}$ denotes the BiGRU information of the ith English text, and $\overrightarrow{h_t^{(1)}}$ denotes the forward GRU information of the i-th text, and $\overleftarrow{h_t^{(1)}}$ denotes the backward GRU information of the ith text. Specifically, after inputting the topic vectors and the semantic vectors obtained from the FinBERT pre-training model into the BiGRU model, the hidden states at each time step are computed by processing them sequentially from the first word to the last word in the forward GRU direction. For example, at time step t = 5, the forward GRU receives the feature vector of the word ''quarterly'' and the hidden state of the previous time step to calculate the current hidden state. Meanwhile, in the reverse GRU direction, the hidden state is calculated at each time step by processing the words sequentially from the last word to the first word. The reverse GRU receives the feature vector of the word ''rise'' and the hidden state of the next time step to calculate the current hidden state. The reset and update gates control the flow of information and state update based on the input and the hidden state of the previous time step. For example, when processing ''quarterly'' in a forward GRU, the reset gate may choose to ignore irrelevant historical information, while the update gate decides which information to keep for the next step. Based on the current input and the hidden state of the previous time step, the hidden state of the current time step is calculated by the gating mechanism. For example, when processing ''quarterly'', the forward GRU generates a hidden state that combines the state of the previous time step with the characteristics of the current input. The output of the bi-directional GRU splices the forward and backward hidden states of each time step to form bi-directional contextual features for each word. The resulting feature matrix dimension contains the bi-directional contextual features for each word in the input text.

5) PROGRESSIVE ATTENTION MECHANISM

The standard attention mechanism is susceptible to the influence of strong emotional words, resulting in the omission of other key contextual information. For example, in the standard attention mechanism, the model may pay too much attention to words with strong emotional colours, such as ''increased'' and ''declined sharply'', and the following Figure 8 shows the standard attention weight distribution.

Here, attention is mainly focused on ''increased'' and ''declined''. In this case, if we focus only on these high-weight words, we may overlook important contextual

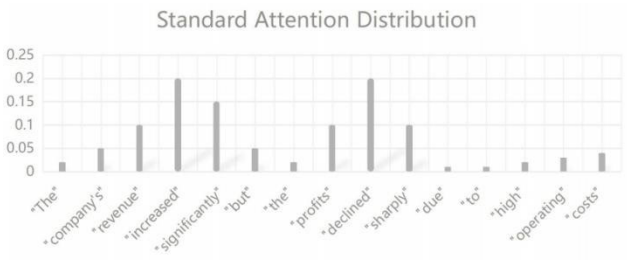

FIGURE 8. The standard attention weight distribution.

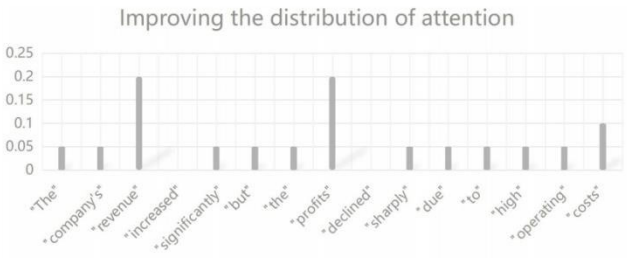

FIGURE 9. The improved attention weight distribution.

information in the sentence, such as ''revenue'', ''profits'', and ''high operating costs'', which are crucial to understanding the overall financial health of the sentence.

The improved mechanism, through dynamic adaptation and self-supervised learning, is able to identify and filter words that have a misleading impact on sentiment categorisation, ensuring that the model focuses on other important contextual information. For example, in the initial stage, the model may recognise ''Increased'' and ''Declined'' as highly weighted words. Through self-supervised learning, if it is noticed that these words overly influence the classification results during the classification process (especially when the classification is incorrect), these words are added to the set of misleading words Misleading Words After masking ''increased'' and ''declined'', the model reallocates attention and may pay more attention to words like ''revenue'', ''profits'' and ''high operating costs'', the Figure 9 below shows the improved attention weight distribution.

Instead of relying only on the strong emotion words ''increased'' and ''declined'', the improved attention mechanism dynamically adjusts the attention weights so that the model can better capture the overall financial information of the sentence, i.e., although ''revenue increased significantly'', ''profits declined sharply'' because of ''due to high While ''revenue increased significantly'', ''profits declined sharply'' due to ''due to high operating costs''. In this way, the classification result of the model will be more accurate because it considers all the important parts of the sentence, thus improving the classification accuracy and robustness of the model.

We are improving the attention mechanism by: we define two sets of words Q_p, Q_n, the Q_p in which we store those words that have a positive impact on semantic prediction,

and. Q_n in stores the words that have misleading influence on semantic prediction, and the attention weights of these words should be reduced.

(a) Generative word representation

Generate a category representation:

$$A = f_a(\theta) \tag{11}$$

based on Q_p an Q_n The previously extracted words are masked to obtain X′:

$$X' = X \setminus \left(Q_p \bigcup Q_n\right) \tag{12}$$

Generated word representation:

$$H = f_{word\_representation}\left(X'\right) \tag{13}$$

(b) Predicting categories and calculating attention weights;

Prediction categories to get the prediction results:

$$\hat{y} = f_{prediction}(H) \tag{14}$$

Calculate the attention weight for each word:

$$\alpha = f_{attention}(H) \tag{15}$$

(c) Dynamically adjusting thresholds, filtering keywords, updating and;

Calculate the expectation of the attentional weights for all words:

$$E(\alpha) = \frac{1}{|H|}\sum\nolimits_{h \in H} \alpha(h) \tag{16}$$

Setting Dynamic Thresholds Determining the Presence of at Least One Word with a Significant Impact on Sentiment Classification Based on a Dynamically Adjusted Threshold:.

$$threshold = f(length, frequency) \tag{17}$$

$$if\ E(\alpha) < threshold, continue \tag{18}$$

$$else, extracth_{max} = argmax_{h \in H}\alpha(h) \tag{19}$$

Update the sentences according to whether they are classified correctly Q_p and Q_n:

$$if\ sentence\ is\ correctly\ classified,\ h_{max}\ is\ put\ into\ Q_p$$
$$else, h_{max}\ is\ put\ into\ Q_n$$

(d) Combine X′, t, y as a tuple and update the parameters.

