Chinese Financial Comments Sentiment Detection Based on the Bert-TCN Model Based on HowNet Disambiguation

Linhan Xia

Beijing Normal University-Hong Kong Baptist University United International College, Zhuhai, China, 519087

temp.xialinhan@gmail.com

Abstract. The introduction of investor sentiment index has set an important milestone in quantitative research within the financial industry. The index became a powerful tool for extracting insights from investors' opinions about future financial markets. In order to solve the problem of ambiguity in Chinese financial news sentiment analysis, we design and implement a BERT model based on HowNet. This study not only provides a comparative analysis of the performance of multiple models such as LSTM, RNN, CNN and BERT on sentiment classification tasks, but also delves into the actual impact of ambiguity resolution on classification results. On top of this, it also conducts a subsequent empirical test of stock prices using sentiment scores. The experimental results verify the excellent performance of our model in sentiment classification, especially in dealing with ambiguity, which provides a strong impetus for sentiment analysis and stock prediction in the Chinese financial field. Overall, this research opens a completely new path for the application of sentiment analysis and prediction in the Chinese financial industry.

Keywords: NLP, Bert, TCN, HowNet, WSD

1 Introduction

As the core asset of China's financial market, stocks are known for their liquidity and potential high returns. Accurate prediction of stock prices has become a key issue in the field of financial research, which is a necessary means for investors to control risks and capture investment opportunities. As a kind of unstructured data, sentiment data on social networks can reflect and drive the overall sentiment of the market, which is an important basis for stock price prediction. The author proposes an innovative prediction method, which combines sentiment features extracted by an optimized BERT model with stock price features for time series prediction. He compares the performance of the optimized BERT model with that of LSTM, RNN, TCN, CNN and other neural network models on sentiment classification [1][2][3][4]. In addition, he uses a Chinese WordNet (HowNet) based disambiguation task to solve

C. Chen et al. (eds.), *Proceedings of the 3rd International Conference on Digital Economy and Computer Application (DECA 2023)*, Atlantis Highlights in Computer Sciences 17,
https://doi.org/10.2991/978-94-6463-304-7_18

the ambiguity problem of Chinese blogs in sentiment classification tasks, and compare the sentiment classification performance of disambiguated and undisambiguated texts. Finally, the author fuses the disambiguated text classification with stock price data to predict stock prices.

2 Related Work

2.1 Stock Price Prediction

Deep neural networks, due to their feature extraction and complex data relationship modeling abilities, have been proven effective in financial time series analysis [5]. The abundance of stock market trading data has boosted deep neural network training and predictions [6][7]. Specifically, the multivariate input LSTM model demonstrates superior accuracy in predicting stock prices and indices. Other combined models like LSTM-CNN also excel in stock market predictions. The aGRU basic model, proposed for predicting stock prices of certain companies and indices, displays improved fitting capacity and reduced errors [8]. New neural network models like the Time Convolutional Neural Network (TCN) have emerged, demonstrating quicker processing speed and higher accuracy in most RNN problems, offering a fresh approach to time series prediction. This study, inspired by the TCN model's performance, proposes a novel method to extract stock price features using deep neural networks, expected to enhance predictive accuracy and model complex financial data relationships [9].

2.2 Emotional Feature Extraction from Financial Forum Text

Emotion analysis often involves three main approaches: machine learning, neural network techniques, and pre-trained models for natural language processing. While machine learning algorithms like Support Vector Machine (SVM) are commonly used for emotion classification, they tend to overlook context information, leading to suboptimal results [10]. In finance, neural network-based strategies are prevalent for extracting and classifying sentiment features. This study focuses on a deep comparison of LSTM, TCN, RNN, and GRU models; however, their limited grasp of semantic information may hinder generalization. Conversely, pre-trained NLP models, particularly BERT, have gained widespread popularity and are frequently employed in various NLP tasks. To address this, the author developed a new ternary classification model for BERT fine-tuning and compared it with four neural network models to select the most practical option.

