Advances in Computer and Communication, 2024, 5(4), 243-259
https://www.hillpublisher.com/journals/acc/
ISSN Online: 2767-2875

The Research on End-to-end Stock Recommendation Algorithm Based on Time-frequency Consistency

Yunxiang Gan[1], Xiaoyang Chen[2,*]

[1]Moloco, Inc., Redwood City, CA 94063, USA.
[2]The Ohio State University, Columbus, OH 43210, USA.

How to cite this paper: Yunxiang Gan, Xiaoyang Chen. (2024) The Research on End-to-end Stock Recommendation Algorithm Based on Time-frequency Consistency. *Advances in Computer and Communication*, 5(4), 243-259.
DOI: 10.26855/acc.2024.10.008

Received: September 22, 2024
Accepted: October 16, 2024
Published: November 13, 2024

[*]**Corresponding author:** Xiaoyang Chen, The Ohio State University, Columbus, OH 43210, USA.

Abstract

In the financial market, the volatility and complexity of stock prices make accurately predicting stock trends a highly challenging task. Traditional stock prediction methods often rely on either time-domain or frequency-domain information alone, which fails to fully capture the multi-scale dynamic characteristics of stock prices, leading to insufficient prediction accuracy. To address the shortcomings of existing stock recommendation algorithms, this paper proposes an end-to-end stock recommendation algorithm based on time-frequency consistency. Firstly, we introduce a time-frequency consistency analysis method, which can simultaneously extract both time-domain and frequency-domain features of stock prices, thus providing a more comprehensive characterization of stock trend changes. Secondly, by integrating prompt learning strategies, the model is guided by pre-designed prompts to identify the lowest-risk buying points within specific time frames, optimizing the stock recommendation decision-making process. Finally, the end-to-end model training ensures seamless integration and automation throughout the entire prediction process, achieving a complete workflow from data input to stock recommendation. Experimental results demonstrate that this method outperforms traditional approaches in terms of prediction accuracy and risk control, offering more reliable decision support for investors.

Keywords

Time-Frequency Consistency; Stock Recommendation; Multi-Scale Dynamic Characteristics; Prompt Learning; Risk Control

1. Introduction

In today's highly volatile and complex financial markets, accurately predicting stock price trends is crucial for investors. Stock price fluctuations are influenced by various factors, including macroeconomic indicators, company performance, market sentiment, and international political situations. Due to the complexity of the interactions among these factors, predicting stock prices has become one of the most challenging tasks in the financial field [1]. In addition, research on corporate fraud in China's A-share listed companies reveals that macroeconomic factors and internal corporate governance have a direct impact on stock price fluctuations [2]. Traditional stock prediction methods, such as those based on time series analysis, moving averages, or regression models, provide tools for trend analysis to some extent but are often limited to single-domain time analysis. This one-dimensional approach fails to fully capture the multi-scale dynamic characteristics of stock prices and shows limitations, especially when dealing with nonlinear and complex market signals [3].

With advances in computing power and data processing technologies, machine learning algorithms have gradually become emerging tools for stock prediction. These methods include Support Vector Machines (SVM), Random Forests (RF), and Neural Networks (NN), which can make predictions by learning complex patterns from large amounts of historical data [4]. In particular, deep learning models, such as Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs), have shown great potential in handling nonlinear time series prediction tasks due to their strong feature extraction and pattern recognition capabilities [5]. However, most of these methods still rely on single-domain time analysis, making it difficult to comprehensively capture the multi-scale dynamic characteristics of stock prices. Frequency domain analysis methods have emerged as a response, revealing cyclical patterns hidden in price fluctuations by decomposing time series data into frequency components. For example, Fourier Transform can convert time series data into a sum of sine waves of different frequencies, helping to identify market behavior at different time scales [6]. Wavelet Transform further extends the application scope of frequency domain analysis by decomposing signals into sub-waves of different scales, providing joint time and frequency analysis [7]. Although these methods offer new tools for capturing hidden periodicity in stock prices, solely relying on frequency domain analysis still has limitations, especially when dealing with complex and volatile market environments, making it difficult to fully capture dynamic characteristics. Therefore, time-frequency analysis methods have gained popularity in recent years. These methods can analyze signals in both the time and frequency domains simultaneously, providing a more comprehensive tool for complex financial markets. This time-frequency analysis method is not only applicable to financial markets but also demonstrates its value in fields such as structural damage identification [8-10].

Despite the potential value of time-frequency analysis methods, their application in stock prediction still faces challenges. Firstly, effectively integrating time-domain and frequency-domain information to comprehensively characterize the dynamic changes in stock prices is a key focus of current research. Secondly, existing prediction algorithms often lack effective risk control in decision optimization. Merely improving prediction accuracy is not enough to cope with market uncertainty. Introducing intelligent decision support mechanisms into the prediction process to help investors make more robust investment decisions in complex and volatile markets is equally important. Comparable risk control methods have been effectively applied in cross-disciplinary deep learning frameworks, particularly when dealing with diverse data distributions, offering valuable insights for optimizing financial decision-making [11, 12]. Additionally, traditional stock prediction processes often involve multiple independent steps (e.g., feature extraction, model training, and decision generation). The separation of these steps not only reduces efficiency but also introduces errors at various stages [13]. To address these issues, this paper proposes an end-to-end stock recommendation algorithm based on time-frequency consistency, aiming to improve stock prediction accuracy and practicality by integrating time-domain and frequency-domain information and incorporating prompt learning strategies. Firstly, the theoretical foundation of time-frequency consistency analysis is explored in depth, and specific methods for stock price analysis are developed. Secondly, prompt learning strategies are designed to enable effective risk assessment and decision optimization by utilizing market features to guide the model. This prompt-based learning strategy is similar to current research methods in advertising recommendation systems, optimizing the model's decision-making process through predefined prompts [14]. Subsequently, an end-to-end stock recommendation model is constructed and trained, integrating time-frequency feature extraction and prompt learning processes, ensuring seamless integration throughout the prediction and decision-making process. At the same time, considering the impact of digital transformation on stock financing in China's A-share market, multidimensional data processing methods from the market have been used to optimize algorithm performance [15, 16]. Drawing on research related to brand reputation, this method further demonstrates its broad applicability in multidimensional data analysis, particularly its effectiveness in complex market environments [17, 18], as well as its potential in social sciences and market forecasting [19]. Finally, experimental design and empirical analysis are conducted to comprehensively evaluate the performance of the model, verifying its effectiveness and stability in different market environments. Similar to end-to-end model design in optical character classification and denoising, this integrated approach enhances prediction efficiency and streamlines the entire process, bringing greater accuracy and automation to stock forecasting [20, 21].

The structure of this paper is arranged as follows: The first part introduces the background, problems, objectives, significance, research methods, and content of the study. The second part reviews existing research in the field of stock prediction, particularly the progress in the application of time-frequency analysis and prompt learning strategies, and clarifies the research direction of this paper. The third part details the proposed end-to-end stock recommendation algorithm based on time-frequency consistency, including time-frequency consistency analysis, the design and application of prompt learning strategies, and the construction and implementation of the end-to-end model. The fourth chapter focuses on experimental design and result analysis, validating the effectiveness of the proposed algorithm through experiments, covering prediction accuracy tests on different datasets, model comparison experiments, and result analysis. The fifth part summarizes the main research findings and theoretical contributions of this paper, discusses the limitations of the study, and proposes future research directions.

The main contributions of this paper are as follows:

(1) This paper applies the time-frequency consistency analysis method to stock price prediction, successfully integrating time-domain and frequency-domain information to comprehensively capture the multi-scale dynamic characteristics of stock prices. Compared with traditional single-domain time or frequency analysis methods, this algorithm demonstrates higher prediction accuracy and robustness in handling complex market signals.

(2) To achieve more robust investment decisions in stock recommendations, this paper designs and applies a prompt learning strategy, guiding the model to identify low-risk buying and selling points through pre-designed market feature prompts. This strategy not only enhances the decision-making ability of the model but also shows significant advantages in risk control. This strategy is similar to emotion-driven marketing strategies that boost sales growth, utilizing prompt-based learning to achieve more effective investment decisions [22].

