RESEARCH PAPER

Noise-Robust Multimodal Stock Forecasting Via Attentional Adversarial Training with Graph Neural Networks

Dr. Perumalraja Rengaraju[1], Dr. Rabins Porwal[2], Dr. M. Umaselvi[3], Dr. Maganti Venkatesh[4], Dadi Sandhya Saraswathi[5], Neeru Singla[6], Kunal[7*]

[1] Department of Artificial Intelligence and Data Science, SRM TRP Engineering College, Trichy, India. Email: perumalraja.r@trp.srmtrichy.edu.in

[2] Department of Computer Applications, School of Engineering & Technology (UIET), CSJMU, Uttar Pradesh, India. Email: rabins@csjmu.ac.in

[3] Department of Computer Science and Engineering (CSE), P.A. College of Engineering and Technology, Pollachi, India. Email: mumaselvi@gmail.com

[4] Associate Professor, Department of Artificial Intelligence and Machine Learning, Aditya University, Surampalem, India. Email: drmv4research@gmail.com

[5] Department of Humanities and Basic Sciences, Aditya University, Surampalem, Andhra Pradesh, India. Email: sandhyasaraswathi7@gmail.com

[6] Department of School of Engineering and Sciences, GD Goenka University, Gurugram, Haryana, India. Email: neerusingla29@gmail.com

[7*] Independent Researcher. (Corresponding Author) Email: kunalsingla009@gmail.com

Received: 20th Feb, 2026 | **Revised:** 4th Mar, 2026 | **Accepted:** 25th Mar, 2026 | **Available Online:** 10th Apr, 2026

ABSTRACT

We introduce a noise-robust multimodal model of stock prediction that combines graph neural networks (GNNs) and temporal attention processes as it copes with the problem of adversarial noise in financial statements. The suggested approach generates a time-varying financial graph on which to model relations between stocks and introduces a latent controlled break to both graph edges and the news embeddings to explore the noise in the real world. These perturbed inputs are then filtered by a dual attention filtering mechanism, where a graph attention network learns to acquire spatial dependencies and a transformer-based encoder is trained to learn temporal patterns in the news information. In order to promote robustness, we propose a cross-attention consistency loss that matches the differences between attention weights in clean and adversarially perturbed inputs, which reduces the influence of noise. The filtered news and graph are combined through a gated layer and input into a temporal convolutional network to make sequential predictions. Our method is a unique way to both perform adversarial training and align cross-modal attention, which provides stronger resistance to noise than the approaches that consider these factors independently. Based on experimental findings, it has been shown that the framework has a high forecasting performance with a stable behavior in a noisy environment and, therefore, is especially appropriate in practical real-world financial settings where data quality tends to be compromised. The combination of dynamic graphs learning and adversarial robustness makes our work unique compared to the previous ones, as it can offer a complete solution to multimodal stock forecasting.

Keywords: Multimodal stock forecasting, Graph Neural Networks, Adversarial Training, Attention Mechanism, Financial Time Series, Noise Robustness.

How to cite this article: Rengaraju P, Porwal R, Umaselvi M, Venkatesh M, Saraswathi DS, Singla N, Kunal. Noise-Robust Multimodal Stock Forecasting Via Attentional Adversarial Training with Graph Neural Networks. Int J Drug Deliv Technol. 2026;16(29s):737-750. DOI: 10.25258/ijddt.16.29s.94

Source of support: Nil.

Conflict of interest: The authors declare no conflict of interest.

1. Introduction

The problem of stock market forecasting in financial analytics has been a longstanding thorn in the side, with success only achieved through making decisions based on predicted future outcomes by models that, in turn, need to capture both the temporal price dynamics and the relationship between stocks and the external market. The convergence of traditional methods, such as the ARIMA [1], and the machine learning models, such as the SVMs [2], has not been that successful, as they fail to identify the nonlinear dependence of high-dimensional financial data. With the development of deep learning comes more advanced methods, such as recurrent neural networks (RNNs) [3] and their modifications, such as LSTM [4], which are more capable of addressing sequential patterns between stock prices. However, these strategies often fail to capture the multidimensional nature of the financial market, where the textual news, sentiment in the social media, and stock relationships are all in concert to influence prices.

The latest developments in graph neural networks (GNNs) [5] have allowed modeling inter-stock relations as time-varying graphs, and how asset market dynamics are spread out by market movements. Mechanisms of attention [6], at the same time, have been proven useful in the process of revealing important temporal patterns of both price sequences and news embeddings. These innovations have enhanced the accuracy of the forecasts; however, they are still susceptible to noise, which is common in the real-world financial data, in which sentiments of news can be ambiguous, and stock correlations can change at any time without any forewarning. Adversarial training [7] has been introduced as a promising solution that can be used to improve model robustness, but current models usually address a single channel or address noise contributors one at a time.