Combine X', t, y as a tuple and combine with the correct result to form a new training corpus $D(k)$:

$$D(k) = \left\{\left(X', t, y\right), (correct\ result)\right\} \tag{20}$$

The parameters are updated in the next iteration using the D (k) to update the parameters in the next iteration. The entire process is shown in the following Figure 10.

The text features are further extracted by the above proposed improved Attention mechanism, and after calculating the relevant result values of the Attention layer, a Dropout layer is added between the Attention layer and the fully connected layer in order to avoid the occurrence of overfitting phenomenon. The input of the output layer is the output of the fully connected layer. The softmax function is used to calculate the input of the output layer in a corresponding way so as to carry out sentiment classification, the equation (21) is as follows:

$$\hat{y} = softmax(wv + b) \tag{21}$$

where w denotes the matrix of weight coefficients to be trained from the Attention mechanism layer to the output layer, and b denotes the to-be-trained bias, and $\hat{y}$ is the prediction label of the output.

TABLE 4. Experimental environment.

Matrix	Configure
Development language	Python 3.9
Algorithmic framework	Pytorch
Ide	Pycharm

IV. EXPERIMENTAL METHODS AND RESULTS

In this section, there are four subsections. In the first subsection, the dataset is divided into training set, validation set and test set. In the second subsection, the experimental platform environment and the final experimental parameter settings are determined through multiple sets of parameter comparison experiments are described. In the third subsection, the comparison results of LFBP with other sentiment analysis models such as BiGRU, BiLSTM-Att, BERTbase, ALBERTbase, RoBERTa, ERNIE, FinBERT are given to validate the effectiveness of LFBP, and the effectiveness of the topic vectors and the improved attention mechanism methods are experimentally verified. In the last section, the validation of the effectiveness of each module of the LFBP model is given through ablation experiments.

A. DATA PRE-PROCESSING

Before training, the dataset needs to be pre-processed by word splitting and de-duplication. Then, the datasets were randomly sampled according to the ratio of 6:2:2, and the datasets were divided into the training set, validation set and test set.

The purpose of doing so is to ensure the representativeness and independence of the datasets, so as to evaluate the classification effect of the model more objectively.

B. ENVIRONMENT AND PARAMETER SETTINGS

The experiments in this paper were conducted on NVIDIA3060GPU for training and testing work, the specific information of the experimental platform is listed in the Table 4 below.

In the experiments, factors such as batch size, number of model training sessions, and discard rate all have an impact on the results. In order to explore the optimal parameter settings in the model, three comparison experiments are designed in this section. The initial values of the main parameters in the experiments are as follows: the sentence length is 128; the