(3) This paper develops an end-to-end stock recommendation model that integrates time-frequency consistency analysis with prompt learning strategies, simplifying multiple independent steps in traditional prediction processes. This model achieves full-process automation from data input to stock recommendation, not only improving prediction efficiency but also enhancing the model's integration and practicality in real-world applications.

2. Related Work

As the global economy continues to develop and financial markets become increasingly complex, accurately predicting stock prices has become a growing challenge. This challenge mainly stems from the non-stationarity, high volatility, frequent fluctuations, and inherent randomness of stock market data. These characteristics often make traditional statistical models and fundamental analysis methods inadequate when dealing with complex time series data [23]. An increasing number of scholars and practitioners are dedicated to developing more precise and efficient predictive models. Traditional stock prediction methods primarily include time series analysis, technical analysis, and fundamental analysis. Building on these methods, researchers have proposed improved time series analysis approaches. For example, Shakir Khan et al. [24] proposed an ARIMA model-based method to accurately predict stock time series. By analyzing five years of historical data for Netflix stock, they compared an automated ARIMA model with a custom ARIMA(p, D, q) model and found that ARIMA(1,1,33) performed best in terms of accuracy, demonstrating the effectiveness of the ARIMA model in stock prediction. Lu Wang et al. [25] proposed a GARCH-MIDAS model that combines asymmetry and extreme volatility effects to model and predict stock price volatility more accurately. Their research indicates that the asymmetric effect has a significantly greater impact on volatility in both the long and short term compared to extreme volatility effects. Through a series of robustness tests, their study also confirmed the model's superior performance in predicting short-term volatility. These methods assume that market prices follow certain historical patterns that can be captured by statistical models. However, as the market environment continues to evolve, especially in the face of nonlinear and highly volatile market conditions, the predictive effectiveness of such methods is often limited.

In recent years, with the rise of deep learning technologies, models like Long Short-Term Memory (LSTM) networks have shown great potential in handling nonlinear time series prediction tasks. However, they mainly rely on time-domain data and struggle to capture more complex market dynamics. Hum Nath Bhandari et al. [26] proposed an LSTM-based method for predicting the next day's closing price of the S&P 500 index. They developed single-layer and multi-layer LSTM models using nine predictive factors, including market data, macroeconomic data, and technical indicators. The study results showed that the single-layer LSTM model outperformed the multi-layer LSTM model in prediction accuracy and fit. Burak Gülmez et al. [27] proposed an optimized deep Long Short-Term Memory network (LSTM-ARO) using the Artificial Rabbits Optimization (ARO) algorithm for stock price prediction. This approach optimizes the LSTM model's hyperparameters to improve prediction accuracy and was tested using Dow Jones Industrial Average (DJIA) stock data. The results showed that LSTM-ARO outperformed other models across various evaluation metrics. In addition to these time-domain models, technical analysis also plays a vital role in financial forecasting. Indicators such as Moving Averages (MA) and the Relative Strength Index (RSI) are commonly used to interpret historical price and volume data. While these indicators are intuitive and widely adopted, they mainly focus on short-term price patterns and often neglect broader market fundamentals, especially in the context of long-term prediction [28]. Moreover, technical indicators typically perform poorly during unexpected market events or extreme fluctuations, limiting their robustness in volatile environments. To address these limitations, researchers have explored frequency-domain methods, which provide a different perspective by transforming time-series data into frequency components, revealing cyclical patterns. Donghwan Song et al. [29] introduced the Padding-Fourier Transform Denoising (P-FTD) method to improve financial time-series predictions. By eliminating noise in the frequency domain, P-FTD addresses the issue of data divergence at the ends of time series and enhances the performance of deep learning models by reducing time lag in predictions. Similarly, Satya Verma et al. [30] proposed a method that integrates Discrete Wavelet Transform (DWT) and Chicken Swarm Optimization (CSO), named DWT-CSO, to enhance stock market prediction. By decomposing data using DWT and applying CSO to select an optimal feature subset, their approach mitigates the challenges of noisy data and excessive features. While frequency-domain techniques like DWT-CSO show potential, they also face challenges in fully integrating time-domain data to capture market dynamics comprehensively.

Time-frequency analysis methods, such as Short-Time Fourier Transform (STFT) and Wavelet Packet Decomposition (WPD), attempt to analyze signals in both the time and frequency domains simultaneously, providing a more comprehensive understanding of market dynamics. For example, Yaqing Luo et al. [31] proposed a wavelet neural network model that incorporates time-frequency analysis. This model uses Gaussian wavelets as the activation function and applies wavelet decomposition to stock price data, enhancing the model's sensitivity to market fluctuations. The results demonstrated a significant reduction in mean squared error when applied to London stock market data, highlighting the potential of time-frequency methods for financial prediction tasks. While these approaches excel at capturing both time and frequency components of market behavior, they still face challenges in volatile and complex market environments. A critical issue in current research is improving models' ability to accurately identify favorable investment opportunities and potential risks amidst market fluctuations. Recently, emerging machine learning optimization methods, such as prompt learning strategies, have shown potential in the financial domain. Defu Cao et al. [32] introduced a framework called TEMPO, which is based on generative pre-trained Transformers for time series prediction. TEMPO leverages two key inductive biases: the decomposition of complex interactions such as trend, seasonality, and residual components, and the ability to adapt to different types of time series distributions using specially designed prompts. This framework enhances prediction accuracy by guiding the model to focus on specific features of the time series data. In another approach, Tian Guo et al. [33] explored stock return prediction by fine-tuning large language models (LLMs) in combination with financial news streams. By integrating textual representation with a prediction module, their method demonstrated the effectiveness of both encoder-only and decoder-only LLMs in improving the performance of long and long-short portfolios. This suggests that LLMs can effectively integrate non-quantitative data, such as financial news, to enhance stock market predictions. These prompt learning strategies are valuable because they optimize the model's decision-making process by guiding it to focus on key market features through pre-designed prompts. In financial prediction, end-to-end models that integrate feature extraction, model training, and decision-making into a unified framework reduce errors associated with traditional multi-step processes. Despite these advances, current stock prediction methods still face limitations in capturing multi-scale market features, achieving effective risk control, and providing reliable decision support. Building on this foundation, this paper proposes an innovative end-to-end stock recommendation algorithm based on time-frequency consistency. This method aims to improve prediction accuracy and offer more practical decision-making support for investors.

3. Method

Figure 1 illustrates the overall architecture of the proposed stock recommendation algorithm, which is trained in an end-to-end manner to ensure time-frequency consistency. First, the input stock price data undergoes feature transformation through an embedding layer, generating processable feature representations. Then, the input data is divided into two parts: Trend and Residual, which are processed by the Time Encoder and Frequency Encoder, respectively, ensuring the extraction of both time-domain and frequency-domain features. The encoded information, processed through patching, is fused with pre-designed prompt embeddings, where prompts guide the model to identify low-risk buying points. Next, the Transformer blocks, which include multi-head attention and normalization, further process the features and combine them with positional embeddings. Finally, the processed features are fed into a neural network for stock trend prediction.

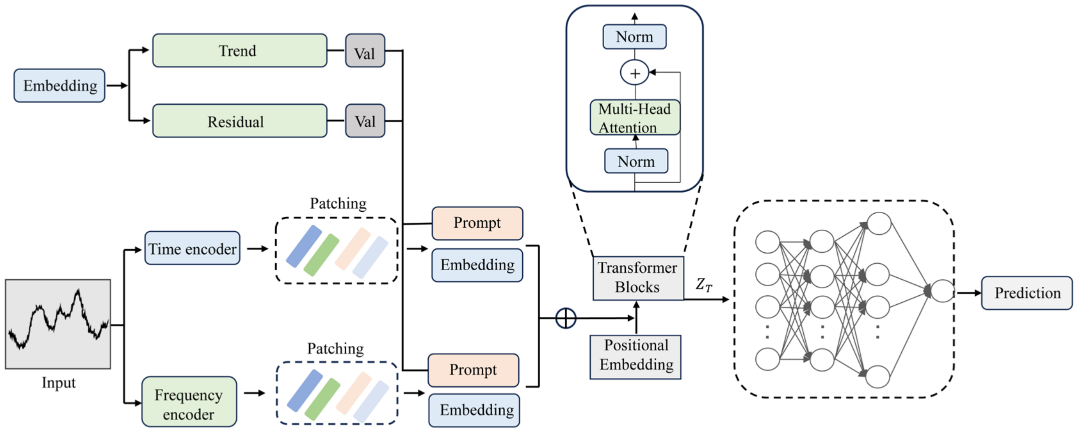

Figure 1. Overall algorithm architecture.