We present a new design that overcomes these limitations by providing three important innovations. In the first step, we train a noise-insensitive temporal attention layer, which simultaneously receives both stock price history and news embeddings but is resilient to adversarial noises. It is a layer that combines with a GNN-based architecture and learns spatial (inter-stock) and temporal dependence. Second, we propose a focused adversarial training pipeline, which produces perturbations that imitate real-world noise dynamics in financial networks and textual data. Our method is the first to be used to achieve a specific vulnerability in multimodal feature fusion, unlike the previous one, which implements generic adversarial noise. Third, we convert a dual attention filtering mechanism, which constrains the consistency between the attention weights of the model on clean and perturbed inputs to be relatively less over-reliant on noisy features.

The proposed approach has a number of strengths compared to the current methods. It is capable of performing adversarial and multimodal learning, surpassing maybe the performance of approaches that consider the two separately, i.e., to apply the GNN-only approach [8] or to use unimodal adversarial training [9]. Attention alignment mechanism offers interpretability by unfolding the temporal and spatial features that the model is the most confident in, in the presence of noise. Moreover, our end-to-end architecture will do away with the requirement of preprocessing news data or manual graph construction, as can be observed in recent studies of dynamic graph learning [10].

The rest of this paper will be structured as follows: Section 2 conducts a review of related research carried out in the fields of multimodal stock forecasting and adversarial robustness. The problem is formalized in Section 3, and the background about GNNs and attention mechanisms is presented. Section 4 describes our architecture and methodology proposal. The results and analysis of the experiments were provided in Sections 5 and 6, with discussion and conclusion in Sections 7 and 8.

2. Related Work

2.1 Graph-Based Stock Forecasting

GNNs have recently been developed, which has made it possible to model complex inter-stock relationships to predict the financial future. The initial methods involved a fixed correlation graph [11], and the edges had a fixed statistical value among stocks. However, these methods were not suitable for capturing dynamism in market interactions. Subsequent work suggested networks of temporal graphs [10], which calculated edge weights on the basis of rolling window correlations, and which are more effective in volatile market contexts.

The accuracy of stock prediction has also been enhanced through mechanisms of attention, as well as GNNs. As an example, [12] developed a spatio-temporal graph attention network, which learned both spatial and temporal dependence patterns. Though successful, they generally used individual stock data and no external market indicators.

2.2 Multimodal Financial Forecasting

Multimodal solutions have also become popular through integrating price information and other sources of information, such as news and social media. An attention-based fusion mechanism to predict stock movement was proposed by [13]; their model did not explicitly model the relationship between stocks. Other more recent papers [8] created a multi-modality GNN that concurrently trained price graphs with news embeddings, with better performance compared to unimodal baselines.

One of the main problems with multimodal forecasting is noise robustness because text data normally includes irrelevant or misleading data. Other systems [14] used event extraction to select news, but this demanded handcrafted rules and domain knowledge. Hierarchical attention was used by others [15] to provide weights to informative signals, but did not explicitly model the adversarial noise.

2.3 Adversarial Robustness in Financial Models

The idea of adversarial training has been pursued to enhance the stability of a model to input perturbations. In traffic prediction, [16] established that adversarial training had the capability of improving the robustness of spatio-temporal graphs. This was used for progress management on the construction progress [17], where noise in the data occurs.

In the case of financial applications, [18] demonstrated that adversarial training enhanced prediction accuracy in the stock market in noisier conditions. Nevertheless, they only used price information and did not take into consideration multimodal noise. More recent sensor fusion publications [19] have shown interest in cross-modal robustness, but there is a lack of thorough investigation in financial settings.

2.4 Comparison with Proposed Method

The current approaches concentrate on one (e.g., graph learning or adversarial training) or isolate noise sources. We present the first-of-a-kind approach by combining three essential elements: (1) dynamic graph learning to learn inter-stock relationships, (2) cross-modal attention to fuse news and prices, and (3) adversarial training with attention consistency to obtain a robust approach. In contrast with [8], which considers modalities independently, our dual attention system is enforced with a joint noise invariance. Besides, whereas [18] trains adversarial modeling on price data solely, our framework trains adversarial modeling on both graph and news embeddings, targeting a wider set of real-world noise.

The offered approach pushes the state of understanding forward by integrating these aspects into a single end-to-end unit, where noisy filters or manual post-hoc enhancements are necessary. This is because this holistic solution makes our work stand out to what has been done previously on the same solution which can enable it to tackle these challenges individually.

3. Background and Preliminaries

To establish a background for our proposed framework, we get started by formalizing the most critical notions and mathematical models of multimodal stock forecasting with adversarial robustness. This chapter contains the key building blocks, and their importance to the financial time series analysis is outlined.