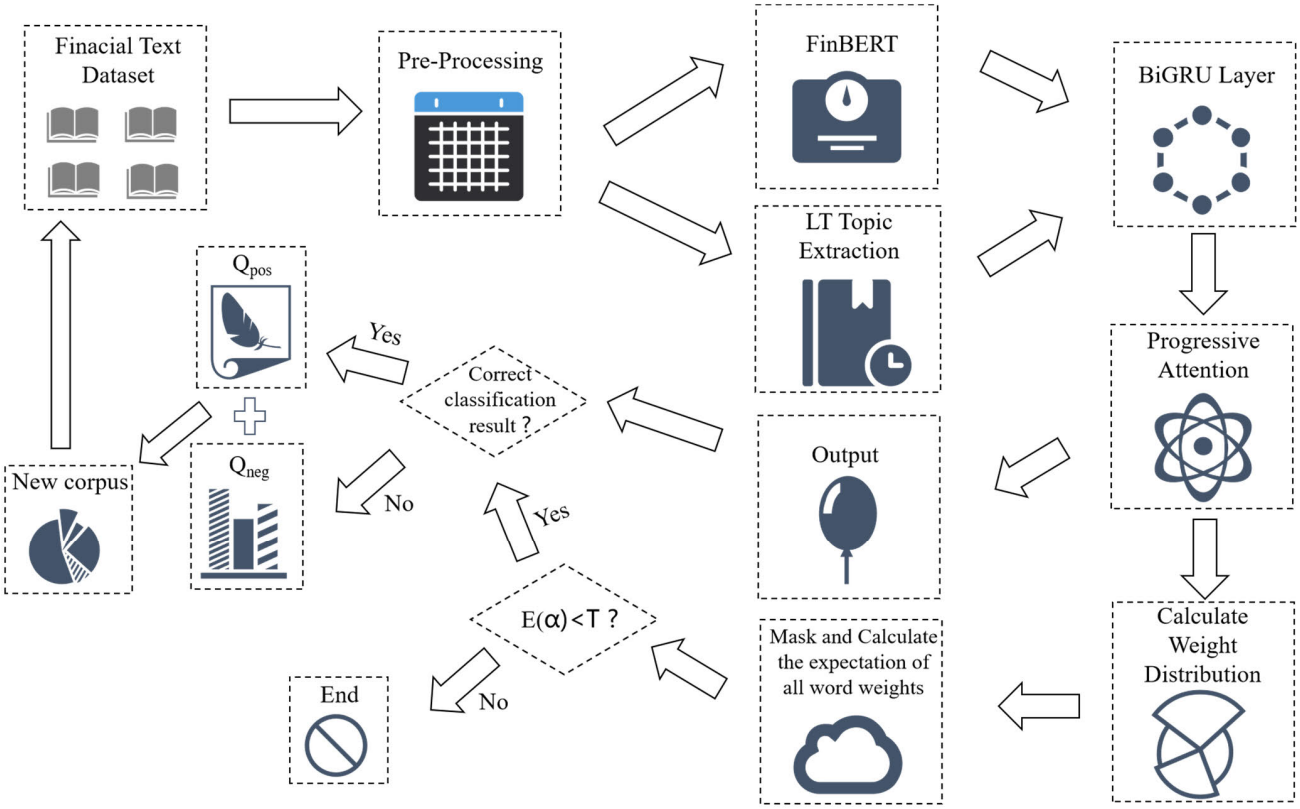

FIGURE 10. The process of progressive attention mechanism.

dropout rate is 0.5; the number of training times is 10; the learning rate is; the optimiser is Adam.

As is shown in the Table 5, when the batch size is set too small, the accuracy F1 value of the model is low, the loss function oscillates violently, and the gradient of the backpropagation changes too much, which leads to difficulty in convergence. ncreasing the batch value can speed up the processing of the same data, and the parameters of the model reached the optimum when increasing the batch value to 64.

Continuing to increase the batch value leads to the need for more memory and computational resources, and the accuracy of the model starts to decrease and the generalisation performance deteriorates. When the Dropout value is too small, the number of randomly discarded neurons is relatively limited, and a large number of neurons are still involved in the computation and parameter updating, resulting in relatively poor overall performance. As the Dropout rate gradually rises and the number of discarded neurons increases, the performance of the model gradually reaches the optimum. However, when the Dropout continues to increase further, the number of discarded neurons is too high, resulting in the model's metrics starting to decline. Therefore, it was chosen to set the Dropout to 0.3. When the number of nodes in the hidden layer of BiGRU was increased from 32 to 128, the accuracy and F1 value of the LFBP model in the dataset gradually increased and reached the optimal level. However, when the number of nodes is further increased to 128, the accuracy rate and F1 value of the model begin to decrease, indicating that the overall performance of the model begins to decline. Therefore, the number of nodes in the hidden layer of the BIGRU module is set to 128.

The model has 12 transformer modules, 12 self-attentive heads, and 768 hidden units. Adam is a first-order adaptive learning rate optimisation algorithm based on stochastic gradient descent (SGD), which dynamically changes the learning rate by calculating the exponentially weighted moving average of the gradient and the exponentially weighted moving average of the square. Therefore, the Adam optimisation algorithm is used to train the model. The parameter Hidden denotes the number of neurons in the hidden layer. The parameter batch_size is the number of samples to be captured for each training, which can affect the extraction of the overall features and the direction of gradient descent of the text data. The parameter dropout is a regularisation technique used to prevent the neural network from overfitting, it is a floating point number between 0 and 1. The parameter epochs is the number of times the model traverses the entire training dataset during the training process, and the training task is stopped if the accuracy is not improved in three iterations. Combined with the above parameter experiments, the final parameter settings of the model are listed in the Table 6 below.

TABLE 5. The comparison of batch parameters, drop rate and hidden nodes.

Batch-size	Accuracy	F1	Dropout	Accuracy	F1	Number of nodes	Accuracy	F1
16	0.8856	0.8723	0.1	0.8941	0.8750	32	0.8719	0.8628
32	0.8946	0.8755	0.2	0.8974	0.8778	64	0.8935	0.8756
64	0.9123	0.8839	0.3	0.9134	0.8853	128	0.9176	0.8825
128	0.9035	0.8818	0.4	0.9023	0.8816	192	0.8962	0.8790
256	0.8952	0.8795	0.5	0.8786	0.8752	256	0.8855	0.8672

TABLE 6. Description of model parameters.

Parametric	Numerical value
Optimiser	Adam
Learning rate	1×10^{-5}
Hidden layer activation	Relu
Hidden	768
Epochs	10
Embedding	128
Batch-size	64
Dropout	0.3

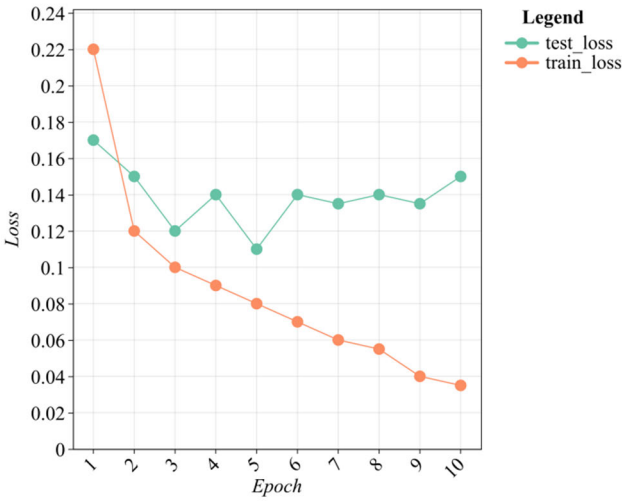

FIGURE 11. Loss curves for the training and test sets.

