Application Analysis and Research of Text Model Based on Improved CNN-LSTM in the Financial Field

Jing Chen, Chensha Li*
Foreign Language School, Guangdong University of Science and Technology, Dongguan, 523083, China

***Abstract*—With the continuous development of information technology, public opinion analysis based on open-source texts and financial situation awareness has become a research hotspot. This study focuses on financial news and commentary information. First, a topic crawler classification model combining the advantages of CNN and LSTM is proposed to improve the topic recognition ability of financial news texts, and a CNN-LSTM-AM stock price fluctuation prediction model is proposed. This model performs sentiment analysis through BiLSTM, integrates multiple emotional factors and market historical data, and demonstrates superior predictive performance compared to traditional models in multiple experiments.**

***Keywords*—Financial information mining; CNN-LSTM model; stock price prediction; sentiment analysis; BiLSTM**

I. Introduction

Under the background of the vigorous development of big data and artificial intelligence technology, open source text technology has shown significant application value in many fields such as public opinion monitoring, social governance, public security and financial risk early warning. Especially in the financial field, the speed and scope of information dissemination have a profound impact on market sentiment fluctuations, stock price changes and even the stability of the entire financial ecosystem. Therefore, how to efficiently and accurately extract valuable financial intelligence from massive, heterogeneous and dynamically updated information sources, and intelligently model market trends and emotional tendencies have become research hotspots and technical problems that are widely concerned by academia and industry.

At present, unstructured data sources such as online news, social media platforms and investor reviews have become an important part of financial open source intelligence [1]. Although these textual data provide rich material for financial event recognition and market analysis, their large and unstructured nature also poses significant challenges: On the one hand, the information collection process is often faced with problems such as semantic ambiguity, unclear topic and serious noise interference, which makes the traditional crawler technology based on rules or keyword matching perform poorly in dealing with the precise extraction task of specific domain intelligence [2]. On the other hand, the sensitivity of financial markets to sudden events requires intelligence systems to have real-time and efficient text perception and semantic understanding capabilities [3]. However, most existing methods still have significant deficiencies in dealing with the structure and semantic modeling of sudden text [4]. In addition, as an important factor driving the volatility of financial markets, the dynamic evolution of sentiment in multi-source data such as news reports and investor comments still lacks an effective fusion modeling mechanism [5].

Current research is confronted with multiple challenges and limitations in the field of financial text analysis. Firstly, with the explosive growth of information volume in the financial market, multi-source heterogeneous data such as news reports, social media and investor comments have become increasingly complex. Traditional text capture and analysis methods based on rules or keyword matching have become inadequate. This type of method has limited capabilities in semantic understanding and topic recognition, and is vulnerable to semantic ambiguity, information noise and interference from unexpected events, resulting in poor performance in highly dynamic and complex financial scenarios. Furthermore, many existing models lack effective integration mechanisms when dealing with the implicit sentiment changes and event-driven characteristics in financial language, making it difficult to comprehensively capture the dynamic evolution process of market sentiment. These deficiencies limit the application effect of the financial intelligence system in market trend prediction, sentiment perception and risk early warning, and also reduce the practicability and stability of the model.

The main objective of this study is to construct an intelligent model system for financial text analysis, enhance the semantic understanding, topic recognition and sentiment prediction capabilities of financial unstructured text data, and thereby achieve more accurate and efficient financial public opinion monitoring and market trend prediction. Specifically, this study aims to solve the problems such as semantic ambiguity, unclear theme, poor real-time performance and weak ability of emotion evolution modeling that traditional methods face when dealing with open texts in the financial field. To this end, the study proposes an improved CNN-LSTM joint model. By introducing the convolutional neural network (CNN) to efficiently extract the local features of the text and combining it with the Long Short-Term Memory network (LSTM) to process the time-dependent information in the text, the recognition ability of hidden topics in financial news and commentary texts is enhanced. Meanwhile, in terms of stock price fluctuation prediction, a CNN-LSTM-AM model based on BiLSTM and the Attention Mechanism (AM) was constructed, integrating emotional factors and historical market data to achieve the joint modeling of emotion-driven and data-driven so as to capture the

deep-seated factors influencing stock price changes more comprehensively. Through systematic model design and multi-dimensional data fusion, this study aims to achieve a complete process optimization from information collection, feature extraction, emotion recognition, to trend prediction, enhance the intelligent level of financial text analysis, and provide a technical basis and theoretical reference for financial decision support.

This study focuses on the application analysis and research of the text model based on the improved CNN-LSTM in the financial field. Section II sorts out the current research progress in the topic recognition and sentiment analysis of financial texts and clarifies the deficiencies of the existing methods. In Section III, this study proposes a focused crawler model combining the advantages of CNN and LSTM for financial text collection and classification, and designs a BiLSTM sentiment analysis model integrating the attention mechanism for stock fluctuation prediction. In Section IV, the performance of the model was verified through multiple sub-experiments, including text crawling, topic classification, comment sentiment analysis and stock price prediction. The experimental results show that the proposed model is superior to the traditional methods in terms of accuracy and practicality. The conclusion section in Section V summarizes the research results and practical significance, and looks forward to the future research directions, proposing that the integration of multi-source data and the improvement of the model's generalization ability can be further explored.

II. Related Work

In the context of information explosion and highly sensitive financial markets, financial intelligence mining and intelligent prediction based on text data is gradually becoming an important research direction in the intersection field of Natural Language Processing (NLP) and financial engineering. With the continuous expansion of open source channels such as social media, news portals and review platforms, a large amount of unstructured text data is emerging, which contains a large amount of potential information that can be used for market trend analysis, public opinion monitoring and investment decision-making [6]. How to effectively integrate these heterogeneous data resources and realize the accurate identification and modeling of financial sentiment, emergencies and market dynamics, it is urgent to rely on advanced deep learning technology to build an intelligent information processing framework [7].

In recent years, convolutional neural Network (CNN) and Long Short-Term Memory network (LSTM) have achieved remarkable results in text classification and sequence modeling tasks [8]. CNN is good at extracting semantic features from local context, while LSTM is able to capture long-distance dependencies, and the joint application of them provides new ideas for deep modeling of complex financial texts [9]. However, in the face of the characteristics of high semantic complexity, strong burstiness and obvious emotional fluctuations in financial corpus, the traditional CNN-LSTM model still has shortcomings in information extraction accuracy, timeliness of event recognition and emotion modeling dimension [10]. Next, starting from the research status of financial text information processing, this study summarizes the development of related model technologies, and analyzes the main challenges and development trends faced by current research.

A. Text Mining and Topic Identification of Financial Texts

With the rapid advancement of internet technology and the media industry, news has become one of the primary channels through which the public accesses social developments and financial information [11]. Traditional print media is swiftly transitioning toward digitalization and network-based formats, and news data is now characterized by large-scale, rapid updates, and diverse content [12]. Effectively classifying and intelligently mining these large-scale news text datasets has emerged as a key research direction in the field of natural language processing (NLP) [13]. In highly sensitive domains such as finance, the intelligent processing of news data holds significant importance for information retrieval, intelligence monitoring, and market behavior prediction.

Early research in text classification primarily relied on shallow learning algorithms such as Naive Bayes, Support Vector Machines (SVM), and K-Nearest Neighbors (KNN), which modeled news texts using the bag-of-words approach for classification [14]. While these methods demonstrated a certain level of classification capability, they often faced challenges such as feature sparsity and poor generalization when applied to large-scale, high-dimensional, and semantically complex financial texts [15]. With the rise of deep neural networks, convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have been increasingly adopted. CNNs are effective in capturing local key features within text and are well-suited for handling phrase-level semantic structures [16]. Meanwhile, Long Short-Term Memory networks (LSTMs), an improved variant of RNNs, perform well in processing long texts and capturing dependencies over time [17]. Researchers have attempted to combine the strengths of both architectures to overcome the limitations of single models in representing complex semantics. For example, Duan et al. conducted some studies using CNNs as feature extractors to generate embedded features, which are then fed into LSTM networks for semantic modeling, significantly improving classification accuracy in financial topic classification tasks [18].