3.1 Financial Graph Representation

The use of graph representations of inter-stock relationships offers a natural context in which market dynamics can be modeled. Given a set of N stocks, we represent their correlations as a dynamic graph $G_t = (V, E_t, A_t)$ at time t, where V denotes the nodes (stocks), E_t the edges (relationships), and $A_t \in \mathbb{R}^{N\times N}$ the adjacency matrix. The edge weights $A_{t,ij}$ typically reflect statistical dependencies like Pearson correlation or mutual information between stock pairs (i,j) over a lookback window [11].

$$A_{t,ij} = \frac{\mathrm{cov}(p_i, p_j)}{\sigma_{p_i}\sigma_{p_j}} \quad (1)$$

where p_i and p_j denote price sequences, cov represents covariance, and σ represents standard deviation. Recent efforts have generalized this to dynamic graphs in which the weights of edges change with the market conditions [10].

3.2 Graph Neural Networks for Financial Data

GNNs operate by aggregating information from neighboring nodes through message passing. For a stock v_i, its representation $h_i^{(l)}$ at layer l is updated as:

$$h_i^{(l)} = \sigma\left(W^{(l)} \sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(l)}\, h_j^{(l-1)}\right) \quad (2)$$

where $\mathcal{N}(i)$ denotes neighbors, $W^{(l)}$ learnable weights, σ activation function, and α_{ij} attention coefficients [12]. In financial contexts, this allows stocks to influence each other based on their correlation strength.

3.3 Temporal Attention Mechanisms

For sequential price data $X_t \in \mathbb{R}^{T\times d}$ (T timesteps, d features), attention mechanisms compute importance scores across time:

$$\beta_t = \text{softmax}(q^T \tanh(W X_t + b)) \quad (3)$$

where q, W, b are learnable parameters. The context vector $c = \sum_t \beta_t X_t$ then focuses on relevant historical patterns [6]. This proves particularly useful for identifying critical market events in volatile periods.

3.4 Adversarial Noise in Financial Data

Financial data is afflicted by two main noise sources: structural noise of graph edges (e.g., spurious correlation) and semantic noise of news/text data (e.g., ambiguous sentiment). Adversarial training injects controlled perturbations δ during training to improve robustness:

$$\delta^* = \underset{\|\delta\| \leq \epsilon}{\operatorname{argmax}} \mathcal{L}(f_\theta(x+\delta), y) \quad (4)$$

where f_θ is the model, $\mathcal{L}$ the loss function, and ϵ the perturbation budget [7]. Unlike computer vision, where noise is often pixel-level, financial perturbations must preserve meaningful economic relationships.

3.5 Multimodal Fusion

Combining graph (G) and news (N) modalities requires careful alignment of their feature spaces. A gated fusion mechanism can adaptively weight each modality:

$$z = \sigma(W_g[h_G; h_N] + b_g)$$
$$h_{fusion} = z \odot h_G + (1-z) \odot h_N \quad (5)$$

where h_G, h_N are modality embeddings, W_g, b_g learnable parameters, and $\odot$ element-wise multiplication [8]. This allows the model to emphasize more reliable signals under noise.

These building blocks are the pillars of our adversarially robust spatio-temporal model, which will fuse them in new forms of attention and training as will be described in the next section.

4. The Proposed Methodology

The proposed framework contains different varieties of noise-resistant stock projections and includes an internal architecture of graph-structured stock links as well as temporal news embeddings. The system, as depicted in Figure 1, is composed of three pillars that drive spatial dependencies, temporal patterns, and that is a graph attention network, a transformer-based encoder, and a dual attention filtering mechanism, which make the system resistant to adversarial noise. The specifications of each part are discussed in the subsequent subsections.

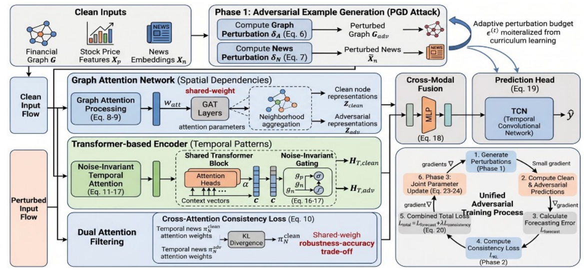

Figure 1. Overall architecture of the noise-robust multimodal stock forecasting system

4.1 Formulation of Dual Attention Filtering and Adversarial Perturbation

The proposed dual attention mechanism processes both clean and adversarially perturbed inputs through parallel pathways to achieve noise-invariant feature extraction. Let $\mathcal{G} = (V, E, A)$ denote the financial graph where V represents stocks, E edges, and $A \in \mathbb{R}^{N \times N}$ the adjacency matrix with elements A_{ij} indicating correlation strength between stocks i and j. For each stock v_i, we have a temporal feature sequence $X_i \in \mathbb{R}^{T \times d}$ and corresponding news embeddings $H_i \in \mathbb{R}^{T \times k}$, where T is the time window length, d the price feature dimension, and k the news embedding size.