C. RESULTS AND DISCUSSION

In order to verify the effect of LFBP model, this paper iteratively trained 7 classical models 10 times on 3 datasets respectively, and recorded the test set accuracy, F1 value of each iteration, which visually reflected the advantages and disadvantages of the models. The selected comparison models are:

BiGRU: training word vectors using text-based word2vec, literature [12] A bidirectional variant of the gated recurrent unit proposed to splice the hidden layer outputs of the last moment in both positive and negative directions as sentence representations.

BiLSTM-Att: Literature [13] The proposed bi-directional long and short-term memory network based on Attention mechanism is used to weight and sum the hidden layer outputs of each moment of BLSTM to obtain the sentence representation.

BERTbase [23]: Based on Google open source bert-base fine-tuned to do classification with [CLS] text aggregation sequences output from the coding layer.

ALBERTbase [40].: Based on the open source model fine-tuning, the sentiment classification is performed using [CLS] after the text feature information extracted from the embedding layer is fed into the fully connected layer.

RoBERTa [24]: Based on the open source model fine-tuning, sentiment classification is performed using [CLS] after the text feature information extracted from the embedding layer is fed into the fully connected layer.

ERNIE [41]: The [CLS] symbol is added at the beginning of each input sequence, which is then fed into the ERNIE 3.0Base pre-training model. The model will encode each symbol to get its hidden state vector. In this paper, we take the hidden state vector corresponding to the [CLS] symbol as the semantic representation of the whole text, and then obtain the probability distribution of emotion classification through a fully connected layer and a Softmax layer.

FinBERT: Literature [34] FinBERT (base) proposed.

LFBP: The sentiment analysis model proposed in this paper.

The experimental results show that the performance of the model using Adam is better. After using Adam's algorithm, the loss on both the training and test sets is reduced to a certain extent. In addition, the model accuracy curves show that the loss in the test set decreases in the first 5 epochs and then remains stable. For the training set, the loss decreases faster in the first 5 epochs and then starts to decrease slowly. This indicates that the model has learnt the training data in the first 5 epochs and is able to generalise well to new data. However, in the later epochs, the model started to overfit the training data, which led to an increase in the loss of the test set. Therefore, earlystopping was added to the algorithm to prevent overfitting caused by continuing training after the loss of the test set started to rise. The details are shown in the Figure 11 below.

TABLE 7. Experimental results.

Model	PhraseBank		SEntiFin		StockSen	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
BiGRU	0.7234	0.6745	0.7511	0.6789	0.7123	0.6904
BiLSTM-Att	0.7526	0.7002	0.7729	0.7017	0.7321	0.7189
BERT	0.8215	0.7503	0.8438	0.7489	0.8147	0.7599
ALBERT	0.8456	0.7618	0.8601	0.7620	0.8341	0.7701
ERNIE	0.8741	0.8021	0.8903	0.7939	0.8578	0.7893
RoBERTa	0.8934	0.8608	0.9113	0.8323	0.8879	0.8323
FinBERT	0.8956	0.8611	0.9245	0.8611	0.9032	0.8678
LFBP	0.9389	0.8876	0.9589	0.9338	0.9432	0.9098

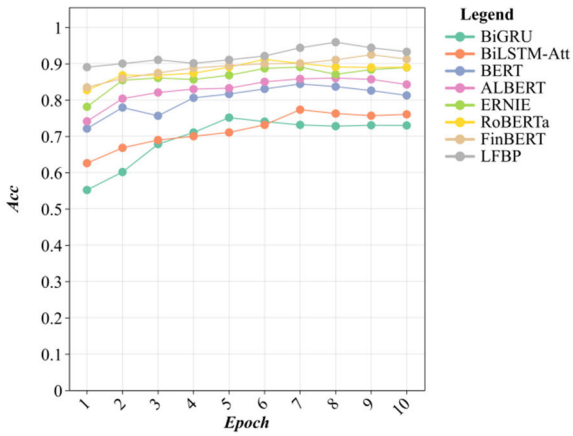

FIGURE 12. Accuracy change curve.

In this paper, we experimentally analyse the trend of the accuracy of this model and seven groups of comparison models under different iterations. From the trend graph, it can be seen that the method has a faster convergence of the loss function compared with the other models, and performs well on both the training set and the test set. It is worth noting that even when the number of iterations is small, the model can obtain a high classification accuracy, which is an advantage that traditional deep learning does not have. Compared with other models, this model has high accuracy, fast convergence and more stability as the number of iterations increases. After the 8th iteration, the model has reached the optimal parameters and the training time of each eopch of the model remains unchanged. This point further shows that the method proposed in this paper is not only effective, but also more beneficial for training. The relationship between the number of experimental iterations and the accuracy is shown in Figure 12.

Since BERT adopts the innovative bidirectional Transformer encoder structure and utilises large-scale Chinese and English datasets for pre-training, it effectively solves the problem of parameter forgetting in traditional deep learning algorithms, and at the same time learns rich information about the syntactic structure of text. The model proposed in this paper combines the text topic vectors extracted from the topic analysis layer on the basis of the fine-tuned Finbert model, which improves the ability of the model to acquire sentiment information. The model proposed in this paper is compared with the following model:BiGRU, BiLSTM-Att, BERTbase, ALBERTbase, RoBERTa, ERNIE, FinBERT are compared, and the results show that the model proposed in this paper outperforms the other models in two indicators. All the comparison experiments are carried out in the same experimental environment, and the validation set effect is evaluated every 10steps of the classification task training, and the optimal effect of the validation set is taken as the reporting index, and the specific results are listed in the following Table 7.

In order to validate the effectiveness of the proposed method of topic feature vectors with improved attention mechanism for the task of text sentiment analysis, this paper conducts experiments on datasets of several segmented domains and compares them with multiple methods. Details are as follows.