3.1 Time-Frequency Consistency Model

The time-frequency consistency model is an innovative method proposed in recent years for time series analysis. It aims to represent time series data in both the time domain and frequency domain simultaneously, ensuring consistency in a unified time-frequency space. This approach is particularly suitable for handling time series data with complex dynamic characteristics. Time series data are prevalent across various fields, such as financial markets, medical diagnostics, and traffic analysis. In financial markets, stock price fluctuations not only reflect trends over time (time domain features) but also contain various periodic and non-periodic components (frequency domain features). Traditional time series analysis methods typically focus on either time domain or frequency domain analysis, making it challenging to comprehensively capture these complex features. The time-frequency consistency model was proposed to overcome this limitation. The core idea is to simultaneously learn feature representations in both the time and frequency domains and enforce consistency between these representations in a latent time-frequency space. This enables the model to better understand and predict the dynamic changes in time series data. The algorithm architecture diagram is shown in Figure 2.

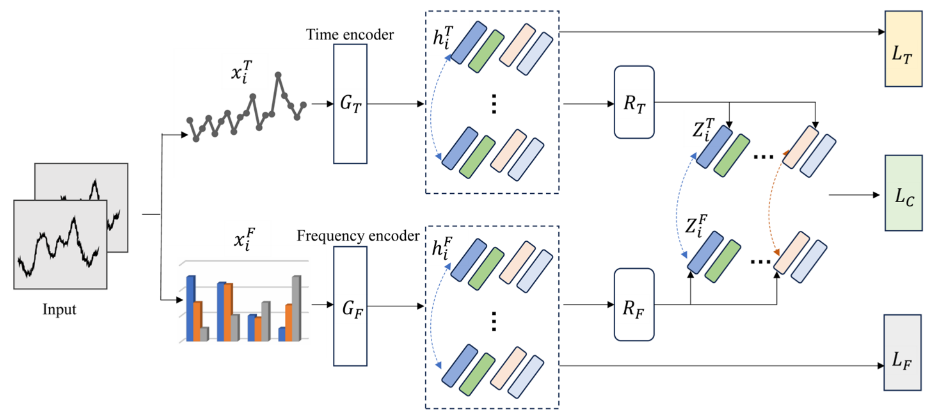

Figure 2. Time-frequency consistency model architecture diagram.

The model comprises a time encoder G_T and a frequency encoder G_F. The time encoder receives the time series input $x(t)$ and maps it to a latent representation in the time domain z_T:

$$z_T = G_T(x(t)) \tag{1}$$

The frequency encoder G_F receives the frequency representation of the time series $x(f)$ and maps it to a latent representation in the frequency domain z_F:

$$z_F = G_F(x(f)) \tag{2}$$

Here, $x(f)$ is the frequency domain representation of the signal obtained through Fourier transform or other spectral analysis methods.

To compare the representations in the time and frequency domains within the same space, the model introduces two projectors: the time domain projector R_T and the frequency domain projector R_F. These projectors map the time and frequency representations into a unified time-frequency consistency space:

$$z_T^{(p)} = R_T(z_T), \quad z_F^{(p)} = R_F(z_F) \tag{3}$$

where $z_T^{(p)}$ and $z_F^{(p)}$ represent the projected time domain and frequency domain representations, respectively.

The model is designed with a loss function that ensures the representations of the same time series in the time and frequency domains are as close as possible in the projected time-frequency space. To achieve this, a time-frequency consistency loss function L_C is introduced:

$$L_C = \sum_{\text{Spair}} \left(d(z_T^{(p)}, z_F^{(p)}) + d(z_T^{(p)}, \tilde{z}_F^{(p)}) + \delta \right) \tag{4}$$

where $d(\cdot,\cdot)$ denotes a distance measure in the projection space, $\tilde{z}_F^{(p)}$ is the representation after frequency domain perturbation, and δ is a constant used to maintain negative sample separation.

This loss function encourages the model to pull the time domain and frequency domain representations of the same time series closer together in the time-frequency space while pushing apart the representations of different time series or perturbed representations. This maintains consistency in the latent space. The model is trained using a contrastive learning framework by constructing positive and negative sample pairs. Positive sample pairs consist of time and frequency domain representations of the same time series, while negative sample pairs are composed of representations from different time series or the original and perturbed representations.

The total loss function of the model consists of three parts. The time domain contrastive loss L_T is used to optimize the time encoder G_T to generate representations invariant to time perturbations:

$$L_T = \sum_i d(z_T, \tilde{z}_T) \tag{5}$$

The frequency domain contrastive loss L_F is used to optimize the frequency encoder G_F to generate representations invariant to spectral perturbations:

$$L_F = \sum_i d(z_F, \tilde{z}_F) \tag{6}$$

The total loss function is:

$$L = \lambda(L_T + L_F) + (1-\lambda)L_C \tag{7}$$

where λ is a hyperparameter that balances the contrastive loss and the consistency loss. By minimizing this total loss function, the model can learn both time domain and frequency domain feature representations while maintaining consistency between them in the time-frequency space, thereby enhancing the model's ability to capture the complex dynamic characteristics of time series data [34].

3.2 Prompt Learning Model

Prompt learning aims to guide the model in identifying the lowest-risk entry points within a specific time frame by using pre-designed prompts. The design of these prompts relies on the analysis of historical stock price data, combined with prior market knowledge and specific investment strategies. These prompts can be specific patterns in the time series, threshold values of indicators, or time windows for certain key events. Subsequently, based on these time-frequency consistency features, prompt learning is used to optimize investment decisions. The architecture diagram is shown in Figure 3.

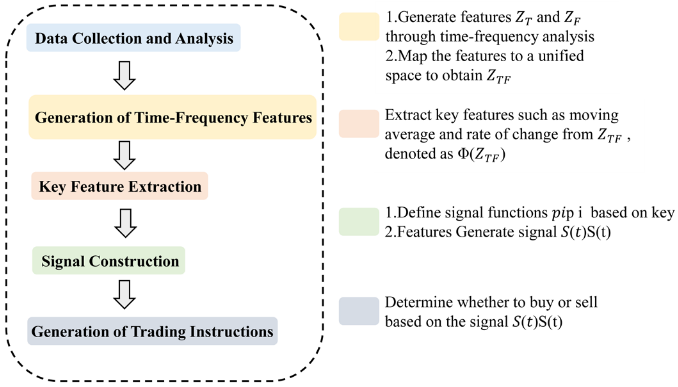

Figure 3. Prompt learning model architecture diagram.

Assume that the time series data $X = \{x_1, x_2, \dots, x_T\}$ and the time-frequency consistency features generated through time-frequency analysis are represented as follows:

$$\begin{aligned} Z_T &= \{z_T(t_1), z_T(t_2), \dots, z_T(t_T)\} \\ Z_F &= \{z_F(f_1), z_F(f_2), \dots, z_F(f_F)\} \end{aligned} \tag{8}$$

where $z_T(t_i)$ represents the time-domain feature, and $z_F(f_j)$ represents the frequency-domain feature. To enable time and frequency features to be compared and fused in the same space, the model uses two projectors to map the time and frequency features into a unified time-frequency consistency space, forming the time-frequency consistency features Z_{TF}. The time-frequency consistency features are represented as:

$$Z_{TF} = \{z_{TF}(1), z_{TF}(2), \dots, z_{TF}(T)\} \tag{9}$$

where $z_{TF}(i)$ is the time-frequency consistency feature at the i-th time step, which combines the time-domain feature $z_T(t_i)$ and the frequency-domain feature $z_F(f_j)$, specifically expressed as:

$$z_{TF}(i) = \varepsilon_1 \cdot z_T(t_i) + \varepsilon_2 \cdot z_F(f_j) \tag{10}$$

Here, ε_1 and ε_2 are learnable weight parameters used to adjust the relative importance of the time-domain and frequency-domain features.