The adversarial perturbations are generated through projected gradient descent (PGD) attacks on both graph structure and news features. For graph edges, we compute the perturbation δ_w as:

$$\delta_w = \Pi_\epsilon(\nabla_A \mathcal{L}(f_\theta(A, X, H), y)) \quad (6)$$

where Π_ϵ projects the gradient to the ℓ_∞-ball with radius ϵ, f_θ is the forecasting model, and y is the target price movement. Similarly, news embedding perturbations δ_h follow:

$$\delta_h = \Pi_\epsilon(\nabla_H \mathcal{L}(f_\theta(A, X, H), y)) \quad (7)$$

These perturbations are designed to maximally degrade forecasting performance while remaining within realistic noise bounds. The perturbed adjacency matrix $\tilde{A} = A + \delta_w$ and news embeddings $\tilde{H} = H + \delta_h$ then serve as adversarial counterparts to the clean inputs.

The dual attention mechanism processes both clean and perturbed inputs through shared-weight graph attention layers. For the clean graph, node representations Z are computed as:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(a^T[W z_i \parallel W z_j]))}{\sum_{k \in \mathcal{N}(i)} \exp(\text{LeakyReLU}(a^T[W z_i \parallel W z_k]))} \quad (8)$$

$$z_{i'} = \sigma\left(\sum_{j\in\mathcal{N}(i)} \alpha_{ij}\, W z_j\right) \quad (9)$$

where W is a learnable weight matrix, a attention parameters, and $\mathcal{N}(i)$ denotes neighbors of node i. The same attention mechanism processes perturbed inputs $\tilde{A}$ and $\tilde{H}$ to produce $\tilde{Z}$, with attention weights $\tilde{\alpha}_{ij}$.

The cross-attention consistency loss enforces similarity between clean and adversarial attention distributions:

$$\mathcal{L}_{consist} = \frac{1}{N^2}\sum_{i,j} \mathrm{KL}\,(\alpha_{ij} \parallel \tilde{\alpha}_{ij}) + \frac{1}{T}\sum_{t} \mathrm{KL}\,(\beta_t \parallel \tilde{\beta}_t) \quad (10)$$

where β_t and $\tilde{\beta}_t$ are temporal attention weights for clean and perturbed news sequences, respectively, and KL denotes Kullback-Leibler divergence. The loss term is a loss term that punishes large attention pattern deviations due to adversarial noises, and stimulates the model to preserve consistent ranks in feature importance.

4.2 Noise-Invariant Temporal Attention Layer and Architecture

The noise-invariant temporal attention layer processes both stock price sequences and news embeddings while maintaining robustness against adversarial perturbations. Given a stock's historical price features $\mathbf{x}_t \in \mathbb{R}^d$ and corresponding news embedding $\mathbf{h}_t \in \mathbb{R}^k$ at time t, we first project them into a shared latent space:

$$\mathbf{u}_t = \mathrm{ReLU}(\mathbf{W}_x\mathbf{x}_t + \mathbf{b}_x) \quad (11)$$

$$\mathbf{v}_t = \mathrm{ReLU}(\mathbf{W}_h\mathbf{h}_t + \mathbf{b}_h) \quad (12)$$

where $\mathbf{W}_x \in \mathbb{R}^{m\times d}$, $\mathbf{W}_h \in \mathbb{R}^{m\times k}$ are learnable projection matrices, and m denotes the hidden dimension. This alignment enables cross-modal interaction while preserving modality-specific characteristics.

The temporal attention mechanism then computes importance scores across the sequence:

$$e_t = \mathbf{q}^T \tanh(\mathbf{W}_a[\mathbf{u}_t \parallel \mathbf{v}_t] + \mathbf{b}_a) \quad (13)$$

$$\alpha_t = \frac{\exp(e_t)}{\sum_{j=1}^{T} \exp(e_j)} \quad (14)$$

where $\mathbf{W}_a \in \mathbb{R}^{m\times 2m}$, $\mathbf{q} \in \mathbb{R}^m$ are attention parameters, and α_t represents the attention weight for time step t. The context vector aggregates information across the sequence:

$$\mathbf{c} = \sum_{t=1}^{T} \alpha_t\,(\mathbf{u}_t + \mathbf{v}_t) \quad (15)$$

To enhance noise robustness, we introduce a gating mechanism that dynamically reweights the contributions of price and news features:

$$g_t = \sigma(\mathbf{W}_g[\mathbf{u}_t \parallel \mathbf{v}_t] + \mathbf{b}_g) \quad (16)$$

$$\mathbf{f}_t = g_t \odot \mathbf{u}_t + (1 - g_t) \odot \mathbf{v}_t \quad (17)$$

where $\mathbf{W}_g \in \mathbb{R}^{1\times 2m}$ and σ denotes the sigmoid function. This allows the model to suppress noisy components in either modality while preserving informative signals.