LT-BERT: The topic vectors output from the topic analysis layer are spliced with the output vectors from the BERT embedding layer before inputting into the coding layer, and after iterative training, sentiment classification is performed using [CLS].

LT-ALBERT: Based on the ALBERT model, the topic vectors extracted by the fusion LT algorithm are used for sentiment classification using [CLS].

RoBERTa-A: Based on the RoBERTa model and the standard attention mechanism, the input is fully connected layer, and the classification is realized by the Softmax function.

RoBERTa-PA: Based on the RoBERTa model, and using the improved attention mechanism method extracted from this paper into the training, the input fully connected layer, after the Softmax function to achieve the classification.

In order to verify whether the fusion of topic features and the improved attention mechanism approach proposed in this paper help to enhance the model sentiment analysis, experiments were conducted on both SEntiFin and

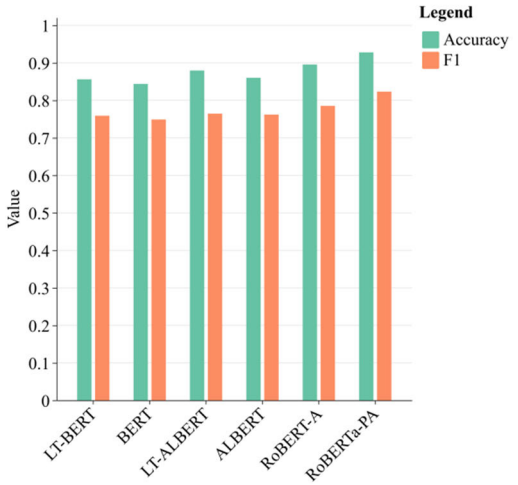

FIGURE 13. Validity verification.

FinacialPhasebank datasets, as shown in Figure 13, where the fusion of topic features as well as the improved attention mechanism approach on both datasets have improved the performance of the models. Taking the results shown on the SEntiFin dataset as an example, the BERT model embeds the preprocessed financial text data, uses the BERT language model to process the semantic information and combines it with the topic vectors extracted from the LT layer, which are jointly involved in the final sentiment categorisation, and compared to the unprocessed BERT model, it improves the accuracy by 1.23 percentage points and the F1 results by 1.01 percentage points. The ALBERT model embeds the preprocessed financial text data, uses the ALBERT language model to process the semantic information and combines with the topic vectors extracted from the LT layer to jointly participate in the final sentiment classification, which improves 1.95 percentage points in terms of accuracy compared with the unprocessed ALBERT model. Afterwards, the addition of the standard attention mechanism to the RoBERTa model resulted in a 1.78 percentage point improvement in F1 results compared to training iterations based on the RoBERTa model and using the improved attention mechanism approach. By comparing with BERT,ALBERT,RoBERTa,LT-BERT,LT-ALBERT, RoBERT-A, RoBERTa-PA models, it proves that the fusion of topic features and the improved attention mechanism method proposed in this paper can help to enhance the effectiveness of the model sentiment analysis.

But why do topic features lead to significant improvements in model performance? We believe that topic features enable the model to extract rich semantic information enabling it to capture the underlying semantic structure in the text. Financial texts often involve complex concepts and themes, and by introducing topic features, the semantics of the text can be understood more comprehensively, thus improving the accuracy of sentiment analysis. In addition, topic features are able to capture long-distance word associations and enhance the understanding of text context, which is particularly important in financial texts where the expression of sentiment may span multiple sentences or even the entire document. In addition, by splicing topic features with word vector features to form a more representative feature matrix, the model is enabled to learn and classify in a richer feature space. Enhancing the contextual dependency of sentiment recognition, topic features can provide contextual information about the text, enabling the sentiment analysis model to determine sentiment polarity based on the topic context. For example, certain themes (e.g., market crashes, financial scandals) may favour negative sentiment, while others (e.g., earnings growth, innovative technologies) may favour positive sentiment. By combining theme features, the model can better adapt to different types of financial texts, improve generalisation to new data, reduce overfitting, and improve the generalisation ability of the model.

Why does the use of improved attention mechanisms lead to significant improvements in model performance? We believe that it is the adaptive attention thresholding and attention weight masking mechanisms that play a role. The adaptive attention threshold dynamically adjusts the attention weight expectation based on the input text features, which improves the flexibility and accuracy of the model and makes it more adaptable to various input texts. In addition, the multi-word masking mechanism selects the word with the highest attentional weight rather than only one word for masking, which can capture multiple key information points more comprehensively, improve model performance, and reduce the noise caused by improperly allocating attention to a single word, so as to capture more contextual information and avoid information loss, which improves the stability of the model.

It can be seen that the FinBERT pre-training model based on knowledge enhancement enhances the semantic representation of words due to the adoption of the bi-directional Transformer encoder with powerful feature extraction capability and by modelling the a priori semantic knowledge of entity concepts and other semantic knowledge in the huge amount of data in the financial domain, and it also fully demonstrates that the pre-training model is able to distinguish different meanings of the same word based on the contextual background of the text sequence. Different meanings of the same word are distinguished, the addition of thematic features makes the model extract rich semantic information enabling it to capture the potential semantic structure in the text, and the adaptive attention threshold of the improved attention mechanism dynamically adjusts the expectation of the attention weights according to the input text features, which improves the model's flexibility and accuracy and thus improves the performance of sentiment analysis.