Traditional text feature extraction methods fall into three main categories: rule-based and statistical methods, machine learning-based methods, and more recently, deep learning methods that have shown superior performance in text classification tasks [19]. Most CNN-based news data classification models are designed to learn local contextual information within fixed-size windows and extract features by sliding these windows across the text, which are then used for classification [20]. However, CNNs have limitations, particularly in capturing dependencies among local features. Addressing this issue, this study proposes a hybrid model that integrates a multi-scale dual-layer CNN with a BiLSTM and an attention mechanism, termed TCNNRes-BiLSTM-Attention (TCBA) [21]. This model simultaneously captures contextual features from both forward and backward directions of the text sequence while incorporating a dual-layer multi-scale convolutional structure [22].

B. Survey of Financial Condition Forecasting Research based on Financial Commentary Analysis

For investors, the stock market is a platform to identify and invest in high-potential companies with the aim of securing substantial returns. Over the years, both investors and researchers have been devoted to the task of stock price prediction. However, the stock market is inherently dynamic, nonlinear, non-stationary, non-parametric, noisy, and chaotic, making it extremely challenging to analyze market trends and price behaviors [23]. The stock market is influenced by a variety of highly interrelated factors, including political events, macroeconomic conditions, commodity price indices, investor expectations, performance of other stock markets, and investor psychology [23]. The correlation between public sentiment and financial markets has long attracted considerable academic attention. Studies have shown that the emotional tendencies reflected in news reports and investor commentary can, to some extent, serve as precursors to market trends—especially in times of information overload or sudden adverse events, where emotional reactions often precede actual market movements [24]. Consequently, sentiment-driven stock price prediction has gradually become an important direction in intelligent financial analysis.

Traditional stock prediction methods primarily rely on statistical techniques for time series analysis. For instance, Matta et al. employed econometric approaches to forecast stock prices, demonstrating that models using multiple predefined variables could yield statistically significant stock return predictions [25]. The Autoregressive Integrated Moving Average (ARIMA) model, introduced by Box and Jenkins, has been widely used for time series forecasting, including stock market returns [26]. However, studies have revealed that ARIMA models tend to perform poorly on financial time series data [27]. To better account for volatility, the GARCH model was proposed to predict stock prices by modeling the relationship between price volatility and returns. Nonetheless, traditional predictive methods often struggle to balance the inherent randomness and regularity of stock movements and are plagued by various limitations, making them less suited to the needs of investors.

With rapid advancements in hardware technology, the development of powerful new algorithms, and the rise of big data and distributed computing technologies, the efficiency of stock volatility prediction has greatly improved. Big data technologies have made it significantly easier to access and store large volumes of historical trading data. In recent years, machine learning algorithms have undergone explosive growth, and many researchers have applied techniques such as Support Vector Machines (SVM), decision trees, Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) networks to stock volatility prediction, achieving promising results. SVM, first introduced by Vapnik, has demonstrated its effectiveness in many pattern recognition problems and is especially powerful in identifying subtle patterns in complex datasets [28]. It has been used in applications such as handwriting recognition, fraud detection, speaker identification, and facial recognition. Sonkavde et al. have employed SVM to identify stocks that might outperform the market by analyzing accounting fundamentals and price data [29]. Experimental results showed that equally weighted portfolios constructed from stocks selected by SVM achieved significant returns over five years, outperforming benchmark indices. Other studies predicted stock price direction using SVM and then applied trend analysis to derive corresponding price predictions. Despite modest profitability, these models still performed better than bank interest rates, suggesting a degree of practical value [30]. Additional research proposed a two-stage feature selection and prediction model using SVM, demonstrating better generalization ability than traditional methods [30]. Furthermore, Principal Component Analysis (PCA) has been introduced into SVM-based models to extract low-dimensional, efficient features, improving training accuracy and preserving the original data's characteristics. Empirical studies showed that PCA-SVM stock selection models contributed more significantly to annual portfolio returns than benchmark indices [31].

C. The Comparison Between the Current Research and this Study

In the current research on text classification and sentiment analysis, the academic community has accumulated a large number of methods based on traditional machine learning and deep learning, forming a relatively rich research system. The existing research mainly focuses on text classification models based on traditional algorithms such as Bayesian and Support Vector Machine (SVM), as well as sentiment analysis methods based on deep neural networks that have emerged in recent years. However, by systematically comparing these existing works with the model proposed in this study, the gaps and deficiencies of the current research can be clearly identified, and the innovation and necessity of this study can be highlighted. Early studies mostly relied on the Naive Bayes model. Although this method laid the foundation for text classification, it has obvious limitations of the independence assumption and is difficult to fully model the complex correlations among text features. LMDCampos et al. attempted to introduce a synonym library modeling through Bayesian networks, improving the representation ability of text semantics, but still remained at the relatively shallow stage of semantic extraction. Luo Huiqin, Zeng Yu, Li Xiaodong and others have optimized Naive Bayes to varying degrees, such as introducing feature weights and improving the calculation method of TF-IDF weights, which has enhanced the classification accuracy. However, the expressive ability and generalization ability of these methods are still restricted by the structure of the model itself. With the introduction of Support Vector Machine, the performance of text classification has been significantly improved. Xiao Zheng, Ding Shengchun, Shi Qiangqiang and others applied SVM to tasks such as sentiment tendency analysis and microblog opinion recognition, and achieved good results. However, the computational efficiency of the SVM algorithm is still relatively low when dealing with large-scale corpora and multi-dimensional emotional features, and the model has poor sensitivity to the word order of the input data, making it difficult to capture the context semantics.

In terms of deep learning, KIM Y improved the performance of sentence-level classification by introducing the CNN structure with parameter tuning, pioneering a new direction for the application of deep learning in text analysis. Zhou Jinfeng

further expanded the structure of CNN, utilized multi-layer convolution to extract local semantic features, and enhanced the expressive ability of the model. However, CNN relies on a fixed window during modeling, lacks sensitivity to time series features, and is unable to effectively retain the context dependency of the text. To solve this problem, subsequent research gradually shifted to models based on RNNS. Huang M introduced tag-specific synthetic functions and structural control strategies in RNN, which improved the fine-grained control ability of sentiment classification. FeiH enhances the semantic capture ability of LSTM through the keyword sensitivity mechanism, achieving a slight performance improvement. RaoG proposed the SR-LSTM model, which models the semantic relationships at the sentence and document levels respectively, through a double-layer structure, and captures the emotional information in long texts more effectively. Although these improved schemes based on LSTM have made certain progress, most studies are still limited to the single task of sentiment classification and lack the ability of multimodal data fusion and interaction modeling between sentiment and market behavior. Therefore, compared with the traditional static modeling methods based on Bayesian or SVM, this study breaks through the limitations of shallow feature engineering. Compared with the existing deep learning models, it has also achieved substantial breakthroughs in structural design and multi-source information integration. Especially in the application of the financial field, it has filled the gap in the modeling of the linkage between sentiment analysis and market behavior, providing a more practical and scalable technical path for subsequent research on financial text analysis and prediction.

III. Model Establishment and Solution

A. Research on Focused Crawler Technology Based on CNN-LSTM

In today's highly dynamic and information-intensive financial market environment, the ability of corporate decision-makers, financial analysts, and policymakers to perceive industry trends and breaking events has become a critical factor influencing the efficiency and quality of decision-making. Faced with rapidly evolving market conditions, financial professionals urgently need to rely on accurate and timely information to assess trends, evaluate risks, and adjust strategies. However, with the explosive growth of internet-based information, financial data is now characterized by large volume, rapid updates, high noise levels, and strong heterogeneity [32]. Traditional methods that depend on manual information retrieval and classification are increasingly unable to meet real-world demands. These manual approaches are not only inefficient but also susceptible to subjective judgment and information delays, which compromise the timeliness and scientific rigor of financial activities [33].