2.3 Chinese Text Disambiguation

A significant issue in pre-training natural language models is the ambiguity often found in Chinese sentences, which affects sentiment feature extraction accuracy. To address this, the author used a Hownet-based algorithm proposed by Xiao Li and

Qingsheng Li for Chinese text disambiguation, and combined it with the BERT model for sentiment feature extraction and classification [11]. Hownet, a semantic knowledge base, offers extensive vocabulary, meanings, and concept information. The algorithm used effectively addresses Chinese text polysemy, and when combined with the BERT model, proves valuable for sentiment feature extraction and classification tasks. The BERT model, trained with disambiguated Chinese and English text, demonstrated good performance in sentiment classification tasks, surpassing other methods in classification accuracy. This method also showed higher classification accuracy and better generalization ability when compared with commonly used Chinese text sentiment classification algorithms.

3 Text Emotional Feature Extraction and Classification Model

This study uses the multivariate dataset from the Eastern Financial Forum to conduct a comprehensive evaluation of LSTM, RNN, TCN, Bi-LSTM, GRU and BERT models. These data sets involve many stocks such as North International, ZTE, and Changhong Group. We split the dataset according to the proportion of positive examples (50%, 75%, and 90%) for subsequent training. The workflow of sentiment classification starts with embedded encodings for natural language processing, transforming raw text data into a dense vector form. This paper adopted Word2Vec, a pre-trained embedding vector method, to encode the text information so as to preserve the semantic similarity between words. Then it used this encoded text data as input to train and test various deep learning models including LSTM, RNN, TCN, Bi-LSTM, GRU, and BERT. In the model training phase, this study choose cross-entropy as the loss function and use Adam optimizer to optimize the model. After comparing experiments on all datasets, we determined the best model in the sentiment classification task.

$$H(p,q) = -\sum_{i=1}^{n} p(x_i)\log\left(q(x_i)\right) \tag{1}$$

3.1 Bidirectional Encoder Representations from Transformers

BERT's pre-training process involves two stages: unsupervised pre-training on large-scale text data, followed by supervised fine-tuning for specific tasks, adapting to varied application scenarios. This study utilizes the pre-training model and fine-tune it for the three tasks outlined in the paper(Table 1).

The BERT model uses a combination of three embedding methods: word, position and paragraph to express the input [11]. With a vector with 768 dimensions, the word embedding method cuts the word into multiple sub-word units with the help of WordPiece (WordPiece is a tokenization strategy for breaking input text into smaller, manageable fragments or tokens) vocabulary. Using the form of sine and cosine functions, the position embedding way provides specific information for each position. Paragraph embedding, on the other hand, uses another 768-dimensional vector to explicitly label each sentence to which part (A or B) it belongs.

$$E = E_{word} + E_{position} + E_{segment} \tag{2}$$

BERT's pre-training process actually pursues two main goals: masked language Model (MLM) and next Sentence prediction (NSP). First, in the MLM stage, the system randomly selects some words from the input sequence to hide, and then uses the surrounding context information to guess what the hidden words are. Secondly, in the NSP step, when two sentences are provided, the task is to decide whether the two sentences can be coherently followed.

$$L_{MLM} = -\frac{1}{N}\sum_{i=1}^{N} \log \mathrm{P}(w_i|C)$$

$$L_{NSP} = -\ logP(I|A, B) \tag{3}$$

BERT is built on top of a transformer encoder consisting of multiple layers of self-attention and feed-forward neural network layers. Using the multi-head attention mechanism, the self-attention layer has the ability to measure the relevance of each position in the input sequence to the rest of the positions. After that, the feedforward neural network layer performs a nonlinear transformation of the output from the self-attention layer by applying two fully connected networks and an activation function.

$$\begin{aligned}&\mathrm{MultiHead}(\mathrm{Q}, \mathrm{K}, \mathrm{V})\\ &= \mathrm{Concal}(\mathrm{head}_1, \ldots, \mathrm{head}_\mathrm{h})\mathrm{W}^\mathrm{O}\end{aligned}$$

$$\mathrm{head}_\mathrm{i} = \mathrm{Attention}(\mathrm{QW}_\mathrm{i}^\mathrm{Q}, \mathrm{KW}_\mathrm{i}^\mathrm{K}, \mathrm{VW}_\mathrm{i}^\mathrm{V})$$

$$\mathrm{Attention}(\mathrm{Q}, \mathrm{K}, \mathrm{V}) = \mathrm{softmax}(\frac{\mathrm{QK}^\mathrm{T}}{\sqrt{\mathrm{d}_\mathrm{k}}})\mathrm{V}$$

$$\mathrm{FFN}(\mathrm{x}) = \max(0, \mathrm{xW}_1 + \mathrm{b}_1)\,\mathrm{W}_2 + \mathrm{b}_2 \tag{4}$$