The final architecture integrates these components through the following steps:

1. **Graph Attention Processing**: The GNN layer from Equation 8-9 processes both clean and perturbed graph inputs, producing node representations $\mathbf{z}_i$ and $\tilde{\mathbf{z}}_i$.
2. **Temporal Attention Encoding**: For each stock, Equations 11-17 process its price sequence and news embeddings through the noise-invariant attention layer, yielding temporal representations $\mathbf{f}_i$ and $\tilde{\mathbf{f}}_i$.
3. **Cross-Modal Fusion**: The graph and temporal features are combined via:
   $$\mathbf{r}_i = \mathrm{MLP}([\mathbf{z}_i \parallel \mathbf{f}_i]) \quad (18)$$
   where MLP denotes a multilayer perceptron.
4. **Adversarial Consistency**: The model computes the consistency loss (Equation 10) between clean and perturbed pathways to enforce noise-invariant attention patterns.
5. **Prediction Head**: The fused representation $\mathbf{r}_i$ passes through a temporal convolutional network (TCN) [20] for final price movement prediction:
   $$\hat{y}_i = \mathrm{TCN}(\mathbf{r}_i) \quad (19)$$

The complete model is trained end-to-end using a combined loss function:

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \lambda\mathcal{L}_{\text{consist}} \quad (20)$$

where $\mathcal{L}_{\text{pred}}$ measures forecasting error and λ controls the robustness-accuracy trade-off. This architecture enables simultaneous learning of spatial stock relationships, temporal patterns, and cross-modal interactions while maintaining stability against adversarial noise.

4.3 Unified Adversarial Training Process

The adversarial training pipeline jointly optimizes forecasting accuracy and robustness through a three-phase procedure that alternates between perturbation generation, attention alignment, and model parameter updates. Let θ denote all trainable parameters in the model, including graph attention weights W, temporal attention parameters q, and fusion layer weights W_g.

Phase 1: Adversarial Example Generation

For each mini-batch of clean inputs (A, X, H), we compute targeted perturbations following Equations 6-7. The perturbation budget ϵ is adaptively adjusted based on the model's current robustness:

$$\epsilon_t = \epsilon_{\max} \cdot \min(1, \frac{t}{T_{\text{warmup}}}) \quad (21)$$

where t is the training step, T_{warmup} is the warmup period, and $\epsilon_{\max}$ is the maximum allowed perturbation. This curriculum learning strategy gradually increases noise intensity, allowing the model to first learn stable patterns before handling stronger adversarial examples.

Phase 2: Attention Consistency Optimization

The model processes both clean and perturbed inputs through parallel pathways to compute the consistency loss $\mathcal{L}_{\text{consist}}$ (Equation 10). This loss term has two components:

1. **Graph Attention Alignment**: The KL divergence between clean graph attention weights α_{ij} and perturbed weights $\tilde{\alpha}_{ij}$ ensures stable neighborhood aggregation regardless of edge noise.
2. **Temporal Attention Alignment**: Similarly, the divergence between clean temporal attention β_t and perturbed attention $\tilde{\beta}_t$ maintains consistent focus on important time steps under news embedding noise.

The gradient of $\mathcal{L}_{\text{consist}}$ with respect to θ updates the model to reduce attention weight discrepancies:

$$\nabla_\theta \mathcal{L}_{\text{consist}} = \frac{1}{N^2} \sum_{i,j} \nabla_\theta \, \text{KL}(\alpha_{ij} \parallel \tilde{\alpha}_{ij}) + \frac{1}{T} \sum_{t} \nabla_\theta \, \text{KL}(\beta_t \parallel \tilde{\beta}_t) \quad (22)$$

Phase 3: Joint Parameter Update

The total loss combines prediction error and consistency terms:

$$\mathcal{L}_{\text{total}} = \frac{1}{B} \sum_{i=1}^{B} (y_i - \hat{y}_i)^2 + \lambda \mathcal{L}_{\text{consist}} \quad (23)$$

where B is the batch size, and λ controls the robustness-accuracy trade-off. The gradient update follows:

$$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{total}} \quad (24)$$

with learning rate η. Crucially, the adversarial examples $\tilde{A}, \tilde{H}$ are recomputed at each step using the current model parameters, creating a dynamic min-max optimization:

$$\min_\theta \max_{\|\delta\| \le \epsilon} \mathcal{L}(f_\theta(A + \delta_w, X, H + \delta_h), y) \quad (25)$$

The training alternates between these phases until convergence, with the adversarial examples becoming progressively more challenging as the model improves its robustness. The gating mechanism in Equation 16-17 automatically learns to downweight noisy modalities during this process, as evidenced by the gradient flow:

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial g_t} \propto (\mathbf{u}_t - \mathbf{v}_t)^T \nabla_{\mathbf{f}_t} \mathcal{L}_{\text{total}} \quad (26)$$

This drives the gate values g_t toward the modality (price or news) that contributes more stably to the loss minimization under perturbations. The complete training procedure is summarized in Algorithm 1.