Compared with general-purpose texts, financial news exhibits strong topic cohesion and a high density of domain-specific terminology. Such news content typically centers on core themes including stock market trends, industry mergers and acquisitions, macroeconomic policies, and corporate earnings reports. The text often includes frequently used professional terms such as "price-to-earnings ratio", "return on equity", "policy dividend", and "acquisition agreement". This semantic concentration provides a favorable basis for building automated topic recognition models, as it allows the models to more easily capture and differentiate the semantic distinctions between various news topics.

- Overview of Focused Crawler Technology

This topic classifier functions as a binary text classification model, distinguishing between topic-relevant and topic-irrelevant content. The topic-focused crawling framework proposed in this study is illustrated in Fig. 1.

The core part of the focused crawler workflow is topic relevance calculation. In the content of news text, the sentences containing the topic content are often short and concise, so before the topic discrimination. Sentences with sentence length between 20 and 50 are selected for combination, and then the topic relevance is judged by the topic discriminator.

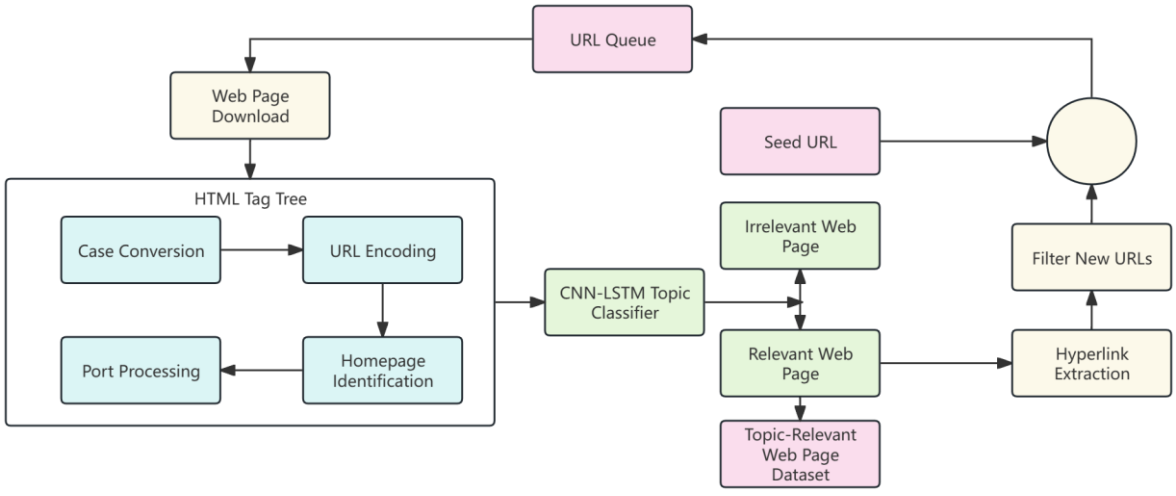

Fig. 1. Framework of focused crawler.

- The Focused crawler classifier

The text classification model based on CNN-LSTM can better extract features in documents. The training flow of the text classification model is shown in Fig. 2. After multiple experimental comparisons, the experimental parameters are set as follows: the learning rate of Adam optimizer is 0.001, the training set and the test set are divided according to the ratio of 8:2, dropout is set to 0.5, and epoch is set to 100.

First, the algorithm takes a preprocessed text dataset as input, where each sample is a document associated with a specific category label. The preprocessing typically includes steps such as tokenization, stop-word removal, and word normalization. Each document is represented as a sequence of words, $W_1, W_2, \ldots, W_n$, which are then fed into an embedding layer. The role of the embedding layer is to map each word to a vector in a continuous space, capturing semantic relationships between words, and thereby producing a sequence of word vectors. These word vectors are subsequently passed through a convolutional layer. The convolutional layer uses a sliding window mechanism to extract local features, effectively capturing word-level associations. After the convolution operation, a pooling layer is typically applied. The pooling layer downsamples the convolution output to reduce feature dimensionality and retain key information, which improves computational efficiency and generalization. The output from the pooling layer is then passed into an LSTM layer. LSTM, a specialized form of recurrent neural network (RNN), is capable of handling long-term dependencies and is well-suited for sequential modeling tasks. In text classification, the LSTM layer learns the sequential and contextual semantics in the document, capturing dependencies across different parts of the text.

The output of the LSTM is passed to a fully connected layer, which integrates the high-level features and produces a fixed-size output vector. This vector is then processed by a Softmax layer, which converts the vector into a probability distribution over all possible classes, thus determining the predicted category of the input document. The entire model is trained using backpropagation. In each training epoch, the model iterates over all training samples, performs predictions, computes the loss (typically using cross-entropy), and then calculates gradients based on the loss. Using these gradients, an optimizer (such as Adam, with a learning rate of 0.001) updates the model parameters to minimize the loss function. This process continues for a predefined number of epochs (set to 100 in the experiment) or until the model converges. To prevent overfitting, a Dropout mechanism is introduced during training, with a dropout rate set to 0.5. Before training begins, the dataset is split into a training set and a test set in an 80:20 ratio. After training, the model is evaluated on the test set to assess its generalization performance.

This pseudocode provides a clear step-by-step view of how a CNN-LSTM model is trained for text classification, covering data preprocessing, embedding, forward pass, loss computation, backpropagation, and model update.

- CNN-LSTM hybrid model

The CNN-LSTM hybrid model is a deep learning model that combines the advantages of Convolutional Neural Network (CNN) and Long Short-Term Memory network (LSTM), which can simultaneously capture the local features and temporal relationships of the input data [4]. The overall structure of the model is shown in Fig. 2.

The computation in the convolutional layer is defined by the following formula:

$$O = f(WZ + b) \tag{1}$$

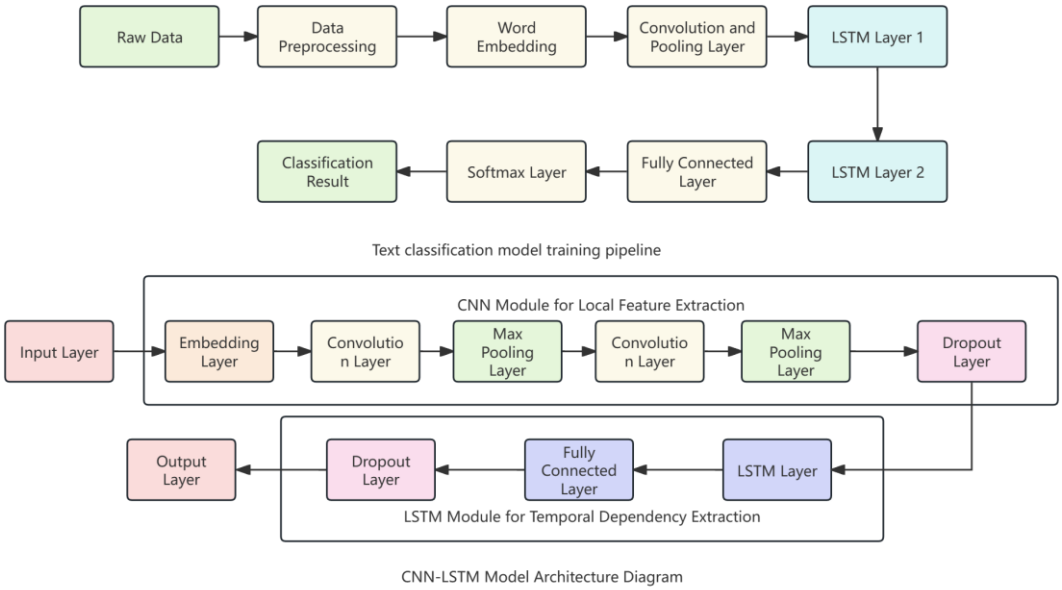

Fig. 2. Text classification training process and classification algorithm framework.

where, O is the output feature matrix after convolution, W is the weight matrix (convolutional kernel), Z is the input word embedding matrix, b is the bias term, f is the ReLU (Rectified Linear Unit) activation function, defined as $\mathrm{ReLU}(x) = \max(0, x)$, where x is the output vector from the previous neural layer.