Sentiment analysis involves detecting and categorizing emotions expressed in a text, including positive, negative, and neutral sentiments. BERT, through pre-training and fine-tuning, demonstrates adaptability to diverse domains and tasks, enabling it to handle ambiguous emotional text to a certain degree (refer to Figure 1).

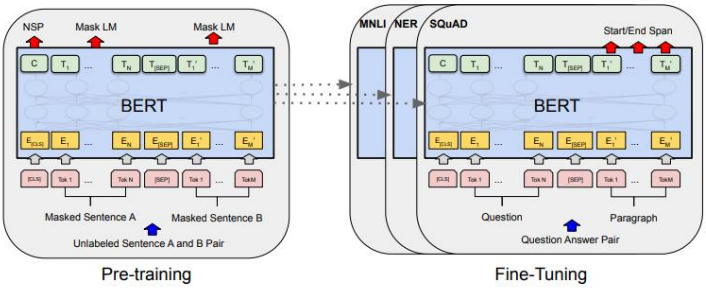

Fig. 1. Fine-tuning of bert model (This paper uses labelled data sets of emotion tripartite to fine-tune it)

Table 1. The architecture of model

Layer (type)	Output Shape	Param
bert (TFBertMainLayer)	multiple	102,267,648
dropout421 (Dropout)	multiple	0
classifier (Dense)	multiple	2,307
Total params: 102,269,955		
Trainable params: 102,269,955		
Non-trainable params: 0		

3.2 A Variety of Neural Network Model Experiments

This paper explores the utilization of different neural network models for text sentiment feature mining and classification. To assess their effectiveness, we train all models on the same training set and evaluate their performance on the test set(Table 2 and Table 3). The criteria for evaluation encompass the following aspects:

Accuracy: Accuracy refers to the fraction of all instances that the model predicts correctly. This is one of the most commonly used evaluation metrics, especially when the dataset is class-balanced.

$$\text{Accurancy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{5}$$

Precision: Precision signifies the fraction of instances that are genuinely positive amongst those predicted to be so by the model. This measure addresses the issue where the model overly favors a particular class.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{6}$$

Recall rate: Recall rate is the fraction of instances that are genuinely positive and are also predicted to be so by the model. The retrieval rate aids in overcoming the problem of missed detection.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{7}$$

F1-Score: F1-Score is the weighted mean of accuracy and recall rate. This indicator considers both accuracy and recall rate simultaneously. It offers a balanced choice between accuracy and recall rate within the same numerical range.

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{8}$$

AUC-roc: In a two-dimensional coordinate system, the construction of the ROC curve depends on two parameters, the horizontal axis is the false alarm rate (FPR), and the vertical axis is the true report rate (TPR). AUC describes the area covered under the ROC curve. When evaluating a binary classification problem, AUC-ROC is a powerful tool that accurately measures how well a model classifies samples.

$$\mathrm{AUC - ROC} = \int_0^1 \mathrm{TPR(FPR)dFPR} \tag{9}$$

Table 2. Fifty percent positive examples of each model (a)

Model	Accuracy	Precision(Positive)	Recall(Positive)	F1-Score(Positive)	Precision(Normal)	Recall(Normal)	F1-Score(Normal)	Precision(Negative)	Recall(Negative)	F1-Score(Negative)	AUC-ROC
RNN	45.10%	40.13%	43.23%	38.52%	44.34%	44.46%	44.35%	44.06%	44.06%	44.67%	0.4749
LSTM	55.30%	54.23%	54.41%	54.28%	54.28%	54.55%	55.15%	55.07%	55.07%	55.10%	0.61
Bi-LSTM	52.70%	51.09%	51.88%	50.65%	51.99%	52.52%	53.14%	52.06%	52.55%	52.55%	0.5792