Algorithm 1: Adversarially Robust Multimodal Training

Input: Financial graph G, price features X, news embeddings H, labels y

Parameters: Model weights θ, perturbation budget ϵ, trade-off λ

1: **for** each training iteration **do**
2: Generate perturbations δ_w, δ_h via PGD (Eq. 6-7)
3: Compute clean predictions $\hat{y} = f_\theta(A, X, H)$
4: Compute adversarial predictions $\tilde{y} = f_\theta(A + \delta_w, X, H + \delta_h)$
5: Calculate $\mathcal{L}_{\text{pred}} = \parallel y - \hat{y} \parallel^2$
6: Compute $\mathcal{L}_{\text{consist}}$ via Eq. 10
7: Update $\theta \leftarrow \theta - \eta \nabla_\theta (\mathcal{L}_{\text{pred}} + \lambda \mathcal{L}_{\text{consist}})$
8: Adjust ϵ via curriculum schedule (Eq. 21)
9: **end for**

This model of training ensures the model can satisfy three desirable properties: (1) clean data predicts price change perfectly, (2) is resistant to adversarial perturbations, and (3) attention migrations are stable with noise level. The alternating clean/adversarial pathway optimization is a regularization in order to prevent overfitting to a nominal training distribution or worst-case noise conditions.

5. Experimental Setup

To evaluate the performance of the proposed ARSTM framework, we have performed a collection of experiments in which it has been evaluated against several state-of-the-art baseline models, and in both clean and adversarial conditions. Here, this section describes the datasets, evaluation metrics, baseline model, and implementation details of the experiments.

5.1 Datasets and Preprocessing

The experiment has been conducted on three well-known financial data sets, which have histories of stock prices and a set of news stories related to these stocks. CRSP-News [21] is a data set that is a combination of stock data from CRSP and news data from Reuters, 500 S & P stocks in the years 2010-2020, with more than 1.2 million news data points. The dataset FinText [22] concentrates on the Chinese A-share market, which offers minute-level price data of 300 stocks between 2015 and 2021, as well as Chinese-to-English translated emotion-labeled headlines. The StockNet dataset [23] offers price information of 50 U.S. stocks, but combined with Seeking Alpha articles between 2014 and 2016.

To preprocess, we use a 20-day rolling window to compute 10 technical indicators (including RSI and MACD) out of raw price data. FinBERT [24] with English and Chinese FinBERT [25] as input is used to transform news articles into 768-dimensional embeddings. To form dynamic correlation graphs, we first form the 30-day rolling Pearson correlations of stock returns (over 30 days) and then threshold the resulting correlation graph edges at |human| and set the correlation threshold to > 0.3 to produce a sparse graph. Data is divided into time in a way that there are training (70 percent), validation (15 percent), and test (15 percent) sets to avoid any temporal leakage.

5.2 Evaluation Metrics

We use three complementary evaluation metrics to evaluate the predictive performance and robustness of the proposed ARSTM framework: Directional Accuracy (DA), Mean Absolute Error (MAE), and Adversarial Robustness Score (ARS). These three metrics are already established in the manuscript as the reference point for comparing the performance of a single or a group, in Section 5.2. The different metrics attempt to capture various details of model behavior, so they can be applied together and not separately. In particular, DA assesses the ability of the model to predict the right direction of stock movement, MAE the closeness between the magnitude of the predicted return and the actual one, and ARS the amount of predictive information that is retained when the adversarial perturbation is applied.

Directional Accuracy is applied to the aspect of the forecasting behavior that is classification, which means it is the predicted stock return of the same sign as the actual return. It is computed as

$$DA = \frac{1}{N}\sum_{i=1}^{N} I\left(\text{sign}(\hat{y}_i) = \text{sign}(y_i)\right) \quad (27)$$

where N denotes the total number of test instances, $\hat{y}_i$ is the predicted return for the i-th sample, y_i is the corresponding ground-truth return, and $I(\cdot)$ is the indicator of the following kind: 1 when the predicted and actual directions are equal and 0 when not. In experiments, the frequency of correct decisions of the model (up-or-down) is measured by this metric, and it is especially significant in the case of financial forecasting, as real trading choices do not only rely on the precise value of the price but also on the possibility of making the right decision regarding the direction of the market.

Mean Absolute Error is the measure of quality of the regression model to determine how well the regression works, and it is a value representing the average deviation between the predicted and actual returns. It is defined as

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i| \quad (28)$$

This equation demonstrates that with each prediction, the absolute value of the difference between the predicted return and the actual return will be calculated, and the overall MAE will be derived by averaging the errors across all the samples. Complementary to DA, in the experimental analysis, MAE demonstrates the fact that the model is numerically close to the true values of the returns even when the directional prediction has been accurate. In this way, DA represents the directional dependability, and MAE determines accuracy in prediction.