After obtaining the time-frequency consistency feature Z_{TF}, a set of prompt signals is constructed using these time-frequency features to guide the model in making buy or sell decisions. To more effectively utilize these features, further processing and feature extraction are performed. By calculating the moving averages, rate of change, frequency domain energy, etc., of these features, different market signals are captured:

$$\Phi(Z_{TF}) = \{\phi_1(Z_{TF}), \phi_2(Z_{TF}), \dots, \phi_k(Z_{TF})\} \tag{11}$$

where $\phi_i(Z_{TF})$ is the i-th feature extracted from the time-frequency consistency features, and k is the number of features.

Based on the extracted features $\Phi(Z_{TF})$, prompt functions p_i are defined, which are used to generate buy or sell prompt signals. A prompt function based on time-frequency energy aggregation can be defined as follows:

$$p_i(\Phi(Z_{TF}), t) = \sum_{f \in F_{\text{selected}}} \phi_i(Z_{TF})(t, f) \tag{12}$$

where F_{selected} represents the selected frequency range, and $\phi_i(Z_{TF})(t, f)$ represents the time-frequency feature value at time t and frequency f. This prompt function indicates the degree of energy aggregation within a specific frequency range, corresponding to a certain market signal such as a trend reversal or price breakout.

To provide more comprehensive decision-making guidance, the outputs of multiple prompt functions are integrated to generate a final prompt signal $S(t)$:

$$S(t)=\sum_{i=1}^{N} w_i \cdot p_i(Z_{TF},t) \tag{13}$$

where w_i is the weight of the prompt function p_i, indicating the importance of each prompt in the combined signal. These weights are automatically adjusted through the model training process to maximize prediction performance.

After obtaining the combined prompt signal $S(t)$, decision rules can be generated based on this signal value to issue buy or sell instructions. When $S(t)$ exceeds a certain threshold θ, it indicates that the current market state is suitable for buying: if $S(t) > \theta$, then buy at time t, similarly, if $S(t)$ falls below another threshold θ', it may suggest selling [35].

3.3 End-to-End Learning

In the stock recommendation algorithm proposed in this paper, the end-to-end learning method is the core of the entire model, enabling the full automation of the process from raw data input to final decision output. Within the end-to-end learning framework, all steps, from time-frequency consistency feature extraction and prompt signal generation to decision optimization, are integrated into a unified model, where joint training directly optimizes the final investment decisions.

The end-to-end learning model consists of several sub-modules that work collaboratively to achieve the overall goal of stock recommendation. The input representation module receives the raw time-series data $X=\{x_1,x_2,\dots,x_T\}$ and performs initial feature extraction. The time-frequency consistency module extracts time-frequency features Z_{TF} from the input representation, capturing the dynamic changes in stock prices across both the time and frequency domains. The prompt learning module constructs prompt signals $S(t)$ based on the time-frequency features, which guide the buy or sell decisions. The decision module generates the final investment decisions based on the prompt signals. The architecture of the entire model can be expressed as the following function composition:

$$D(X)=f_{\text{decision}}\left(f_{\text{prompt}}\left(f_{\text{TF}}(X)\right)\right) \tag{14}$$

where f_{TF} represents the time-frequency consistency module, f_{prompt} represents the prompt learning module, and f_{decision} represents the final decision module. The output $D(X)$ is the final decision result.

The joint loss function in end-to-end learning simultaneously optimizes the parameters of all sub-modules, thereby directly enhancing the quality of the final decisions. The total loss function L_{total} is composed of the following three parts:

Time-frequency consistency loss L_{TF} is used to optimize the representation of time-frequency features, ensuring consistency across both time and frequency domains.

$$L_{\text{TF}}=\sum_{t=1}^{T}\left\| z_T^{(p)}(t)-z_F^{(p)}(t)\right\|^2 \tag{15}$$

Here, $z_T^{(p)}(t)$ is the time-domain projected feature, $z_F^{(p)}(t)$ is the frequency-domain projected feature.

Prompt learning loss L_{prompt} optimizes the generation of prompt signals to accurately reflect potential market trends and risks.

$$L_{\text{prompt}}=\sum_{t=1}^{T}\left\| S(t)-\hat{S}(t)\right\|^2 \tag{16}$$

where $S(t)$ is the prompt signal generated by the model, and $\hat{S}(t)$ is the target signal based on historical data.

Decision loss L_{decision} directly optimizes the final investment decisions, minimizing risk while maximizing returns.

$$L_{\text{decision}}=-\sum_{t\in B} r(t)+\tau\sum_{t\in B}\sigma(t) \tag{17}$$

where $r(t)$ represents the expected return at time t, $\sigma(t)$ represents the corresponding risk measure, and τ is a balancing coefficient.

The joint loss function is defined as:

$$L_{\text{total}} = \alpha L_{\text{TF}} + \beta L_{\text{prompt}} + \gamma L_{\text{decision}} \tag{18}$$

where α, β, and γ are hyperparameters that adjust the weights of each loss component and are automatically tuned during model training.

By minimizing the joint loss function L_{total}, the parameters of all modules are optimized simultaneously. The model is trained using gradient descent, gradually updating the parameters to reduce the overall loss:

$$\theta^* = \arg\min_\theta L_{\text{total}}(\theta) \tag{19}$$

where θ represents all the model parameters, including those of the time-frequency consistency module, prompt learning module, and decision module. During training, the model continuously learns from historical data, gradually improving its ability to predict and make decisions for future markets. Through end-to-end joint training, the model can automatically capture complex patterns in the input data and directly use these patterns to guide decision-making.

4. Experiment

4.1 Experimental Environment

The experimental environment in this study includes both hardware and software configurations, as well as data sources. On the hardware side, the experiments were conducted on a computer equipped with an Intel Core i7 processor, 64GB of RAM, and an NVIDIA GTX 3090 graphics card. On the software side, Python was primarily used as the programming language, along with the TensorFlow deep learning framework for model training and inference. Additionally, data processing libraries such as Pandas and NumPy were utilized to efficiently manage and process time-series data.

4.2 Experimental Data

- S&P 500 Index Constituents Dataset

The S&P 500 Index Constituents dataset [36] contains the stock price data of the 500 largest publicly traded companies listed on U.S. exchanges, which are part of the S&P 500 index. The S&P 500 is widely regarded as one of the best benchmarks for measuring the overall performance of the U.S. stock market, representing a broad cross-section of industries and sectors in the economy.

- Shanghai A-Share Dataset

The Shanghai Stock Exchange A-Share dataset [37] contains stock data of companies listed on the Shanghai Stock Exchange that issue A-shares. A-shares are shares of mainland China-based companies, denominated in Renminbi (RMB), and available for trading by domestic investors and qualified foreign investors. The dataset provides insight into the performance of China's domestic stock market.

- NASDAQ 100 Index Constituents Dataset

The NASDAQ 100 Index Constituents dataset [38] contains stock data from the NASDAQ 100, which is composed of the largest non-financial companies listed on the NASDAQ stock exchange. It includes key sectors such as technology, consumer services, and healthcare, making it a highly popular index for analyzing the performance of leading tech and growth companies.

- FTSE 100 Index Constituents Dataset

The FTSE 100 Index Constituents dataset [39] contains stock data of the 100 largest companies listed on the London Stock Exchange by market capitalization. It serves as a major indicator of the performance of the U.K. stock market and includes multinational corporations, providing insight into both domestic and international business environments.

4.3 Evaluation Metrics

- Mean Absolute Error (MAE)

MAE is used to measure the average absolute error between the model's predicted results and the actual stock prices. The lower the MAE value, the more accurate the model's prediction of stock prices. Since MAE only considers the absolute value of the errors, it avoids the issue of positive and negative errors canceling each other out. Therefore, MAE directly reflects the magnitude of the model's prediction bias. The formula for MAE is as follows:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{20}$$

where y_i is the actual stock price, $\hat{y}_i$ is the model's predicted stock price, and n is the number of samples.