D. ABLATION EXPERIMENTS

In order to verify the impact of each module of the model on the performance of the whole sentiment classification model and the effectiveness of each module, ablation experiments of different structures in the model are set up,

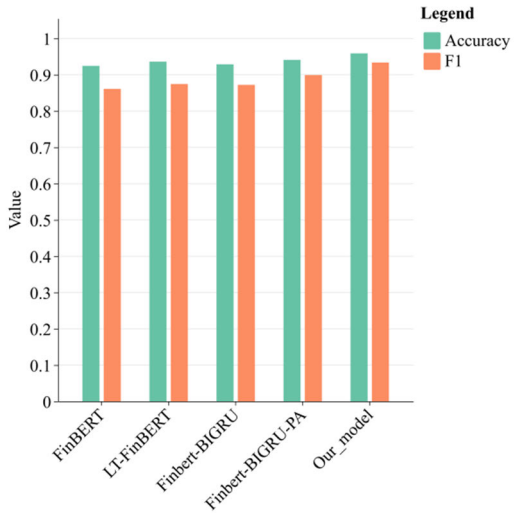

FIGURE 14. Validity verification.

and the specific details of the model are as follows: FinBERT: the pre-trained model in the financial domain; LT-FinBERT: the pre-trained model increases the topic vectors; Finbert-BIGRU: the pre-trained model increases the BiGRU module; Finbert-BIGRU-PA: pre-training model adding BiGRU module and using an improved attention mechanism method for training iterations; Our-model: the model proposed in this paper. In this section, ablation experiments are conducted which are used to analyse the performance of each module and the results are recorded in Table 8 and Figure 14. It is found through ablation experiments that each module of the model plays a key role in improving the performance.

Firstly, the LT-FinBERT model embeds the preprocessed financial text data, uses the FinBERT language model to process the semantic information and combines with the topic vectors extracted from the LT layer, which together participate in the final sentiment classification. Compared with the unprocessed FinBERT model, there is an improvement of 1.34 percentage points in the F1 results. In the following, based on the FinBERT model, the BiGRU module is added to enhance the model's temporal features, ability of the text. Compared with the unprocessed FinBERT model, there is an improvement of 1.1 percentage points on the F1 results. Afterwards, based on the FinBERT-BiGRU model and using the improved attention mechanism method for training iterations, the semantic feature output of the FinBERT model is extracted and combined with BIGRU to extract the key information contained in the temporal features, and the improved attention mechanism method is used for training iterations, which results in an improvement of 3.8 percentage points in the F1 results compared to the unprocessed FinBERT model. Finally, based on the FinBERT-BiGRU model, the LT topic extraction module is added to obtain the final model. By comparison, it can be seen that the model proposed in this paper improves about 7.27 percentage points on the F1 results compared to the unprocessed FinBERT model, which confirms the effectiveness of the model.

TABLE 8. Results of ablation experiments.

Model	Accuracy	F1
FinBERT	0.9245	0.8611
LT-FinBERT	0.9361	0.8745
Finbert-BIGRU	0.9287	0.8721
Finbert-BIGRU-PA	0.9409	0.8991
Our_model	0.9589	0.9338

V. CONCLUSION

In this paper, we explore the possibility of applying topic linguistic features and improved attention mechanism methods in the field of financial text analysis to ameliorate the problems encountered in sentiment analysis in challenging contexts such as the financial sector, such as unclear sentiment polarity of fused texts, high context dependency, and highly specialised and also expression-specific linguistic features of financial texts. Specifically, theme extraction is performed using the theme analysis layer, and then the extracted themes are spliced with the original data and fed into the FinBERT pre-training model to learn the semantic features of the text. Next, the obtained word vectors are transferred to the BiGRU model to enhance the text sequence information so as to improve the ability to learn long sequences of semantic sequences, and to enhance the weights of the feature words and integrate the text semantics by applying an improved attention mechanism. The results show that the LFBP-based sentiment analysis model exhibits better performance than other models in all indicators.

In addition, the optimisation of the model over several iterations by means of an improved approach to the attention mechanism leads to better results in terms of performance. The improved attention mechanism enhances the performance of the model through several aspects. Adaptive Attention Threshold dynamically adjusts the attention weight expectation based on input text features, which improves the flexibility and accuracy of the model and makes it more adaptable to various input texts. For complex financial reports, the adaptive threshold can be dynamically adjusted, so that the model pays more attention to key indicators such as ''growth rate'' and ''net profit''. In addition, the multi-word masking mechanism selects the word with the highest attention weight, instead of only selecting one word for masking, which can capture multiple key information points more comprehensively, improve the model performance, and reduce the noise caused by improper allocation of attention to a single word, so as to capture more contextual information, avoid information loss, and thus improve the stability of the model. For example, when dealing with the sentence ''The company's quarterly earnings exceeded expectations, leading to a rise in stock prices'', the improved attention mechanism makes the model pay more attention to ''quarterly earnings'' and ''rise in stock prices'', which enables the model to understand the sentiment tendency of the text more

comprehensively. With these improvements, the attention mechanism is able to capture the key information in the text more accurately and comprehensively, which significantly improves the performance of the financial text sentiment analysis model. It can also be noted from our experiments that the use of improved attention-based mechanisms represents a qualitative leap forward when it comes to increasing the accuracy of the model.

However, in sentiment analysis, the fact that fully extracting some features usually achieves better performance than others is ignored, except in some texts. For example, when some linguistic cues are hard to obtain with a transformer, they can be guessed. This is an example of expressive lengthening. Whereas, for rich semantic information, topic features can capture the underlying semantic structure in the text. Financial texts often involve complex concepts and themes, and by introducing thematic features, the semantics of the text can be understood more comprehensively, thus enhancing the accuracy of sentiment analysis. On the other hand, contextual associations are enhanced, and thematic features can capture long-distance word associations and enhance the understanding of text context. This is particularly important in financial texts, where the expression of sentiment may span multiple sentences or even the entire document. Therefore, the use of thematic features of text offers the possibility of performance enhancement in sentiment analysis, which improves the accuracy of combining with transformers.