Table 3. Fifty percent positive examples of each model (b)

Model	Accuracy	Precision(Positive)	Recall(Positive)	F1-Score(Positive)	Precision(Normal)	Recall(Normal)	F1-Score(Normal)	Precision(Negative)	Recall(Negative)	F1-Score(Negative)	AUC-ROC

Attention-LSTM	51.70%	50.23%	51.39%	49.56%	51.38%	52.31%	51.48%	51.48%	50.73%	51.10%	0.5612
CNN	48.50%	45.64%	48.49%	45.07%	48.26%	50.17%	49.02%	49.02%	47.05%	48.02%	0.5273
TCN	59.60%	58.55%	58.49%	58.52%	58.93%	58.88%	59.14%	59.14%	59.19%	59.16%	0.6478
BERT	79.00%	79.23%	78.02%	79.06%	79.38%	79.42%	78.91%	78.91%	79.30%	78.91%	0.8725

This research reveals the superiority of the BERT model over other neural network models across various metrics. BERT's cutting-edge position in natural language processing tasks can be attributed to its unique features such as pre-training, bidirectionality, the incorporation of the Transformer architecture, fine-tuning abilities, and multilingual support. These attributes allow BERT to excel in handling natural language data, thereby making it one of the most sought-after language models in current times. Consequently, this study opts for the BERT model to carry out sentiment feature extraction and categorization for stock price forecasting in this work.

4 Chinese Sentence Disambiguation Model Based on HowNet

In a bid to enhance the model and its generalization capabilities, this work introduces a disambiguation model built upon Chinese WordNet (HowNet) [12]. The goal is to amplify the precision of sentiment feature extraction and categorization assignments by effectively implementing disambiguation of Chinese texts.

4.1 Chinese WordNet: HowNet

HowNet serves as an artificial intelligence knowledge base, explicitly crafted to encapsulate the semantic details pertaining to words. It hinges on the principle of 'sememes,' whereby the essence of a word is broken down into multiple fundamental semantic components, each corresponding to a specific Chinese term. The interaction among these elements is termed the 'sememes relationship,' collectively generating an extensive lexicon knowledge network. Within the structure of HowNet, Chinese words take form through various semantic representations, comprised of one or more sememes [12]. These sememes represent the minimal units employed in concept description.

4.2 Ambiguity Removal Based on HowNet

HowNet, by only considering individual words, doesn't fully capture the precise semantics of a complete Chinese text. To address this shortcoming, the author introduces a disambiguation algorithm in this research. This method utilizes the open-source tool, LTP, developed by Harbin Institute of Technology's Natural Language Processing Center for Chinese sentence partitioning and part-of-speech identification [13]. LTP dissects Chinese text into single words and offers syntactic and semantic interdependencies. The proposed disambiguation algorithm weighs the fluidity and wholeness of semantics emerging from different notions within Chinese sentences. An elevated semantic fluidity suggests a higher probability of aligning with the original Chinese text semantics. The integrity and fluidity of sentence semantics are assessed using the relations and distance of sentence dependencies. StanfordCoreNLP, an open-source tool from Stanford's Natural Science Research Office, is employed to analyze the semantic dependency distance in Chinese sentences [14]. The pseudo-code explaining this process is also made available in this work.

Algorithm 1 Calculate Smoothness Score

```
function CAL_sm_sc(DEPENDENCY_PARSE)
  weights ← {('nsubj', 1), ('csubj', 1), ('amod', 0.5)}
  weights ← {('prep', 0.5), ('conj', 0.5), ('punct', 0.1))}
  weights ← {('root', 0), ('det', 0.5), ('compound', 0.5)}
  score ← 1
  FOR i = 0 TO len(dependency_parse) − 1 DO
    word ← dependency_parse[i][0]
    rel_type ← dependency_parse[i][1]
    parent_index ← dependency_parse[i][2]
    IF parent_index = −1 THEN
      CONTINUE
    END IF
    path_length ← CAL_SM_SC(dep_par, i, par_i − 1)
    weight ← weights.get(rel_type, 0)
    IF path_length > 2 THEN
      score ← score × weight × 0.1
    ELSE
      score ← score × weight
    END IF
  END FOR
  RETURN score
END FUNCTION
```