- Mean Squared Error (MSE)

MSE is used to measure the overall error of the model by taking the average of the squared prediction errors. It not only measures the size of the prediction error but also amplifies the effect of large errors due to the squaring operation, which makes it particularly sensitive to extreme market fluctuations. The lower the MSE value, the better the overall performance of the model's predictions. The formula for MSE is as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{21}$$

- Coefficient of Determination (R^2)

R^2 measures the model's ability to explain the variability in stock prices, with a value between 0 and 1. The closer it is to 1, the better the model's fit. In stock recommendation algorithms, R^2 can help us understand the correlation between the predicted stock prices and the actual prices. A high R^2 value indicates that the model captures stock price trends well, whereas a low R^2 value may suggest that the model has not fully utilized the data to make accurate predictions. The formula is as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \overline{y}\right)^2} \tag{22}$$

where $\bar{y}$ is the mean of all actual stock prices.

- Normalized Discounted Cumulative Gain (NDCG)

NDCG is commonly used as an evaluation metric in recommendation systems, suitable for assessing the performance of ranking tasks. NDCG effectively measures whether the positions of highly relevant items in the recommendation list are reasonable and whether the model can correctly identify and prioritize important stocks or other financial products. A high NDCG value indicates that the model can accurately rank the most relevant items at the top, thus improving user satisfaction and the success rate of investment decisions.

$$\mathrm{NDCG} = \frac{\mathrm{DCG}}{\mathrm{IDCG}} \tag{23}$$

where DCG is the Discounted Cumulative Gain, which considers the importance of the position by calculating the cumulative gain with a discount factor for the ranking position.

$$\mathrm{DCG} = \sum_{i=1}^{n}\frac{rel_i}{\log_2(i+1)} \tag{24}$$

where i is the position of the recommended item in the list, and rel_i is the relevance score of the i-th recommended item. Generally, the top few items in the list are more important, and by discounting for the position, the quality of the recommendation system can be more accurately reflected. IDCG represents the Ideal Discounted Cumulative Gain, which refers to the maximum DCG value that can be achieved in the ideal case (where the most relevant items are ranked at the top).

$$\mathrm{IDCG} = \sum_{i=1}^{|REL|}\frac{rel_i}{\log_2(i+1)} \tag{25}$$

where $|REL|$ represents the total number of recommended items in the ideal ranking.

4.4 Experimental Comparison and Analysis

To validate the effectiveness of the proposed end-to-end stock recommendation algorithm based on time-frequency consistency, we conducted extensive experiments and comparative analysis. The experiments utilized several publicly available stock market datasets, including the S&P 500 Index Constituents dataset, the Shanghai A-Share dataset, and others. We compared the proposed method with various other methods, focusing on evaluating its performance in terms of prediction accuracy and risk control.

Table 1. Comparison of relevant indicators of the proposed method with other methods on four datasets

Model	S&P 500 Index Constituents Dataset				Shanghai A-Share Dataset			
	MAE	MSE	R^2	NDCG	MAE	MSE	R^2	NDCG
Yang et al. [40]	0.264	0.136	0.862	0.374	0.253	0.124	0.883	0.382
Liu et al. [41]	0.194	0.112	0.894	0.384	0.186	0.101	0.912	0.391
Lu et al. [42]	0.284	0.153	0.849	0.362	0.261	0.135	0.857	0.376
Chaudhari et al. [43]	0.337	0.197	0.837	0.354	0.303	0.142	0.869	0.379
Wijerathne et al. [44]	0.163	0.073	0.912	0.391	0.158	0.064	0.924	0.412
Mahmoodi et al. [45]	0.192	0.092	0.896	0.388	0.176	0.087	0.922	0.396
Ours	0.078	0.036	0.944	0.436	0.064	0.027	0.952	0.447
Model	NASDAQ 100 Index Constituents Dataset				FTSE 100 Index Constituents Dataset			
	MAE	MSE	R^2	NDCG	MAE	MSE	R^2	NDCG
Yang et al. [40]	0.324	0.195	0.842	0.362	0.335	0.194	0.842	0.358
Liu et al. [41]	0.242	0.117	0.882	0.372	0.257	0.129	0.871	0.374
Lu et al. [42]	0.276	0.142	0.877	0.368	0.286	0.162	0.841	0.359
Chaudhari et al. [43]	0.312	0.163	0.853	0.351	0.322	0.185	0.841	0.339
Wijerathne et al. [44]	0.154	0.071	0.905	0.396	0.169	0.079	0.907	0.388
Mahmoodi et al. [45]	0.176	0.086	0.893	0.384	0.172	0.081	0.899	0.386
Ours	0.094	0.063	0.924	0.415	0.069	0.035	0.942	0.436

From the data in Table 1, it is evident that our proposed algorithm outperforms other methods on the S&P 500 Index Constituents Dataset and the Shanghai A-Share Dataset, particularly in terms of both predictive accuracy and error minimization. Our method demonstrates notable advantages, achieving lower MAE and MSE values compared to other approaches, while also showing superior performance in R^2 and NDCG metrics. Specifically, in terms of MAE and MSE, our method achieved the lowest values of 0.078 and 0.036 on the S&P 500 dataset, and 0.064 and 0.027 on the Shanghai A-Share dataset, indicating that our method significantly outperforms others in error control. Additionally, for the R^2 and NDCG metrics, our method also stands out, reaching 0.944 and 0.436 on the S&P 500 dataset, and 0.952 and 0.447 on the Shanghai A-Share dataset, far exceeding other comparative methods. In contrast, while the method by Wijerathne et al. also has relatively high R^2 and NDCG values, it still falls short of our method in terms of error metrics. Our method still outperforms other methods on various indicators in the NASDAQ 100 and FTSE 100 Index Constituents Datasets. Figure 4 provides a visual comparison of these results.