The Long Short-Term Memory (LSTM) network is a special type of Recurrent Neural Network (RNN) designed to overcome the problems of vanishing or exploding gradients that occur when handling long sequences. LSTM introduces a gating mechanism—comprising the input gate, forget gate, and output gate—that allows the network to selectively remember or forget information. This mechanism enables LSTM to effectively capture long-range dependencies in sequential data. The fundamental architecture of an LSTM cell is illustrated in Fig. 3.

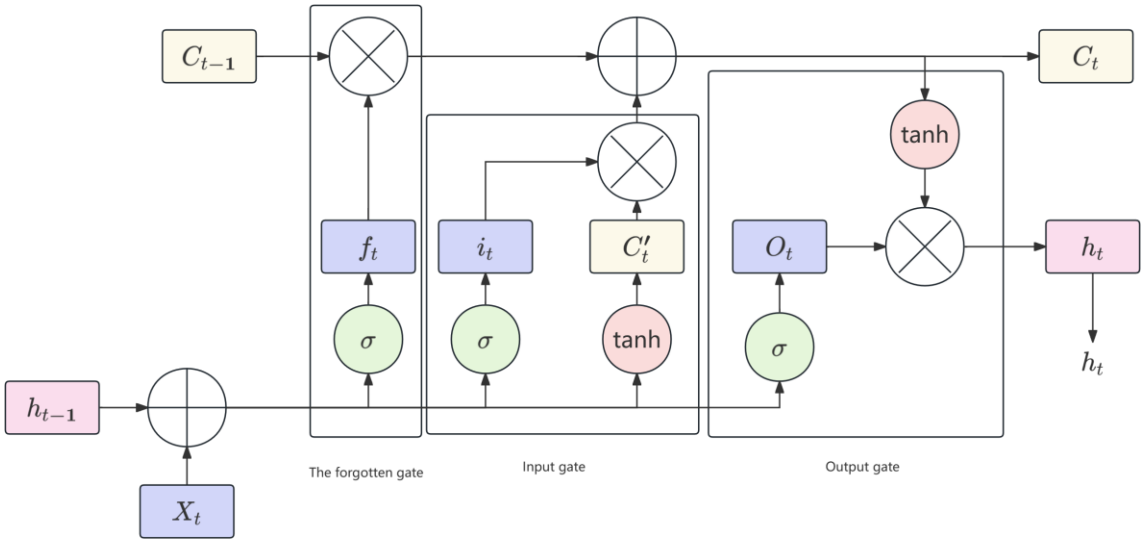

Fig. 3. Diagram of the basic unit structure of an LSTM model.

Lastly, the hidden state output h_t is computed by applying the output gate to the tanh-transformed current cell state c_t. The variables t and $t-1$ refer to the current and previous time steps, respectively, highlighting the sequential nature of the data being processed. Together, these components enable the LSTM to maintain and update information over time, effectively capturing long-term dependencies in sequential text data.

B. Sentiment Analysis of News and Investor Reviews

Investor comments have emotional tendencies, and the subjective views and emotional tendencies of comment authors can be mined through sentiment analysis. This plays an important role in the research field of the relationship between reviews and stocks, and the addition of review features can make the stock market prediction results more comprehensive, objective and explanatory.

The CNN-LSTM model first extracts the features of stock day-level funds, market prices, technical factors, news and comments through CNN, and then inputs the extracted features into the LSTM network for training to extract relevant time series information. At the same time, the Attention Mechanism is introduced after the LSTM network. This allows the network to focus on important features during training. The overall network structure of the proposed CNN-LSTM-AM model is shown in Fig. 4.

The basic unit of the Long Short-Term Memory (LSTM) model consists of four main components: the forget gate, input gate, memory cell, and output gate. These components work together to manage the balance between long-term memory retention and short-term information update, enabling the network to capture complex dependencies in sequential data.

The forget gate f_t decides which information from the previous memory cell should be discarded. It uses a sigmoid activation to output a value between 0 and 1 for each element in the memory cell:

$$f_t = \sigma\left(W_f[h_{t-1}, x_t] + b_f\right) \tag{2}$$

The input gate i_t determines which new information is to be added to the memory cell:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{3}$$

At the same time, a candidate memory content $\tilde{c}_t$ is created via a tanh layer:

$$\tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{4}$$

The new memory cell state c_t is computed by combining the previous memory cell c_{t-1}, the forget gate f_t, the input gate i_t, and the candidate content $\tilde{c}_t$:

$$c_t = f_t \times c_{t-1} + i_t \times \tilde{c}_t \tag{5}$$

The output gate o_t decides which part of the memory cell should be output.
The final hidden state h_t is obtained by applying the output gate to the updated cell state after passing through a tanh function:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (6)$$

$$h_t = o_t \times \tanh(c_t) \quad (7)$$

In the LSTM formulas, f_t, i_t, and o_t represent the states of the forget gate, input gate, and output gate, respectively. These gates control how information flows through the LSTM cell at each time step. The symbols W_f, W_i, and W_o denote the weight matrices associated with each of these gates, while b_f, b_i, and b_o are their corresponding bias terms. The notation $[h_{t-1}, x_t]$ indicates the concatenation of the hidden state from the previous time step h_{t-1} and the current input vector x_t. This concatenated vector serves as the input to the various gate functions.

The sigmoid function, represented by σ, is used to squash the outputs of the gates to values between 0 and 1, effectively determining the degree to which information should be remembered or forgotten. The candidate memory content $\tilde{c}_t$ is a proposed update to the cell state, calculated using the tanh function to ensure its values fall within the range of -1 to 1. The current memory cell state, denoted as c_t, is updated by combining the retained portion of the previous cell state c_{t-1} with the new candidate content $\tilde{c}_t$, scaled by the input gate. This combination is carried out through element-wise multiplication, indicated by the symbol $\times$, also known as the Hadamard product.

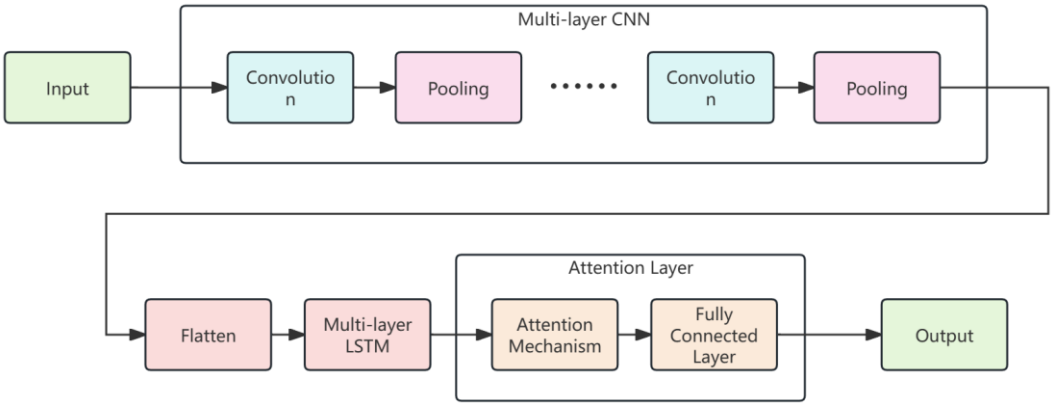

Fig. 4. CNN-LSTM-AM network structure diagram.

IV. Experimental Results and Analysis

A. Text Crawler Experiments

- Experimental parameter setting

In this section, the experiments were conducted using PyCharm as the development environment, with deep learning tasks implemented using PyTorch, a scientific computing library based on Python. The hardware setup includes an Intel i9-14900K CPU and an NVIDIA GeForce RTX 2080Ti 22GB GPU. All Python third-party libraries required in the experiments were managed using Anaconda.