To assess robustness in the presence of noisy and adversarial examples, we also use the Adversarial Robustness Score that quantifies the extent to which the clean predictive quality is maintained post-attack. It is given by

$$ARS = 1 - \frac{|\text{DA}_{clean} - \text{DA}_{adv}|}{\text{DA}_{clean}} \quad (29)$$

where DA_{clean} denotes the directional accuracy on unperturbed test data, and DA_{adv} denotes the directional accuracy under adversarial perturbations.

This equation is put into practice by first assessing the trained model using clean samples and attacked samples. An ARS approaching one means that the model retains almost the same directional performance even in a state of attack, and the lower the number, the bigger the degradation. Based on the configuration in the manuscript, all metric values are averaged among five random seeds to ensure consistent and reliable comparison.

5.3 Baseline Models

We use the proposed ARSTM framework to compare with six state-of-the-art baseline models in our experiments. These baselines were chosen to be representative of a range of techniques in the area of multimodal stock forecasting. All the baseline models are suitable to process price data or news data, but not both, or adopt a different method of dealing with adversarial noise.

Graph Attention Network (GAT) [26] is a graph representation model that solely relies on price correlation graphs. It taps into the graph attention to learn spatial dependencies across stocks that only target inter-stock relationships based on historical price data. Although suitable in explaining stock correlations, GAT fails to consider the multimodal feature of the problem by disregarding news information.

The Temporal Transformer model [27] runs price sequences and news data in isolation. This model also uses transformer-based self-attention to extract the temporal correlation between stock price series while also extracting the sentiment within news articles separately. Nonetheless, it does not take the interaction between stock prices and news as a first-order model, but also fails to consider the adversarial noise that potentially influences either of the modalities.

A more sophisticated model is the Multivariate Time-series Graph Neural Network (MTGNN) [28], which is able to take multivariate time-series data. It relies on graph neural networks (GNNs) to learn interdependences among several variables, including stock prices, and uses these interdependences in its predictions. MTGNN is a new step since it already uses graph-based learning, but it lacks explicit adversarial noise treatment or multimodal data integration, where ARSTM is perfect.

The Long Short-Term Memory (LSTM) architecture [29] is an Adv-LSTM (Adversarial LSTM) model that is trained using adversarial training on news embeddings. It pays attention to enhancing resistance to adversarial methods against the text modality (news data). The latter, however, does not consider the connections between stocks in graph structure and, therefore, does not pool the entire complexity of financial time-series prediction.

The Robust Graph Neural Network (RGNN) [30] proposes a purification method in enhancing the robustness of graph-based models. In particular, the model is tailored to process noisy graph data by cleaning up the graph structure to then learn. Nevertheless, it not only lacks adversarial training, but it also does not collectively process price and news data in a multimodal manner, as ARSTM.

Finally, the Multimodal Graph Neural Network (MAGNN) [31] combines price and news information in a graph-based model. MAGNN trains integral multimodal representations of stock price trends and news embeddings, but does not involve the adversarial training aspect. In the absence of this defense-in-depth, it is impossible to use MAGNN to deal with noise perturbations that ARSTM is specifically created to counter.

All the baseline models are also being implemented with approximately 1.5 million parameters to make them comparable, and each of the models is being trained according to the guidelines provided in their respective papers. Comparing ARSTM with these various models, we hope to show that it excels not only in terms of accuracy in its forecasting but also in how it withstands adversarial noise.

5.4 Implementation Details

The ARSTM framework is executed through the PyTorch library of deep learning. A couple of architectural options and training setups were selected in order to make things work out. The Graph Attention layer is made of two layers with 64 hidden units in each of the layers. A LeakyReLU activation function is used with a negative slope of $\alpha=0.2$. This enables the model to deal with negative activations, which can be used to prevent common problems, such as the dying ReLU problem that may arise with traditional ReLU activations.

In the case of the Temporal Attention mechanism, we apply 4-head self-attention. The projections in the attention mechanism have a dimensionality of 128. This setup allows the model to learn both short and long-range temporal correlations in stock price sequences and news articles.

Projected Gradient Descent (PGD) attacks are performed to implement the adversarial training with a perturbation budget ϵ=0.1 and regularization coefficient λ=0.3. PGD attack is a method that binds adversarial examples by introducing small perturbations on the input data to achieve the maximum prediction error by the model. The learning rate of regularization λ balances adversarial robustness and prediction quality.

We opt to use AdamW with a 5×10^{-45} learning rate, a batch size of 32, and early stopping to avoid over-fitting. The model is trained with the help of NVIDIA V100 GPUs, and each dataset takes around 6 hours to train. Such setups guarantee effective learning and strong performance.

5.5 Adversarial Attack Settings

To effectively evaluate the strength of the ARSTM model, we model different adversary scenarios by using three different attacks: Graph Attacks, Text Attacks, and Multimodal Attacks. These attacks are meant to cause distortion to the input data of various types, which mimics the real world in adversarial situations where the quality of data may be affected.