As future work, we build even larger corpora, retrain the LFBP-based models, retrain the models in real time using incremental learning mechanisms, and perform error analysis to check if the accuracy of the system has improved. In another sense, we can explore how emotions are propagated through entities and their relationships and discover novel products that can be invested in the short or medium term, which in turn can be used to model different types of customers and their preferences by extracting demographic and psychographic information, and build recommendation systems to help companies and individuals choose investments based on their preferences.

REFERENCES

[1] S. Ghazi and M. Schneider, "An empirical study of the sentiment capital asset pricing model," *SSRN Electron. J.*, vol. 1, no. 1, pp. 1–20, 2020.

[2] C. M. Liapis, A. Karanikola, and S. Kotsiantis, "Investigating deep stock market forecasting with sentiment analysis," *Entropy*, vol. 25, no. 2, p. 219, Jan. 2023.

[3] S. García-Méndez, F. de Arriba-Pérez, A. Barros-Vila, and F. J. González-Castaño, "Targeted aspect-based emotion analysis to detect opportunities and precaution in financial Twitter messages," *Expert Syst. Appl.*, vol. 218, May 2023, Art. no. 119611.

[4] Q. Xiao and B. Ihnaini, "Stock trend prediction using sentiment analysis," *PeerJ Comput. Sci.*, vol. 9, Mar. 2023, Art. no. e1293.

[5] H. Yin and Q. Yang, "Investor sentiment mining based on bi-LSTM model and its impact on stock price bubbles," *Stud. Nonlinear Dyn. Econometrics*, 2023, doi: 10.1515/snde-2022-0028. [Online]. Available: https://www.degruyter.com/document/doi/10.1515/snde-2022-0028/html#MLA

[6] K. Du, F. Xing, R. Mao, and E. Cambria, "Financial sentiment analysis: Techniques and applications," *ACM Comput. Surv.*, vol. 56, no. 9, pp. 1–42, Oct. 2024.

[7] K. L. Tan, C. P. Lee, and K. M. Lim, "A survey of sentiment analysis: Approaches, datasets, and future research," *Appl. Sci.*, vol. 13, no. 7, p. 4550, Apr. 2023.

[8] T. Loughran and B. McDonald, "Textual analysis in finance," *SSRN Electron. J.*, vol. 12, pp. 357–375, Jul. 2020.

[9] A. M. Rojo López and M. Á. Orts Llopis, "Metaphorical pattern analysis in financial texts: Framing the crisis in positive or negative metaphorical terms," *J. Pragmatics*, vol. 42, no. 12, pp. 3300–3313, Dec. 2010.

[10] M. Rodríguez-Ibánez, A. Casánez-Ventura, F. Castejón-Mateos, and P.-M. Cuenca-Jiménez, "A review on sentiment analysis from social media platforms," *Expert Syst. Appl.*, vol. 223, Aug. 2023, Art. no. 119862.

[11] N. K. Singh, D. S. Tomar, and A. K. Sangaiah, "Sentiment analysis: A review and comparative analysis over social media," *J. Ambient Intell. Humanized Comput.*, vol. 11, no. 1, pp. 97–117, Jan. 2020.

[12] X. Zhu, P. Sobihani, and H. Guo, "Long short-term memory over recursive structures," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2015, pp. 1604–1612.

[13] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, and B. Xu, "Attention-based bidirectional long short-term memory networks for relation classification," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 207–212.

[14] X. Li, H. Xie, L. Chen, J. Wang, and X. Deng, "News impact on stock price return via sentiment analysis," *Knowl.-Based Syst.*, vol. 69, pp. 14–23, Oct. 2014.

[15] M. Wankhade, A. C. S. Rao, and C. Kulkarni, "A survey on sentiment analysis methods, applications, and challenges," *Artif. Intell. Rev.*, vol. 55, no. 7, pp. 5731–5780, Oct. 2022.

[16] S. Fraiberger, D. Lee, D. Puy, and R. Ranciere, "Media sentiment and international asset prices," *IMF Work. Papers*, vol. 18, no. 274, 2018, Art. no. 103526.

[17] F. H. Khan, U. Qamar, and S. Bashir, "A semi-supervised approach to sentiment analysis using revised sentiment strength based on SentiWordNet," *Knowl. Inf. Syst.*, vol. 51, no. 3, pp. 851–872, Jun. 2017.

[18] R. Pandarachalil, S. Sendhilkumar, and G. S. Mahalakshmi, "Twitter sentiment analysis for large-scale data: An unsupervised approach," *Cognit. Comput.*, vol. 7, no. 2, pp. 254–262, Apr. 2015.

[19] N. Schoon, "Earnings conference calls and stock returns: The incremental informativeness of textual tone," *CFA Dig.*, vol. 42, no. 2, pp. 14–16, May 2012.

[20] M. Caporin and F. Poli, "Building news measures from textual data and an application to volatility forecasting," *Econometrics*, vol. 5, no. 3, p. 35, Aug. 2017.

[21] T. Renault, "Intraday online investor sentiment and return patterns in the U.S. stock market," *J. Banking Finance*, vol. 84, pp. 25–40, Nov. 2017.