In the control experiment, the maximum collection number of crawlers is set to 1000, that is, when T = 1000, the harvest rate of the theme crawler is compared as shown in Fig. 5. With the increase of the number of crawled pages, the harvest rate of each control experiment decreases, but the model proposed in this section is better than other groups, and the harvest rate is 0.72 at the end of the crawling task. The harvest rate is 0.10 higher than that of the focused crawler based on feature enhanced RNN, 0.26 higher than that of the focused crawler based on HMM, and 0.05 higher than that of the breadth-first strategy crawler.

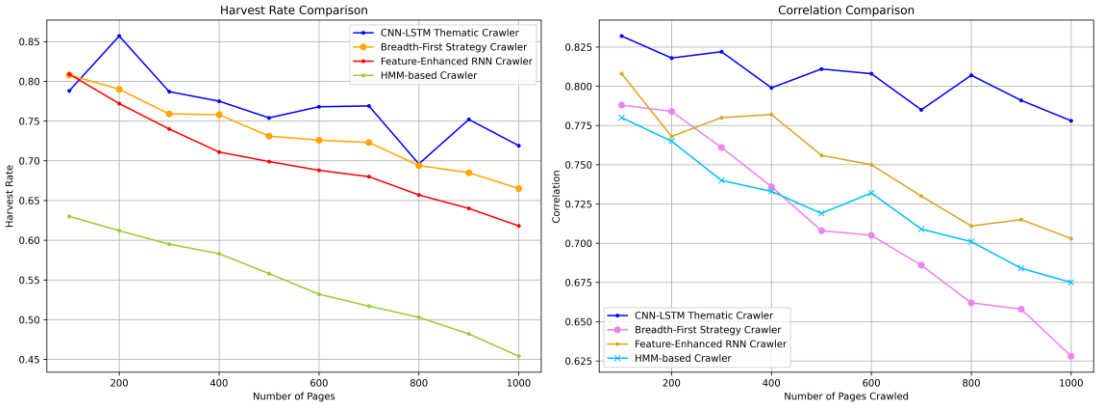

Fig. 5. Comparison of harvest rate and relevance of focused crawler

The correlation comparison of focused crawlers as the number of crawled pages increases, the overall correlation of each model decreases. At the end of the crawling task, the correlation of the model proposed in this section is 0.78, which is 0.08 higher than that of the focused crawler based on feature-enhanced RNN. Compared with the focused crawler based on HMM, the correlation is 0.11 and 0.15 higher than the breadth-first strategy crawler.

From the comparison of the above experimental results, it can be seen that the crawler based on HMM has the worst performance in the harvest rate and relevance index of the topic crawler because it does not discriminate the topic relevance. The focused crawler based on CNN-LSTM proposed in this study is better than other models based on content evaluation or link structure recognition.

B. *Experimental Design and Evaluation of Topic Classification Models*

- Dataset and metrics

In recent years, the text content of the news has a high time efficiency, and the use of the earlier data set may lead to the model's problem of the new term, so in the subject classification model experiment, on the one hand, according to the influence of the news reading quantity and the number of critics, the financial news text is used as the self-established language material library. On the other hand, the thcnews library, which is released by Tsinghua University, is based on the historical data of sina news between 2005 and 2011, which is divided into 14 categories, which randomly select 4,000 financial related news texts and 6,000 other types of news text [34]. The url is http://thuctc.thunlp.org/. Then, the material library is divided into financial topics, using 1 tag to mark the current data, which is related to the financial theme, which indicates that the current data is irrelevant to the financial theme. The distribution of each dataset is shown in Fig. 6, which consists of a random selection document in a mixed news cube, a training set, a validation set, and a test set.

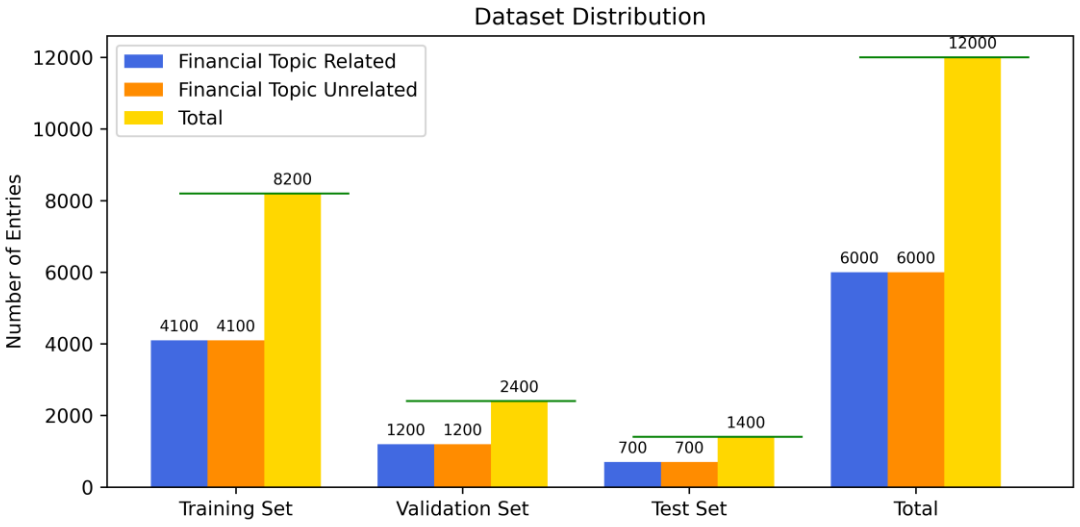

Fig. 6. Dataset distribution diagram.

- Experimental design and result analysis

By designing comparative experiments with machine learning and deep learning models, experiments are carried out on a mixed news corpus. The results of model comparison experiments are shown in Table I.

TABLE I MODEL COMPARISON OF EXPERIMENTAL RESULTS

Algorithm	Accuracy	F－Measure
SVM	83.6	80.2
RNN	86.3	83.5
CNN	91.1	87.2
RCNN	92.2	89.6
CNN－LSTM	93.4	90.8

By comparing the model proposed in this section with four models including Support Vector Machine (SVM), Recurrent Neural Network (RNN), Convolutional Neural Network (CNN) and Recurrent Convolutional Neural network (RCNN), the accuracy and F-Measure of the CNN-LSTM model can be concluded. The accuracy of the CNN-LSTM model reaches 93.4%. It indicates that the ability to correctly classify text is the strongest among all the compared models. Similarly, it also performs well on F-Measure, reaching 90.8%, indicating a good balance between precision and recall. Through longitudinal comparison, it is found that CNN-LSTM and RCNN models are superior to the traditional SVM model and the basic RNN model in accuracy and F-Measure. This is because for RCNN model, the advantages of RNN and CNN are combined, and for CNN-LSTM model, CNN can effectively capture local features. LSTM can capture long-term dependencies and process sequence data better through the characteristics of the two models to achieve good classification results. Therefore, the CNN-LSTM model proposed in this section is able to have better results in the topic classification task.

C. *Experiments on Sentiment Analysis of Reviews*

- Optimization parameter experiment

In this experiment, several different batch sizes of 8, 16, 32, 64, and 128 were selected for exploration. As shown in Table II, the experimental results show that when the batch size is chosen as 64, the model has the best effect, and the accuracy reaches 83.06%.

TABLE II ACCURACY STATISTICS FOR DIFFERENT BATCHSIZES

Batchsize	Accuracy
8	0.8127
16	0.8296
32	0.8257
64	0.8306
128	0.8263

In machine learning, the learning rate has a great impact on the performance of the model. The learning rate determines how much the parameters of the neural network are updated in each iteration. If the learning rate is too large, the parameter update step size will be too large, and the global optimal solution may be skipped, resulting in unstable training or failure to converge. If the learning rate is too small, the parameter update step size will be too small, which will lead to too slow training speed and need more iterations to converge. In this experiment, 0.5, 0.2, 0.1, 0.05, 0.02 and 0.01 were selected for exploration, and the experimental results are shown in Table III. As can be seen from the table, as the learning rate decreases from 0.5 to 0.05, the Accuracy of the model increases continuously. When the learning rate is 0.05, the model has the best effect, and the accuracy reaches 84.13%. When the learning rate is slowly decreased from 0.05 to 0.01, the model Accuracy decreases. Therefore, the learning rate of 0.05 is appropriate.