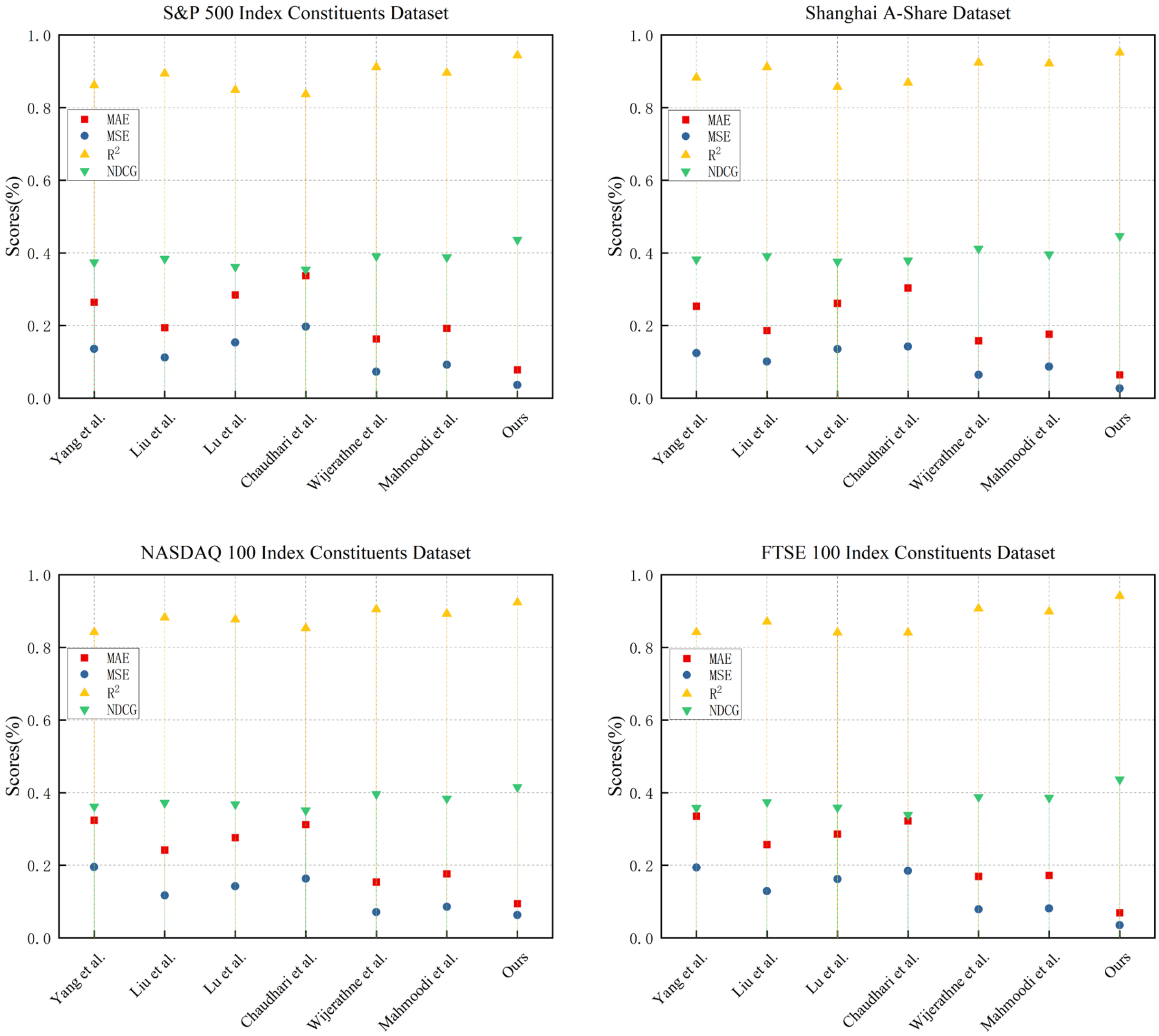

Figure 4. Visual comparison of relevant indicators on four datasets.

To demonstrate the efficiency advantage of our proposed method, we compare its computational efficiency with other methods. The following table summarizes key metrics such as parameter size, inference time, and training time across four datasets, highlighting the efficiency improvements brought by our approach.

From the data in Table 2, our method excels in terms of parameter size, inference time, and training time. Firstly, for the number of parameters, our method has the smallest parameter scale across all four datasets. For example, on the S&P 500 and NASDAQ 100 datasets, our model parameters are 336.54M and 342.64M, significantly reduced compared to other models, indicating that our model is more lightweight and efficient. On the Shanghai A-Share dataset, our model also maintains a smaller parameter size of 338.69M, further highlighting its efficiency. Secondly, for inference time, our model showed the fastest inference speed across all datasets. On the S&P 500 and FTSE 100 datasets, the inference times are 306.46ms and 275.74ms, lower than other methods, indicating that our model responds faster in real-time inference. Finally, for training time, our method also demonstrated the shortest training time across all datasets. On the NASDAQ 100 dataset, our training time was 177.84 seconds, and on the FTSE 100 dataset, it was 220.34 seconds, more efficient than other models. Notably, on the Shanghai A-Share dataset, our method achieved the fastest training time of 176.52 seconds, further proving its efficiency. This proves that while maintaining prediction accuracy, our model can significantly reduce computation costs and training time, improving overall efficiency. Figure 5 provides a visual comparison of these results.

Table 2. Comparison of training indicators on four datasets.

Model	S&P 500 Index Constituents Dataset Parameters (M)	Inference Time (ms)	Training Time (s)	Shanghai A-Share Dataset Parameters (M)	Inference Time (ms)	Training Time (s)
Yang et al. [40]	391.04	396.70	184.80	351.70	356.89	285.14
Liu et al. [41]	377.57	363.19	183.82	374.00	365.12	265.35
Lu et al. [42]	358.40	392.19	202.62	372.88	331.20	242.99
Chaudhari et al. [43]	384.77	383.27	207.68	368.49	355.77	263.88
Wijerathne et al. [44]	390.79	373.21	258.84	359.01	367.47	267.78
Mahmoodi et al. [45]	366.53	342.79	214.83	353.25	389.96	208.06
Ours	336.54	306.46	171.24	338.69	316.73	176.52

Model	NASDAQ 100 Index Constituents Dataset Parameters (M)	Inference Time (ms)	Training Time (s)	FTSE 100 Index Constituents Dataset Parameters (M)	Inference Time (ms)	Training Time (s)
Yang et al. [40]	373.22	393.86	253.88	381.06	314.66	235.83
Liu et al. [41]	382.11	372.26	239.43	372.00	301.52	287.33
Lu et al. [42]	355.96	382.87	235.64	359.78	295.74	235.82
Chaudhari et al. [43]	374.27	379.35	284.10	368.58	376.55	291.35
Wijerathne et al. [44]	361.01	386.80	247.54	357.61	340.72	257.43
Mahmoodi et al. [45]	356.60	359.99	284.54	377.29	325.23	229.29
Ours	342.64	322.59	177.84	342.57	275.74	220.34

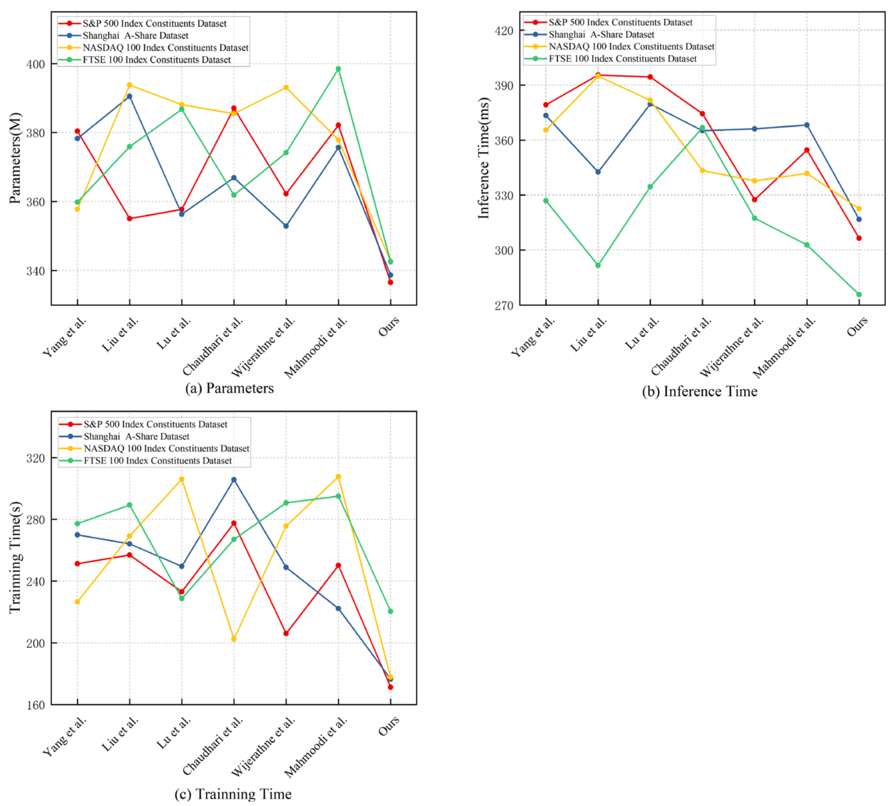

Figure 5. Visual comparison of training indicators.

To further evaluate the effectiveness of the proposed stock recommendation algorithm, we conducted ablation experiments by gradually introducing the Time-Frequency Consistency (TFC) and Prompt Learning mechanisms. Detailed comparisons were made across four major datasets. Table 3 summarizes the experimental results by comparing the performance of different models across multiple metrics.

Table 3. Ablation experiments on four datasets.