This work elucidates the roles of LTP and the semantic dependency analysis algorithm, both fundamental to the Chinese text disambiguation algorithm rooted in HowNet. The disambiguation algorithm proceeds as follows (as shown in Figure 2):

First, statements needing disambiguation are processed and divided by LTP, yielding Part of Speech (POS) information. Second, based on LTP's segmentation results, each word is matched to a Concept within HowNet. If there exists only one Concept at this point (the word only has one mean), it's integrated into the sentence semantics. If there are multiple concepts, the algorithm proceeds to the third step. Third, words associated with numerous concepts are extracted, and all possible concepts are listed. Through traversing combinations of all concepts, several Chinese text semantic sequences are obtained. Finally, for Chinese text embodying multiple semantic sequences, this study employs the aforementioned semantic dependency analysis algorithm to discern the semantic sequence exhibiting optimal continuity and fluency.

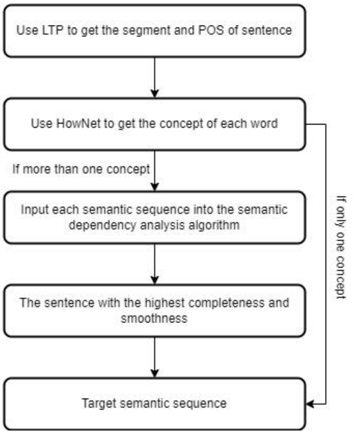

Fig. 2. The algorithm flow proposed in this paper

4.3 Performance Comparison Before and After Disambiguation Based on BERT Model

Once the sentence disambiguation process concludes, two Chinese text datasets will be prepared for training on the BERT model (which has already been validated as optimal for this dataset). One dataset will contain post-disambiguation Chinese text and the other will consist of non-disambiguated Chinese text. The performance of the identical model on these two datasets will then be evaluated and compared.

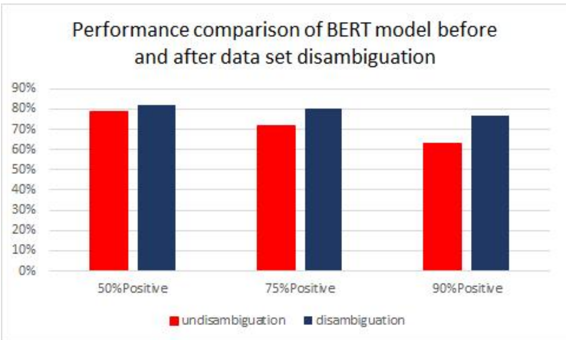

Fig. 3. Comparative results of the generalization accuracy of the BERT model trained on two datasets with different proportions of positive samples

From accuracy prediction, it's evident that Chinese text disambiguation enhances the model's capacity for generalization. As a result, this study applies the disambiguation-BERT model in conjunction with the previously discussed TCN model for predicting stock assets (as depicted in Figure 3).

5 Stock Price Prediction Model Integrating Emotional Characteristics and Classification

This study introduces the Bert-TCN prediction model, primarily built around modules for sentiment feature extraction and categorization in Chinese text, as well as the extraction of stock price features and an integrated prediction module.

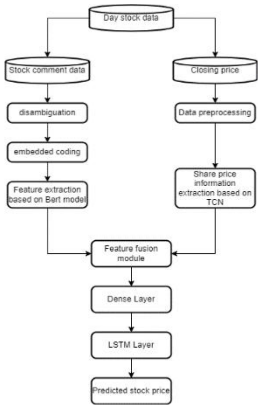

Fig. 4. The flow chart of feature fusion prediction model is presented in this paper

Initially, this research utilizes web scraping techniques to gather individual stock comments from different trading days on the Eastern Wealth Individual Stock Forum. Following word segmentation, POS tagging, and disambiguation, the BERT model comes into play for the extraction and classification of emotional features, consequently yielding emotional features for the respective trading day. Concurrently, the corresponding closing stock price undergoes feature extraction using the TCN model. Ultimately, the attention mechanism undertakes feature integration, while the LSTM time series neural network forecasts the stock price for the corresponding trading day, yielding the predicted closing price of the stock (as illustrated in Figure 4).