TABLE III ACCURACY VALUES FOR DIFFERENT LEARNING RATES

Learning rates	Accuracy
0.5	0.7526
0.2	0.7909
0.1	0.8306
0.05	0.8413
0.02	0.8335
0.01	0.8316

By adjusting the parameters above, the accuracy of the final model reaches a relatively excellent level. The stock review text of the test set is input into the model to predict the sentiment tendency, and the final evaluation index of the model is shown in Table IV below. It can be found that the model realizes a more accurate classification of investor comments, which lays a foundation for subsequent stock prediction.

TABLE IV INVESTOR COMMENTS CLASSIFICATION PREDICTION EVALUATION

	F1 score	Recall	Accuracy
Positive class	0.76	0.81	84.13%
Negative class	0.77	0.79	
Neutral class	0.79	0.83	

- Comparison model analysis

In this section, as a comparison, the investor comment vectors processed by word2vec are input into BP neural network, LSTM neural network, BiLSTM neural network and other machine learning models to compare their classification effects. After parameter tuning, the classification effect of each model is shown in Table V. It can be seen that the BERT-bilSTM model based on Bert and Bi-LSTM model has the best classification effect on investor comments, and its accuracy on the test set reaches 84.13. Therefore, in this study, Bert-BiLSTM is used to classify the sentiment of investor comments.

TABLE V EFFECT OF EACH MODEL INVESTOR COMMENT SENTIMENT CLASSIFICATION TASK

Model	Macro__F1__score	Macro__Recall	Accuracy
BP	0.74	0.79	79.38%
LSTM	0.76	0.79	82.74%
BiLSTM	0.76	0.80	83.24%
Bert—BiLSTM	0.77	0.81	84.13%

In this study, we first conducted a systematic exploration of two types of key hyperparameters, namely Batch Size and Learning Rate, and precisely depicted their influence on the performance of sentiment classification by controlling variables one by one. In terms of Batch Size, the experimental results (Table II) show that during the process when the Batch Size of the model was doubled successively from 8 to 64, the accuracy rate generally showed an upward trend: increasing from 0.8127 (Batch Size = 8) to 0.8306 (Batch Size = 64). This phenomenon can be attributed to the fact that a moderately increased batch size can not only reduce the variance of gradient estimation and stabilize the model update to a certain extent, but also will not cause a sharp decrease in the number of effective iterations due to excessive video memory occupation. It is notable that when the Batch Size is further increased to 128, the accuracy rate drops back to 0.8263 instead, which is consistent with the common "generalization deterioration" in large-scale training: Excessive batch size integrates too many sample signals in a single update, weakening the ability of parameters to "noise-driven" traverse saddle points, thereby reducing the performance of the test set. Comprehensively considering the performance peak and computing cost, Batch Size = 64 was determined as the optimal configuration.

The learning rate experiment (Table III) further reveals the significant influence of the optimized step size on the convergence quality. When the learning rate decreases from 0.5 to 0.05, the accuracy rate rises rapidly from 0.7526 to 0.8413, indicating that an overly large step size can cause the gradient update to cross the optimal region or even oscillate, while gradually reducing the step size can refine the parameter search and significantly improve the model discrimination ability. However, when the learning rate was further reduced to 0.02 and 0.01, the accuracy rates dropped to 0.8335 and 0.8316 respectively. This indicates that although a too small learning rate can avoid oscillations, it inhibits the ability to fully explore the parameter space and jump out of local minima, and slows down the convergence speed - thereby reducing the final accuracy. Therefore, considering the comprehensive training stability and performance, a learning rate of 0.05 was selected as the best. After the above two hyperparameters were optimized in place, the three-category evaluation index of the model on the test set (Table IV) showed that the overall accuracy rate reached 84.13%, while maintaining a relatively balanced category recall rate (positive 0.81, negative 0.79, neutral 0.83) and F1 score (in the range of 0.76-0.79). It

indicates that the parameter configuration takes into account the identification of different emotional tendencies, and there is no significant bias towards any category.

Further comparative experiments (Table V) compared Bert-BiLSTM with baselines such as BP, one-way LSTM and BiLSTM. Under the unified optimal hyperparameter Settings, Bert-BiLSTM established a leading advantage with an accuracy rate of 84.13%, a macro average recall rate of 0.81, and a macro average F1 score of 0.77. This result indicates that the superposition of BERT's pre-trained semantic representation and BiLSTM's bidirectional context capture ability significantly improves the comprehensive modeling effect of fine-grained semantics and emotional cues in investor comments. However, neither the shallow BP network with the same hyperparameters nor the one-way/two-way LSTM can achieve such accuracy on the same dataset.

D. Research on Stock Volatility Prediction Model

In general, the deeper the CNN layers are, the more abstract and high-level feature information the network can extract, and the more expressive the model can be. However, too deep networks also bring problems such as training difficulty, vanishing gradient and exploding gradient, which make training more difficult. In this study, the CNN network was constructed from a relatively shallow number of layers, and the input layer, 1 convolutional layer, and 1 pooling layer were initially set, and then the convolutional layers and pooling layers were gradually increased until 5 convolutional layers were reached. Since the amount of data is not large, a too deep network structure is not used. A total of five experiments were conducted this time, and the obtained results are shown in Table VI, where each index is the average value of the prediction evaluation of stocks, including New Hope.

TABLE VI THE AVERAGE SCORE OF THE RISE AND FALL OF DIFFERENT NUMBER OF HIDDEN LAYERS PREDICTION

Model types	Number of hidden layers	Accuracy	F1—score	AUC
CNN	1	0.58	0.58	0.59
	2	0.57	0.58	0.58
	3	0.57	0.57	0.58
	4	0.56	0.57	0.56
	5	0.57	0.57	0.56
LSTM	1	0.58	0.58	0.59
	2	0.57	0.58	0.58
	3	0.57	0.58	0.57
	4	0.56	0.57	0.56
	5	0.57	0.58	0.58

It can be seen from the data that the AUC score is the highest when the number of CNN network hidden layers is 1. As the number of hidden layers increases, the AUC score slowly decreases and then fluctuates between 0.58 and 0.56. According to the experimental results, when the number of hidden layers of the CNN of the model in this study is 1, it is the optimal solution to select the data set.

After adding one layer of LSTM, its AUC and accuracy decrease. As the number of hidden layers increases, the average AUC score fluctuates between 0.58 and 0.56. Therefore, 1 hidden layer is selected as the optimal solution in this study, that is, the number of LSTM hidden layers in the initial model is kept constant. In this section, a CNN-LSTM-AM hybrid neural network model is constructed based on the attention mechanism, CNN neural network, and LSTM neural network. Then, news and investor sentiment are integrated into the training data to train the CNN-LSTM-AM model and use it to predict the rise and fall of stock prices. However, the specific impact of the addition of attention mechanism, news and investor sentiment on the model is worth considering, so this section comprehensively analyzes the impact of these two factors on the model. The comprehensive performance comparison is shown in Table VII.

In general, after adding the features of news and investor comments, the four evaluation indicators of Accuracy, F1-score, Recall and AUC of each major model have different degrees of improvement. Specifically, after adding the sentiment features of news and investor comments, each evaluation index of the BP neural network model is increased by 0.025 on average. The average improvement of the CNN model indicators is 0.0325. Each evaluation index of the LSTM model was increased by 0.0275 on average. After adding news and comment features, the Accuracy, F1-score and AUC indicators of the CNN-LSTM model are increased by 0.03, and the Recall is increased by 0.02. The CNN-LSTM-AM model constructed in this study has the best effect. After adding news and comment features, the average AUC index reaches 0.64, and the Accuracy and F1-score both reach 0.63, which is an obvious improvement effect. When the sentiment features of news and investor comments are not added, the overall performance of CNN-LSTM-AM is also better than that of other comparison models. In summary, all models have significantly improved in each evaluation index after adding news and comment features. The CNN-LSTM-AM model performs better than the above comparison models. Therefore, it can be considered that the features of news and investor comments are effective features, which can be used to increase the accuracy of the model in stock prediction.