[22] X. Li and H. Ming, "Stock market prediction using reinforcement learning with sentiment analysis," *Int. J. Cybern. Informat.*, vol. 12, no. 1, pp. 1–20, Jan. 2023.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[24] L. Zhao, L. Li, X. Zheng, and J. Zhang, "A BERT based sentiment analysis and key entity detection approach for online financial texts," in *Proc. IEEE 24th Int. Conf. Comput. Supported Cooperat. Work Design (CSCWD)*, May 2021, pp. 1233–1238.

[25] K. Mishev, A. Gjorgjevikj, I. Vodenska, L. T. Chitkushev, and D. Trajanov, "Evaluation of sentiment analysis in finance: From lexicons to transformers," *IEEE Access*, vol. 8, pp. 131662–131682, 2020.

[26] B. Zhang, H. Yang, T. Zhou, M. Ali Babar, and X.-Y. Liu, "Enhancing financial sentiment analysis via retrieval augmented large language models," in *Proc. 4th ACM Int. Conf. AI Finance*, Nov. 2023, pp. 349–356.

[27] B. Zhang, H. Yang, and X.-Y. Liu, "Instruct-FinGPT: Financial sentiment analysis by instruction tuning of general-purpose large language models," 2023, *arXiv:2306.12659*.

[28] G. Fatouros, J. Soldatos, K. Kouroumali, G. Makridis, and D. Kyriazis, "Transforming sentiment analysis in the financial domain with ChatGPT," *Mach. Learn. Appl.*, vol. 14, Dec. 2023, Art. no. 100508.

[29] P. Rodriguez Inserte, M. Nakhlé, R. Qader, G. Caillaut, and J. Liu, "Large language model adaptation for financial sentiment analysis," 2024, *arXiv:2401.14777*.

[30] H. O. Ahmad and S. U. Umar, "Sentiment analysis of financial textual data using machine learning and deep learning models," *Informatica*, vol. 47, no. 5, pp. 1–18, May 2023.

[31] R. Pan, J. A. García-Díaz, F. Garcia-Sanchez, and R. Valencia-García, "Evaluation of transformer models for financial targeted sentiment analysis in Spanish," *PeerJ Comput. Sci.*, vol. 9, May 2023, Art. no. e1377.
[32] T. Jiang and A. Zeng, "Financial sentiment analysis using FinBERT with application in predicting stock movement," 2023, *arXiv:2306.02136*.
[33] N. Hu, P. Liang, and X. Yang, "Whetting all your appetites for financial tasks with one meal from GPT? A comparison of GPT, FinBERT, and dictionaries in evaluating sentiment analysis," Jul. 2023. [Online]. Available: https://ssrn.com/abstract=4426455 and https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4426455#paper-references-widget
[34] Z. Liu, D. Huang, K. Huang, Z. Li, and J. Zhao, "FinBERT: A pre-trained financial language representation model for financial text mining," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 4513–4519.
[35] M. Maia, S. Handschuh, A. Freitas, B. Davis, R. McDermott, M. Zarrouk, and A. Balahur, "WWW'18 open challenge: Financial opinion mining and question answering," in *Proc. Companion Web Conf. Web Conf.*, 2018, pp. 1941–1942.
[36] K. Cortis, A. Freitas, T. Daudert, M. Huerlimann, M. Zarrouk, S. Handschuh, and B. Davis, "SemEval-2017 task 5: Fine-grained sentiment analysis on financial microblogs and news," in *Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 519–535.
[37] P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala, "Good debt or bad debt: Detecting semantic orientations in economic texts," *J. Assoc. Inf. Sci. Technol.*, vol. 65, no. 4, pp. 782–796, Apr. 2014.
[38] F. Xing, L. Malandri, Y. Zhang, and E. Cambria, "Financial sentiment analysis: An investigation into common mistakes and silver bullets," in *Proc. 28th Int. Conf. Comput. Linguistics*, 2020, pp. 978–987.
[39] A. Sinha, S. Kedas, R. Kumar, and P. Malo, "SEntFiN 1.0: Entity-aware sentiment analysis for financial news," *J. Assoc. Inf. Sci. Technol.*, vol. 73, no. 9, pp. 1314–1335, Sep. 2022.
[40] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," 2019, *arXiv:1909.11942*.
[41] Y. Sun et al., "ERNIE 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation," 2021, *arXiv:2107.02137*.
[42] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.
[43] Q. Mei, X. Shen, and C. Zhai, "Automatic labeling of multinomial topic models," in *Proc. 13th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2007, pp. 490–499.
[44] W. Kou, F. Li, and T. Baldwin, "Automatic labelling of topic models using word vectors and letter trigram vectors," in *Proc. 11th Asia Inf. Retr. Societies Conf.*, 2015, pp. 253–264.
[45] E. Zosa, L. Pivovarova, M. Boggia, and S. Ivanova, "Multilingual topic labelling of news topics using ontological mapping," in *Proc. Eur. Conf. Inf. Retr.*, 2022, pp. 248–256.

GANGLONG DUAN was born in January 1977. He is currently an Associate Professor and the Master's Supervisor of the Department of Management Science and Engineering, School of Economics and Management, Xi'an University of Technology. He has published more than 30 articles in important Chinese and English academic journals at home and abroad, including 24 SCI and EI indexes. He won four scientific and technological achievement awards, two patents, and seven software copyrights.

SHUNFEI YAN was born in Yancheng, China, in November 1999. He is currently pursuing the master's degree with the Department of Management Science and Engineering, School of Economics and Management, Xi'an University of Technology.

MENG ZHANG was born in Jinzhong, China, in December 1999. She is currently pursuing the master's degree in accounting with the School of Economics and Management, Xi'an University of Technology.