5.1 Feature Fusion Based on Attention Mechanism

In the context of neural network models, the attention mechanism serves as a system enabling the network model to prioritize critical information when handling diverse inputs. It aids the network in the execution of selective attention injection, dynamic adjustment, interference reduction, and enhancement of interpretability. In this study, the attention mechanism is harnessed to amalgamate the emotional attributes of stock comments with stock price characteristics.

This study regards the stock price attributes as the key value, and the sentiment features of stock comments from the trading day as the query value (in this study, sentiment features equate to the three-category output of the Bert model). By computing the correlation or similarity between corresponding query values, the appropriate attention weight is determined. This is then normalized using the softmax function, and when summed with the corresponding key value, the attention value is obtained (as demonstrated in Figure 5).

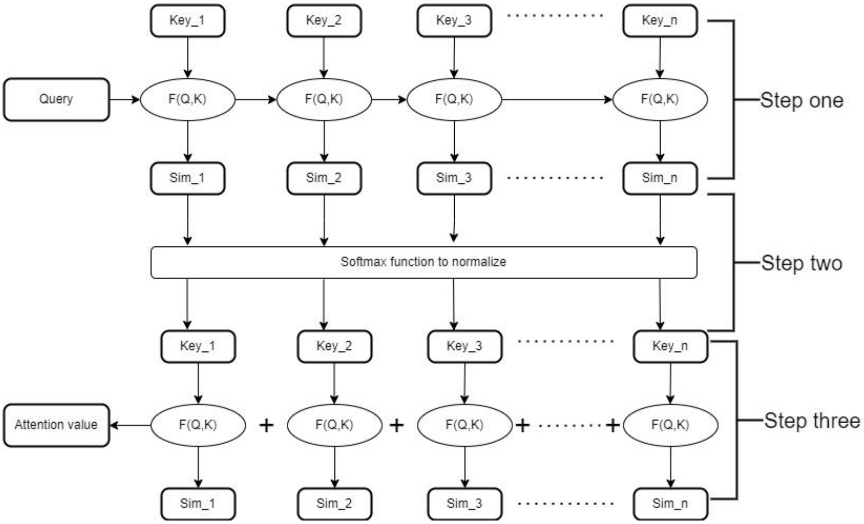

Fig. 5. Diagram of fusion of emotional features

Step 1: Calculate each query and each Key to get the weight coefficient:

$$\mathrm{Sim}\left(\mathrm{Q}, \mathrm{K}_{\mathrm{i}}^{\mathrm{T}}\right) = \mathrm{Q} \cdot \mathrm{K}_{\mathrm{i}}^{\mathrm{T}} \tag{10}$$

Step 2: softmax was used to normalize the weight coefficient:

$$\mathrm{alpha}_{\mathrm{i}} = \mathrm{Softmax}(\mathrm{c}^{\mathrm{Sim}_{\mathrm{i}}}) = \frac{\mathrm{c}^{\mathrm{Sim}_{\mathrm{i}}}}{{}_{\mathrm{j}=1}^{\mathrm{L_x}} \mathrm{e}^{\mathrm{Sim}_{\mathrm{i}}}} \tag{11}$$

Step 3: Sum the weight and key values to get the final attention value:

$$A(Q, K, V) = {}_{i=1}^{L_x} \alpha_i \cdot V_i \tag{12}$$

5.2 Experiment Data

Two main datasets are utilized in this research: stock data and Eastern Wealth Review data, covering the time span from March 18, 2020, to March 3, 2023. Following preprocessing, the data is further divided into distinct training and test sets. The study collects data for 13 stocks from the Shenzhen, Shanghai, and GEM exchanges via the wind platform, focusing on essential characteristics such as opening price, closing price, highest price, lowest price, turnover, and turnover rate (refer to Figure 6 and Figure 7).