TABLE VII WHETHER TO ADD NEWS AND INVESTOR SENTIMENT PERFORMANCE COMPARISON (MEAN VALUE)

	Join or not	Accuracy	F1—score	Recall	AUC
BP	No	0.55	0.55	0.53	0.55
	Yes	0.57	0.57	0.56	0.58
CNN	No	0.58	0.57	0.57	0.59
	Yes	0.61	0.61	0.60	0.62
LSTM	No	0.58	0.59	0.57	0.59
	Yes	0.61	0.60	0.61	0.62
CNN—LSTM	No	0.59	0.58	0.58	0.60
	Yes	0.62	0.61	0.60	0.63
CNN—LSTM—AM	No	0.60	0.60	0.58	0.62
	Yes	0.63	0.63	0.61	0.64

E. Model Backtesting Experiment

This study uses CNN-LSTM-AM to mine the time series information in stock data, and combines news information and investor comment information in the stock forum. In this subsection, the CNN-LSTM-AM model of this study is compared with the traditional CNN model, LSTM model and CNN-LSTM model based on the use of the strategies in this study. After multiple parameter adjustments and optimizations,

Tables VIII to X list the backtest returns, maximum drawdown rates, and Sharpe rates of the selected 8 stocks from January 1, 2022 to January 1, 2025.

TABLE VIII COMPARISON OF BACKTEST RETURNS FOR SELECTED STOCKS

Stock	Benchmark	CNN-LSTM-AM	CNN-LSTM	CNN	LSTM
Everbright Securities	15.74%	108.55 %	85.22%	74.38%	78.82%
GoerTek Inc.	175.04%	268.69 %	178.44 %	148.78 %	169.37 %
Vanke A	-33.94%	17.37%	12.17%	6.25%	10.29%
New Hope Liuhe	-23.85%	78.38%	62.69%	56.48%	67.03%
Digital China Health	59.15%	98.45%	93.33%	73.16%	85.56%
Jinling Pharmaceutical	8.82%	57.14%	48.12%	46.63%	49.88%
Luzhou Laojiao	208.15%	412.35 %	369.18 %	343.81 %	398.82 %
Changan Automobile	110.91%	183.68 %	165.37 %	155.44 %	171.65 %

Table IX shows the comparison of the maximum drawdown rates in the simulation backtesting between the model in this study and other models. It can be seen from the table that the average maximum drawdown rate of the model in this study is 28.78%, and the maximum drawdown rates of stocks are all lower than those of the benchmark and other models. Overall, CNN-LSTM-AM reduces the maximum drawdown rate of the model.

TABLE IX COMPARISON OF MAXIMUM DRAWDOWN RATES FOR SELECTED STOCKS

Stock	Benchmark	CNN-LSTM-AM	CNN-LSTM	CNN	LSTM
Everbright Securities	52.49%	23.46%	24.30%	30.58%	28.13%
GoerTek Inc.	47.87%	32.19%	33.45%	39.34%	35.26%
Vanke A	44.30%	26.89%	31.41%	37.32%	33.51%
New Hope Liuhe	73.75%	36.61%	41.12%	46.41%	47.36%
Digital China Health	43.71%	21.58%	29.65%	38.36%	31.64%
Jinling Pharmaceutical	34.33%	24.81%	31.29%	35.84%	32.95%
Luzhou Laojiao	48.27%	30.34%	34.37%	42.54%	40.41%
Changan Automobile	49.92%	34.39%	41.29%	46.71%	43.62%
Average	49.33%	28.78%	33.36%	39.64%	36.61%

As shown in Table X, the average Sharpe ratio of CNN-LSTM-AM is 1.12, which not only exceeds the benchmark value of 0.43, but also is the highest among the four models. Especially in the backtesting of Vanke A and New Hope, the models successfully increased the benchmark Sharpe ratio, which was negative to positive.

TABLE X COMPARISON OF SHARPE RATIOS FOR SELECTED STOCKS

Stock	Benchmark	CNN-LSTM-AM	CNN-LSTM	CNN	LSTM
Everbright Securities	0.09	0.75	0.59	0.45	0.55
GoerTek Inc.	1.17	1.84	1.21	0.98	1.13
Vanke A	-0.66	0.49	0.37	0.21	0.28
New Hope Liuhe	-0.09	1.04	0.89	0.77	0.81
Digital China Health	0.52	0.79	0.60	0.47	0.51
Jinling Pharmaceutical	0.05	0.21	0.07	0.04	0.06
Luzhou Laojiao	1.40	2.42	2.13	1.82	1.93
Changan Automobile	0.94	1.39	1.28	1.14	1.22
Average	0.43	1.12	0.89	0.74	0.81

Based on the above-mentioned experimental results, it is confirmed that the stock trading model based on CNN-LSTM-AM is a feasible quantitative timing strategy. Compared with the CNN, LSTM and CNN-LSTM models, this model shows a higher average backtest yield and Sharpe rate, as well as a lower maximum drawdown rate. This means that this model can achieve higher returns while undertaking relatively lower risks, providing certain references for users' investment decisions.

V. CONCLUSION

A. The Content and Significance of this Study

This study set out to investigate how an improved CNN-LSTM architecture can be leveraged—together with modern language-representation techniques and sentiment factors—to advance open-source intelligence (OSINT) mining, emergent-event detection, and stock-price fluctuation prediction in the financial domain. By unifying local-feature perception in convolutional layers with the long-range dependency modeling capacity of LSTM, the proposed CNN-LSTM topic-crawler raises both the harvest rate and the mean topical relevance of crawled documents, outperforming vanilla CNN, LSTM, and classical TF–IDF baselines. These gains materially reduce the latency between news release and intelligence ingestion, a critical requirement for time-sensitive financial analysis. The hierarchical detector first filters non-emergent content and then refines the categorization of bona fide sudden events by fusing word-vector dispersion features that encode class–term association strength and contextualized representations from BERT. Coupling these dual channels with an attention-augmented TextCNN boosts F1 scores across all emergent categories, validating the merit of jointly capturing local salience and global semantics in highly volatile news streams. Integrating news sentiment, investor-comment sentiment (obtained via a BERT-BiLSTM classifier), and historical trading factors into a convolutional–recurrent network enhanced with an attention mechanism (CNN-LSTM-AM) yields superior directional-accuracy and RMSE metrics relative to single-source or single-architecture benchmarks.

B. Significance of Research

Taking the financial news text as the breakthrough point, this study deeply discusses the application path and practical effect of the text processing technology based on the improved CNN-LSTM model in financial intelligence mining, which has significant theoretical and application value. By introducing the hybrid architecture of CNN and LSTM, combined with its advantages in local feature extraction and temporal dependency modeling, the topic news crawler system constructed in this study realizes the efficient classification and accurate collection of financial news texts. Compared with the traditional crawler system based on keywords or rule-driven, the system greatly improves the acquisition efficiency and relevance of target information, and provides strong technical support for the construction of a high-quality intelligence database in the financial field. As an important external variable affecting the volatility of the stock market, the rapid identification and classification of emergencies is an important part of the financial intelligence system.

For the future, the research can be deepened along the following directions. Firstly, a multilingual and multi-market cross-domain financial intelligence knowledge base was constructed, and an incremental online learning framework was introduced to realize the rapid adaptation and transfer generalization of the model in complex heterogeneous scenarios. Self-supervised contrastive learning and joint fine-tuning strategies are combined to mitigate the performance degradation caused by cross-lingual semantic drift. Secondly, the data modality and interaction dimension should be expanded: in addition to news and text public opinion, image, audio and transaction depth data should be fused, and multimodal Transformer and spatio-temporal graph neural network should be introduced to jointly model the market microstructure signal end-to-end, so as to improve the prediction strength of sudden black swan events and complex chain reactions. Furthermore, combined with symbolic reasoning and a knowledge graph, financial regulations, macro policies, and enterprise knowledge can be embedded into the event extraction and sentiment analysis process, and the causal inference ability and explanation depth of the model can be enhanced by executable logic rules.