Model	S&P 500 Index Constituents Dataset				Shanghai A-Share Dataset			
	MAE	MSE	R^2	NDCG	MAE	MSE	R^2	NDCG
baseline	0.286	0.163	0.784	0.364	0.273	0.151	0.789	0.381
+TFC	0.185	0.069	0.864	0.396	0.177	0.052	0.873	0.411
+Prompt	0.124	0.043	0.916	0.419	0.112	0.035	0.927	0.426
+TFC-Prompt	0.078	0.036	0.944	0.436	0.064	0.027	0.952	0.447
Model	NASDAQ 100 Index Constituents Dataset				FTSE 100 Index Constituents Dataset			
	MAE	MSE	R^2	NDCG	MAE	MSE	R^2	NDCG
baseline	0.316	0.196	0.754	0.352	0.279	0.159	0.776	0.373
+TFC	0.224	0.069	0.826	0.381	0.182	0.062	0.869	0.399
+Prompt	0.151	0.064	0.896	0.401	0.121	0.041	0.921	0.420
+TFC-Prompt	0.094	0.063	0.924	0.415	0.069	0.035	0.942	0.436

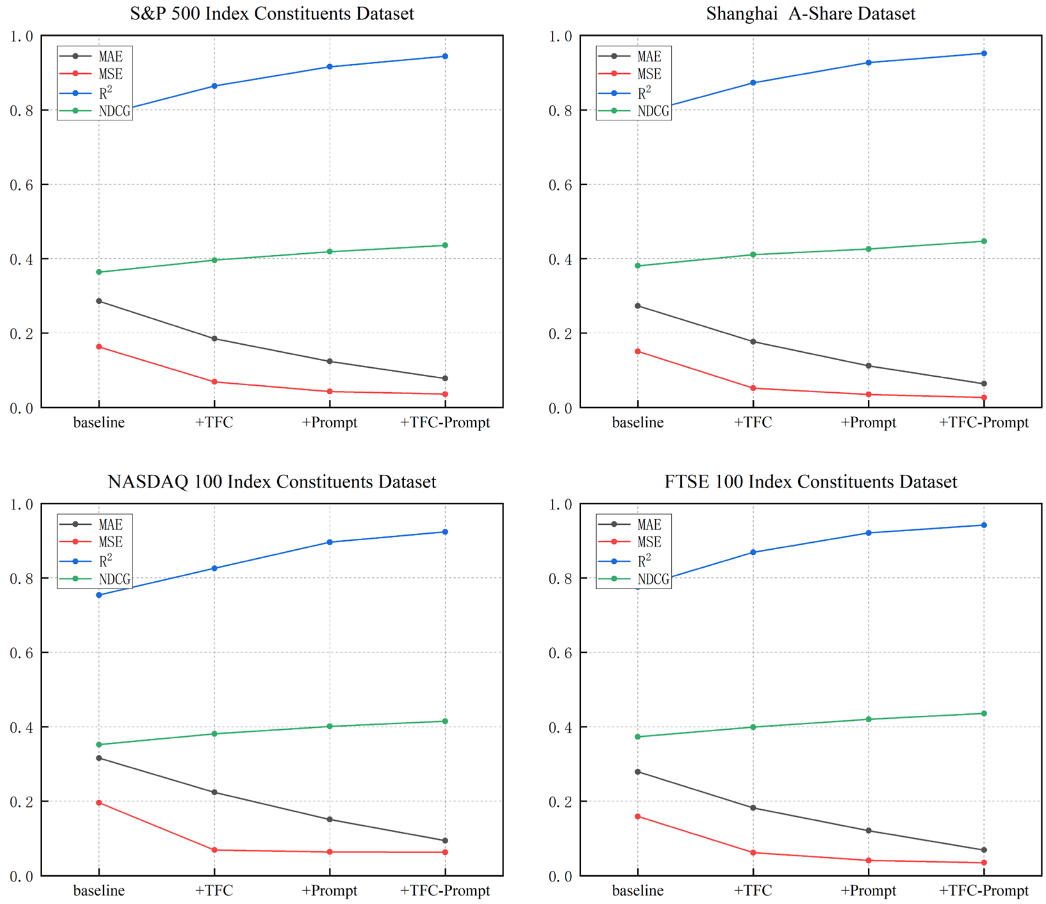

Figure 6. Visual comparison of ablation experiments on four datasets.

The data in Table 3 shows that the stock recommendation algorithm, which combines Time-Frequency Consistency (TFC) and Prompt learning, significantly outperforms the baseline model across multiple datasets. On the S&P 500 Index Constituents Dataset, the baseline model's MAE is 0.286, MSE is 0.163, R^2 is 0.784, and NDCG is 0.364. After introducing Time-Frequency Consistency (TFC), the MAE on the S&P 500 dataset dropped to 0.185, MSE dropped to 0.069, R^2 increased to 0.864, and NDCG rose to 0.396. Further introducing Prompt learning improved performance even more, with the S&P 500 dataset's MAE and MSE reaching 0.124 and 0.043, R^2 rising to 0.916, and NDCG to 0.419. Ultimately, when using the TFC-Prompt strategy, the S&P 500 dataset achieved the lowest MAE of 0.078, MSE of 0.036, R^2 of 0.944, and NDCG of 0.436. The algorithm continues to perform well on other datasets. As the model gradually introduces the TFC and Prompt mechanisms, all indicators show significant improvement. This indicates that the synergy between Time-Frequency Consistency and Prompt learning can provide more accurate and reliable investment strategies for stock recommendations. Figure 6 visually depicts these trends.

5. Conclusion

This paper proposes an end-to-end stock recommendation algorithm based on time-frequency consistency analysis and prompt learning strategies. It addresses the limitations of traditional stock prediction methods that focus only on time-domain or frequency-domain information, enabling the model to fully capture the multi-scale dynamic characteristics of stock prices. By introducing time-frequency consistency analysis, the model simultaneously considers both time-domain and frequency-domain features, resulting in more accurate stock price predictions. The prompt learning strategy further optimizes the decision-making process by identifying low-risk entry points and enhancing risk control in stock recommendations. Furthermore, the end-to-end model simplifies the entire prediction process, from data input to final recommendation, significantly improving prediction efficiency and reducing model complexity. Experimental results show that the algorithm significantly outperforms existing models in terms of prediction accuracy, risk control, and computational efficiency across multiple datasets. On the S&P 500 Index Constituents Dataset, the proposed algorithm achieved the lowest MAE of 0.078 and MSE of 0.036, with an R^2 value of 0.944 and an NDCG score of 0.436. Additionally, the model continued to perform well across multiple datasets, demonstrating its advantage in prediction accuracy. However, this study has certain limitations. The model's performance exhibits a degree of dependency on the quality of input data and the prompt design, both of which may necessitate further refinement to accommodate varying market conditions and diverse financial instruments. Additionally, the deployment of the model in real-time trading environments warrants further investigation to ensure its resilience and adaptability in the face of rapidly evolving market dynamics. Future research will prioritize enhancing the model's robustness by integrating advanced machine learning methodologies, such as reinforcement learning and adversarial training. Moreover, extending the model's application to a broader range of financial markets and instruments will contribute to a more comprehensive understanding of its generalizability and efficacy.

References

[1] Zhao C, et al. Progress and prospects of data-driven stock price forecasting research. International Journal of Cognitive Computing in Engineering. 2023 June;4:100-108.

[2] C. L. Yingda Tang. Examining the Factors of Corporate Frauds in Chinese A-share Listed Enterprises. OAJRC Social Science. 2023;4(3):63-77.

[3] Sheth, D, & Shah, M. Predicting stock market using machine learning: best and accurate way to know future stock prices. Int J Syst Assur Eng Manag. 2023;14(1):1-18.

[4] Pashankar SS, Shendage JD, & Pawar J. Machine Learning Techniques for Stock Price Prediction-A Comparative Analysis of Linear Regression, Random Forest, and Support Vector Regression. Journal Of Advanced Zoology. 2024 March;45(S4):118-127.

[5] Nourbakhsh, Z, & Habibi, N. Combining LSTM and CNN methods and fundamental analysis for stock price trend prediction. Multimed Tools Appl. 2023;82(12):17769-17799.

[6] Olbrys J, & Mursztyn M. Measuring stock market resiliency with Discrete Fourier Transform for high frequency data. Physica A: Statistical Mechanics and its Applications. 2019;513:248-256.