Stock Code	Stock Name
SZ000001	Ping An Group
SZ000009	Bao'an China
SZ000058	Shenzhen SEG
SZ000060	Zhongjin Lingnan
SZ000063	ZTE
SZ002251	Backgammon
SZ300463	mike bio
SH600004	Baiyun Airport
SH600009	Shanghai Airport
SH600958	Orient Securities
SH688114	MGI
SH688981	SMIC
SH605028	Shimao Energy

Fig. 6. Data sources used to train stock price feature extraction

Stock Code	Stock Name	Number of comment
SZ000001	Ping An Group	36408
SZ000009	Bao'an China	67245
SZ000058	Shenzhen SEG	45624
SZ000060	Zhongjin Lingnan	38472
SZ000063	ZTE	89584
SZ002251	Backgammon	49037
SZ300463	mike bio	27612
SH600004	Baiyun Airport	15367
SH600009	Shanghai Airport	18923
SH600958	Orient Securities	28763
SH688114	MGI	10983
SH688981	SMIC	59863
SH605028	Shimao Energy	17860

Fig. 7. Data sources used to extract investor sentiment feature extraction

5.3 Evaluation Index Selection

In order to evaluate the improvement of the training performance of the model proposed in this paper, MAE, RMSE, MAPE and R^2 are selected according to the stock data prediction evaluation indicators proposed by Jin et al.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_i|$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i)^2}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - y_i}{y_i}\right| \times 100\%$$

$$R^2 = 1 - \frac{{}_{i=1}^{n}(y_i - y_i)^2}{{}_{i=1}^{n}(y_i - y_i)^2} \tag{13}$$

5.4 Stock Price Model Experiment and Analysis

(1) Experimental Content and Results

In assessing the potency of our proposed model, we opted for a comparative study involving the use of LSTM, GRU, TCN, Bert-LSTM (an LSTM model enriched with sentiment features), Bert-GRU (a GRU model merged with sentiment features), along with the newly proposed Bert-TCN model. This set of experiments aimed to predict the closing prices of six different stocks. In these models, the last three incorporated the mean sentiment value from the preceding five trading days prior to the targeted stock price date. To eliminate the factor of accidental experimental outcomes, multiple trials were carried out, and an average value was derived. The detailed outcomes of the experiments will be displayed further in the text, with the best prediction results highlighted in bold. As is evident, the Bert-TCN model exhibits superior predictive performance compared to other models.

(2) Comprehensive Analysis of Experimental Results

Results As depicted in Tables 4 and 5, the predictive performance of the TCN model surpasses that of LSTM and GRU models. It presents lower MAE, RMSE, and MAPE values, while its R2 values are nearing one. The superior performance of the TCN model can be attributed to the causal links among its convolutional layers, facilitating the retention of historical data and preventing information loss. Compared to the standalone LSTM, GRU, and TCN models, as well as the sentiment feature-enhanced Bert-TCN models, there is a marked improvement in various performance metrics. This suggests that the integration of sentiment features can potentially enhance the accuracy of stock price prediction. Moreover, the Bert-TCN model's predictive results for the closing prices of six stocks across diverse sectors and industries outperform those of baseline models, indicating superior predictive performance of the Bert-TCN model.

Table 4. Experimental Results (a)

Model	Evaluation index	Ping An Group	Bao An China	Shenzhen SEG	Zhongjin Lingnan	ZTE	Backgammon	Mike bio	Baiyun airport	Shanghai Airport
LSTM	MAE	3.6253	3.8745	5.1639	4.9956	2.5783	5.2731	2.2132	5.1873	3.2372
	RMSE	5.67	5.85	6.83	6.47	4.85	6.56	3.94	6.37	4.26
	MAPE	6.97	6.98	6.75	6.9	4.64	6.89	3.44	6.53	4.12
	R^2	0.556	0.55	0.468	0.533	0.684	0.491	0.749	0.464	0.613
GR	MAE	2.6129	5.8796	4.1781	4.9802	6.5317	4.2298	5.1765	1.2141	2.2576

U	RMSE	5.588	5.9435	6.6541	6.7223	5.6876	3.6713	3.0207	4.8802	6.4135
	MAPE	4.864	7.9127	4.6534	4.4876	6.4735	4.9275	5.3906	2.3335	2.0604
	R^2	0.5574	0.5493	0.4965	0.5349	0.6805	0.4892	0.7529	0.4647	0.6153

Table 5. Experimental Results (b)