ACKNOWLEDGMENT

Thanks for the data support provided by the Project of English Digital Resources Construction Institute (No.: GKY-2022CQJG-9)

REFERENCES

[1] K. B. Hansen and C. Borch, "Alternative data and sentiment analysis: Prospecting non-standard data in machine learning-driven finance," Big Data & Society, vol. 9, no. 1, p. 20539517211070701, 2022.

[2] X. Luo, H.-H. Chen, and Q. Guo, "Semantic communications: Overview, open issues, and future research directions," IEEE Wireless Communications, vol. 29, no. 1, pp. 210–219, 2022.

[3] S. Sajid, "Anticipating The Stock Market Trend Using Natural Language Processing Techniques," Electrical Engineering in School of Electrical Engineering & Computer …, 2024.

[4] A. Peivandizadeh et al., "Stock market prediction with transductive long short-term memory and social media sentiment analysis," IEEE Access, 2024.

[5] K. Du, F. Xing, R. Mao, and E. Cambria, "Financial sentiment analysis: Techniques and applications," ACM Computing Surveys, vol. 56, no. 9, pp. 1–42, 2024.

[6] A. C. Ikegwu, H. F. Nweke, C. V. Anikwe, U. R. Alo, and O. R. Okonkwo, "Big data analytics for data-driven industry: a review of data sources, tools, challenges, solutions, and research directions," Cluster Computing, vol. 25, no. 5, pp. 3343–3387, 2022.

[7] C. Kyrkou, P. Kolios, T. Theocharides, and M. Polycarpou, "Machine learning for emergency management: A survey and future outlook," Proceedings of the IEEE, vol. 111, no. 1, pp. 19–41, 2022.

[8] R. K. Behera, M. Jena, S. K. Rath, and S. Misra, "Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data," Information Processing & Management, vol. 58, no. 1, p. 102435, 2021.

[9] L. Khan, A. Amjad, K. M. Afaq, and H.-T. Chang, "Deep sentiment analysis using CNN-LSTM architecture of English and Roman Urdu text shared in social media," Applied Sciences, vol. 12, no. 5, p. 2694, 2022.

[10] B. Chakravarthi, S.-C. Ng, M. Ezilarasan, and M.-F. Leung, "EEG-based emotion recognition using hybrid CNN and LSTM classification," Frontiers in computational neuroscience, vol. 16, p. 1019776, 2022.

[11] M.-F. de-Lima-Santos, L. Mesquita, J. G. de Melo Peixoto, and I. Camargo, "Digital news business models in the age of industry 4.0: Digital Brazilian news players find in technology new ways to bring revenue and competitive advantage," Digital Journalism, vol. 12, no. 9, pp. 1304–1328, 2024.

[12] D. Giomelakis, "Semantic search engine optimization in the news media industry: Challenges and impact on media outlets and journalism practice in Greece," Social Media+ Society, vol. 9, no. 3, p. 20563051231195545, 2023.

[13] K. Sharifani, M. Amini, Y. Akbari, and J. Aghajanzadeh Godarzi, "Operating machine learning across natural language processing techniques for improvement of fabricated news model," International Journal of Science and Information System Research, vol. 12, no. 9, pp. 20–44, 2022.

[14] K. K. Kiilu, "Sentiment Classification for Hate Tweet Detection in Kenya on Twitter Data Using Naïve Bayes Algorithm," JKUAT-COETEC, 2021.

[15] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning--based text classification: a comprehensive review," ACM computing surveys (CSUR), vol. 54, no. 3, pp. 1–40, 2021.

[16] Y. Yan, Y. Kang, W. Huang, and X. Cai, "Chinese medical named entity recognition utilizing entity association and gate context awareness," PloS one, vol. 20, no. 2, p. e0319056, 2025.

[17] N. Chandra, L. Ahuja, S. K. Khatri, and H. Monga, "Utilizing gated recurrent units to retain long term dependencies with recurrent neural network in text classification," J. Inf. Syst. Telecommun, vol. 2, p. 89, 2021.

[18] G. Duan, S. Yan, and M. Zhang, "A Hybrid Neural Network Model for Sentiment Analysis of Financial Texts Using Topic Extraction, Pre-Trained Model, and Enhanced Attention Mechanism Methods," IEEE Access, 2024.

[19] H. Chen, L. Wu, J. Chen, W. Lu, and J. Ding, "A comparative study of automated legal text classification using random forests and deep learning," Information Processing & Management, vol. 59, no. 2, p. 102798, 2022.

[20] A. Mohammed and R. Kora, "An effective ensemble deep learning framework for text classification," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 10, pp. 8825–8837, 2022.

[21] Y. Yue, Y. Peng, and D. Wang, "Deep learning short text sentiment analysis based on improved particle swarm optimization," Electronics, vol. 12, no. 19, p. 4119, 2023.

[22] H. Zhao and W. Xiong, "A multi-scale embedding network for unified named entity recognition in Chinese Electronic Medical Records," Alexandria Engineering Journal, vol. 107, pp. 665–674, 2024.

[23] Z. Wang, L. Wang, C. Huang, S. Sun, and X. Luo, "BERT-based chinese text classification for emergency management with a novel loss function," Applied Intelligence, vol. 53, no. 9, pp. 10417–10428, 2023.

[24] A. R. A. Aljanabi, "The impact of economic policy uncertainty, news framing and information overload on panic buying behavior in the time of COVID-19: a conceptual exploration," International Journal of Emerging Markets, vol. 18, no. 7, pp. 1614–1631, 2023.

[25] C. E. d. Matta, N. M. P. Bianchesi, M. S. d. Oliveira, P. P. Balestrassi, and F. Leal, "A comparative study of forecasting methods using real-life econometric series data," Production, vol. 31, p. e20210043, 2021.

[26] B. Muma and A. Karoki, "Modeling GDP Using Autoregressive Integrated Moving Average (ARIMA) Model: A Systematic Review," Open Access Library Journal, vol. 9, no. 4, pp. 1–8, 2022.

[27] V. I. Kontopoulou, A. D. Panagopoulos, I. Kakkos, and G. K. Matsopoulos, "A review of ARIMA vs. machine learning approaches for time series forecasting in data driven networks," Future Internet, vol. 15, no. 8, p. 255, 2023.

[28] A. Shen, M. Dai, J. Hu, Y. Liang, S. Wang, and J. Du, "Leveraging Semi-Supervised Learning to Enhance Data Mining for Image Classification under Limited Labeled Data," arXiv preprint arXiv:2411.18622, 2024.

[29] G. Sonkavde, D. S. Dharrao, A. M. Bongale, S. T. Deokate, D. Doreswamy, and S. K. Bhat, "Forecasting stock market prices using machine learning and deep learning models: A systematic review, performance analysis and discussion of implications," International Journal of Financial Studies, vol. 11, no. 3, p. 94, 2023.

[30] B. T. Khoa and T. T. Huynh, "Forecasting stock price movement direction by machine learning algorithm," International Journal of Electrical and Computer Engineering (IJECE), vol. 12, no. 6, pp. 6625–6634, 2022.

[31] X. Liu, J. Guo, H. Wang, and F. Zhang, "Prediction of stock market index based on ISSA-BP neural network," Expert Systems with Applications, vol. 204, p. 117604, 2022.

[32] V. Kumar, "Data Analytics and Supply Chain Management: Leveraging Big Data to Optimize Production and Logistics in Fashion and Textiles," in Use of Digital and Advanced Technologies in the Fashion Supply Chain: Springer, 2025, pp. 89–105.

[33] J. G. George, "Advancing Enterprise Architecture for Post-Merger Financial Systems Integration in Capital Markets laying the Foundation for Machine Learning Application," Aus. J. ML Res. & App, vol. 3, no. 2, p. 429, 2023.

[34] M. Sun et al., "Thuctc: an efficient chinese text classifier," GitHub Repository, 2016.