[7] Sharma DK, Hota HS, & Rababaah AR. Forecasting US stock price using hybrid of wavelet transforms and adaptive neuro fuzzy inference system. International Journal of System Assurance Engineering and Management. 2024;15(2):591-608.

[8] Zhang G, & Zhou, T. Finite Element Model Calibration with Surrogate Model-Based Bayesian Updating: A Case Study of Motor

FEM Model. Innovations in Applied Engineering and Technology. 2024;3(1):1-13.
doi: https://doi.org/10.62836/iaet.v3i1.232.

[9] Z. Wang, Y. Zhao, C. Song, X. Wang, & Y. Li. A new interpretation on structural reliability updating with adaptive batch sampling-based subset simulation. Structural and Multidisciplinary Optimization. 2024;67(1):7.

[10] X. Chen & H. Zhang. Performance Enhancement of AlGaN-based Deep Ultraviolet Light-emitting Diodes with AlxGa1-xN Linear Descending Layers. Innovations in Applied Engineering and Technology. 2023;1-10.

[11] Y. Zhu, Y. Zhao, C. Song, & Z. Wang. Evolving reliability assessment of systems using active learning-based surrogate modelling. Physica D: Nonlinear Phenomena. 2024;457:133957.

[12] Y. Jia & J. Lei. Experimental Study on the Performance of Frictional Drag Reducer with Low Gravity Solids. Innovations in Applied Engineering and Technology. 2024;1-22.

[13] Yañez C, Kristjanpoller W, & Minutolo MC. Stock market index prediction using transformer neural network models and frequency decomposition. Neural Computing and Applications. 2024;36:1-21.

[14] Y. Gan & D. Zhu. The Research on Intelligent News Advertisement Recommendation Algorithm Based on Prompt Learning in End-to-End Large Language Model Architecture. Innovations in Applied Engineering and Technology. 2024;1-19.

[15] Y. Tang. Investigating the Impact of Digital Transformation on Equity Financing: Empirical Evidence from Chinese A-share Listed Enterprises. Journal of Humanities, Arts and Social Science. 2024;8(7):1620-1632.
doi: https://dx.doi.org/10.26855/jhass.2024.07.014.

[16] Y. Tang & C. Li. Exploring the Factors of Supply Chain Concentration in Chinese A-Share Listed Enterprises. Journal of Computational Methods in Engineering Applications. 2023;1-17.

[17] C. Li & Y. Tang. The Factors of Brand Reputation in Chinese Luxury Fashion Brands. Journal of Integrated Social Sciences and Humanities. 2023;1-14.

[18] P. Ren, Z. Zhao, & Q. Yang. Exploring the Path of Transformation and Development for Study Abroad Consultancy Firms in China. arXiv preprint arXiv. 2024;2404:11034.

[19] Z. Zhao, P. Ren, & Q. Yang. Student self-management, academic achievement: Exploring the mediating role of self-efficacy and the moderating influence of gender insights from a survey conducted in 3 universities in America. arXiv preprint arXiv. 2024;2404:11029.

[20] S. Xiong, X. Chen, & H. Zhang. Deep Learning-Based Multifunctional End-to-End Model for Optical Character Classification and Denoising. Journal of Computational Methods in Engineering Applications. 2023;1-13.

[21] S. Xiong, H. Zhang, M. Wang, & N. Zhou. Distributed Data Parallel Acceleration-Based Generative Adversarial Network for Fingerprint Generation. Innovations in Applied Engineering and Technology. 2022;1-12.

[22] C. Li & Y. Tang. Emotional Value in Experiential Marketing: Driving Factors for Sales Growth–A Quantitative Study from the Eastern Coastal Region. Economics & Management Information. 2024;1-13.

[23] Q. Zhang, Y. Zhang, F. Bao, Y. Liu, C. Zhang, & P. Liu. Incorporating stock prices and text for stock movement prediction based on information fusion. Engineering Applications of Artificial Intelligence. 2024;127:107377.

[24] S. Khan, & H. Alghulaiakh. ARIMA model for accurate time series stocks forecasting. International Journal of Advanced Computer Science and Applications. 2020;11(7).

[25] L. Wang, F. Ma, J. Liu, & L. Yang. Forecasting stock price volatility: New evidence from the GARCH-MIDAS model. International Journal of Forecasting. 2020;36(2):684-694.

[26] H. N. Bhandari, B. Rimal, N. R. Pokhrel, R. Rimal, K. R. Dahal, & R. K. Khatri. Predicting stock market index using LSTM. Machine Learning with Applications. 2022;9:100320.

[27] B. Gülmez. Stock price prediction with optimized deep LSTM network with artificial rabbits optimization algorithm. Expert Systems with Applications. 2023;227:120346.

[28] Y. A. GIWA & S. Studies. Over the counter stocks data acquisition & analysis with time series prediction. International Journal of Social Sciences and Scientific Studies. 2024;4(1):3643-3670.

[29] D. Song, A. M. C. Baek, & N. Kim. Forecasting stock market indices using padding-based fourier transform denoising and time series deep learning models. IEEE Access. 2021;9:83786-83796.

[30] S. Verma, S. P. Sahu, & T. P. Sahu. Discrete wavelet transform-based feature engineering for stock market prediction. International Journal of Information Technology. 2023;15(2):1179-1188.

[31] Y. Luo. Wavelet Neural Network Model with Time-Frequency Analysis for Accurate Share Prices Prediction. In Intelligent Computing: Proceedings of the 2021 Computing Conference, Volume 3, 2021: Springer, pp. 286-297.

[32] D. Cao, et al. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. 2023.

[33] T. Guo, & E. Hauptmann. Fine-Tuning Large Language Models for Stock Return Prediction Using Newsflow. 2024.

[34] Y. Xiao, J. Huang, & J. Yang. TFCSRec: Time–frequency consistency based contrastive learning for sequential recommendation. Expert Systems with Applications. 2024;245:123118.

[35] S. Xie, Q. Pan, X. Wang, X. Luo, & V. Sugumaran. Combining prompt learning with contextual semantics for inductive relation prediction. Expert Systems with Applications. 2024;238:121669.

[36] Y. Jiao, & J. Jakubowicz. Predicting stock movement direction with machine learning: An extensive study on S&P 500 stocks. In 2017 IEEE International Conference on Big Data (Big Data), 2017: IEEE, pp. 4705-4713.

[37] S. Feng, C. Xu, Y. Zuo, G. Chen, F. Lin, & J. XiaHou. Relation-aware dynamic attributed graph attention network for stocks recommendation. Pattern Recognition. 2022;121:108119.

[38] Q. Bui, & R. Ślepaczuk. Applying Hurst Exponent in pair trading strategies on Nasdaq 100 index. Physica A: Statistical Mechanics and its Applications. 2022;592:126784.

[39] V. Cioli, T. Giacomo, A. Giannozzi, & O. Roggi. Stock Liquidity and Corporate Investment Policy after FTSE 100 Index Additions. International Journal of Business and Management. 2023;17(10):1-20.

[40] Z. Yang, T. Zhao, S. Wang, & X. Li. MDF-DMC: A stock prediction model combining multi-view stock data features with dynamic market correlation information. Expert Systems with Applications. 2024;238:122134.

[41] Y. Liu, S. Huang, X. Tian, F. Zhang, F. Zhao, & C. Zhang. A stock series prediction model based on variational mode decomposition and dual-channel attention network. Expert Systems with Applications. 2024;238:121708.

[42] M. Lu, & X. Xu. TRNN: An efficient time-series recurrent neural network for stock price prediction. Information Sciences. 2024;657:119951.

[43] K. Chaudhari, & A. Thakkar. Data fusion with factored quantization for stock trend prediction using neural networks. Information Processing & Management. 2023;60(3):103293.

[44] N. Wijerathne, et al. Deep Learning Based Personalized Stock Recommender System. In International Conference on Neural Information Processing, 2023: Springer, pp. 362-374.

[45] A. Mahmoodi, L. Hashemi, M. Jasemi, S. Mehraban, & J. Laliberté, R. C. Millar. A developed stock price forecasting model using support vector machine combined with metaheuristic algorithms. OPSEARCH. 2023;60(1):59-86.