TCN	MAE	5.4061	2.4718	2.5472	3.6939	1.9537	2.2603	1.7028	3.2653	1.3406
	RMSE	6.4821	3.8127	4.9785	3.2498	2.9456	4.3792	7.0457	3.2159	5.0774
	MAPE	5.9724	4.1695	5.6409	6.8474	5.6776	4.6188	6.3234	4.3905	5.9877
	R^2	0.5636	0.5471	0.5038	0.5401	0.6949	0.4947	0.7569	0.4635	0.6162
Bert-TCN	MAE	0.6521	0.5837	0.3827	0.9872	0.4721	0.3728	0.7664	0.8265	0.8372
	RMSE	0.8372	0.9827	1.1123	0.9871	2.0082	1.1321	0.7652	0.8293	1.1264
	MAPE	0.7895	0.2689	0.9872	0.7845	0.8247	0.9824	0.8415	0.9725	1.0587
	R^2	0.9958	0.8945	0.9958	0.9857	0.9745	0.9674	0.9896	0.9713	0.9358

6 Conclusion

This investigation proposes a BERT model informed by the Hownet algorithm, designed to bring clarity to the ambiguity associated with sentiment analysis of Chinese financial news. We conducted a comparative performance assessment of various emotional classification models, including LSTM, RNN, CNN, and BERT, to evaluate the impact of disambiguation on the precision of classification. The incorporation of sentiment scores in the retroactive analysis of stock prices is another key focus of this work. An evident enhancement in the stock price prediction performance was observed in the Bert-TCN hybrid model, positioning it as a robust predictor in the context of the Chinese financial market. This innovative model, as introduced in this research, promises to bring significant improvements to sentiment interpretation and forecasting within China's financial landscape.

References

1. S. Hochreiter and J. Schmidhuber, 1997. "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997.

2. K. Cho, B. van Merrienboer, et al. “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation,” in IEEE Conference on Computer Vision and Pattern Recognition, vol. abs/1406.1078, pp. 1724-1733. 2014.
3. C. Zhao, Y. Ye, and M. Yao. “Stock Volatility Prediction Based on Sentiment Analysis of Financial Text,” in Computer Science, vol. 47, no. 5, pp. 79-83. 2020
4. Z. Dong and Q. Dong. “HowNet - A Hybrid Language and Knowledge Resource,” in Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering, pp. 820-824. IEEE. 2003.
5. A. I. McLeod and W. K. Li. “Diagnostic checking ARMA time series models using squared-residual autocorrelations,” Journal of Time Series Analysis, vol. 4, no. 4, pp. 269-273. 1983
6. L. Bauwens, S. Laurent, and J. V. K. Rombouts. “Multivariate GARCH models: A survey,” Journal of Applied Econometrics, vol. 21, no. 1, pp. 79-109. 2006.
7. K. Chen, Y. Zhou, and F. Dai. “A LSTM-based method for stock returns prediction: A case study of China stock market,” in 2015 IEEE International Conference on Big Data (Big Data), pp. 2823-2824. 2015.
8. T. Kim and H. Y. Kim. “Forecasting stock prices with a feature fusion LSTM-CNN model using different representations of the same data,” PloS one, vol. 14, no. 2, p. e0212320. 2019.
9. Q. Zhang, D. Yan, and J. Han. “Stock Price Prediction Based on Deep Learning and Decomposition Algorithms,” Computer Engineering and Applications. 2021.
10. Pei Dawei, Zhu Ming. Stock Price Prediction Based on Multiple-Factor and Multi-Variable Long Short Term Memory[J]. Computer Systems & Applications, 28(8): 30-38. 2019.
11. Bai S J, Kolter J Z, Koltun V.An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling [OL]. arXiv Preprint, arXiv: 1803.01271.
12. Ren R, Wu D D, Liu T X. Forecasting Stock Market Movement Direction Using Sentiment Analysis and Support Vector Machine [J]. IEEE Systems Journal, 213(1): 760-770. 2019.
13. Devlin J, Chang M W, Lee Kenton, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [OL]. arXiv Preprint, arXiv: 1810.04805.
14. Li, X., & Li, Q. (2015). Calculation of sentence semantic similarity based on syntactic structure. Mathematical Problems in Engineering. 2015.