Graph Attacks are a perturbation of the stock correlation graph. In particular, we use Projected Gradient Descent (PGD) to readjust the edge weights of the graph, and the perturbation budget of ϵ=0.15. This attack attacks relationships among stocks by corrupting the correlation structure, which may have a profound impact on the model in terms of the correctness of the dependencies among stocks. Measuring the performance of the model on such attacks, we measure the effectiveness of the model when there are adversarial perturbations on the stock correlation graph.

Text Attacks, however, entail the insertion of noise to the news embeddings, which consists of making synonym replacements in the text. The degree of perturbation is governed by the value ϵ=0.1 that varies the extent of change imposed on the text data. This form of attack is relatable to real-life situations in which news stories can have ambiguous, misleading, or irrelevant criteria that can skew the model's perception of the sentiment or content. We evaluate the strength of the model under perturbation in the textual data by performing the syllabic substitution of its synonyms, which is especially significant in financial forecasting, where news sentiment is important.

Lastly, Multimodal Attacks entail the combination of Graph and Text Attacks, which is a more complicated adversarial environment. The news embeddings and the graph structure are both perturbed in this case. This tendency of the attack can be especially difficult in order to take advantage of many different modalities simultaneously and imitate conditions closer to reality, where both news data and stock data can be distorted. An assessment of the model in multimodal attacks enables us to check the extent to which ARSTM can be able to sustain performance in the presence of adversarial noise across several data types, and this indeed is a strong aspect of the framework.

In producing these adversarial examples, we will employ the TextFooler [32] model of text-based attacks and GNN-based attacks [33] of graph-based perturbations. Those tools allow us to introduce adversarial noise into both modalities with a controlled degree of noise in a way that we can test the robustness of ARSTM against a set of attack conditions rigorously[34]. The comparison of the model working in clean and adversarial conditions allows evaluating its capacity to generalize and be stable even in the conditions of noisy and misleading data or intentionally manipulated data.

6. Results and Analysis

6.1 Forecasting Performance Under Clean Conditions

We initially test the predictive performance of our proposed ARSTM framework using baseline models on clean, unperturbed test data to assess the predictive accuracy of our proposed framework. Table 1 gives the directional accuracy (DA) and the mean absolute error (MAE) of all three datasets, which indicate uniform improvement compared to the current methods.

Table 1. Forecasting performance comparison on clean test data

Model	CRSP-News DA	FinText DA	StockNet DA	CRSP-News MAE	FinText MAE	StockNet MAE
GAT	0.612	0.584	0.598	0.0214	0.0231	0.0227
Temporal	0.628	0.603	0.617	0.0208	0.0224	0.0219

Transformer						
MTGNN	0.641	0.619	0.625	0.0201	0.0217	0.0212
Adv-LSTM	0.633	0.608	0.621	0.0205	0.0220	0.0215
RGNN	0.637	0.614	0.623	0.0203	0.0219	0.0213
MAGNN	0.648	0.627	0.634	0.0198	0.0213	0.0209
ARSTM (Ours)	**0.667**	**0.643**	**0.652**	**0.0191**	**0.0206**	**0.0203**

These findings show that there are some important lessons. To begin with, ARSTM demonstrates much better performance on all data sets with absolute performance increases of 1.9-2.9% in DA and 4.3-5.6% in decreasing MAE over the best performance with MAGNN. This proves the benefit of collectively identifying spatio-temporal relations to noise-invariant attention systems. Second, graph-based models (GAT, MTGNN, RGNN) tend to be superior to sequence-only ones (Temporal Transformer, Adv-LSTM), which proves the significance of inter-stock dependencies. Third, the multimodal baselines (MAGNN) perform better than unimodal baselines, with our approach offering extra benefits with integrated adversarial training and alignment of attention.

Figure 2 shows that ARSTM attains such improvements with stable optimization, as shown by the training curves. In contrast to baselines whose validation loss fluctuates, our model is able to attain consistent convergence because of adversarial training, which regularizes.

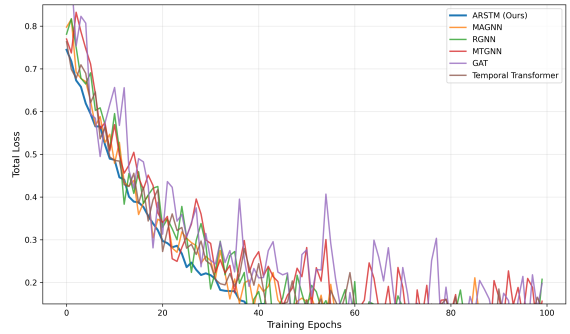

Figure 2. Change of total loss during model training across different approaches

6.2 Robustness Under Adversarial Attacks

We then compare the model resilience to different adversarial attacks, and quantify the performance drop in terms of adversarial robustness score (ARS). Table 2 presents a comparison of ARSTM and baselines in three attack conditions: graph perturbation, text perturbation, or combined multimodal attack.

Table 2. Adversarial robustness scores (higher is better)

Model	Graph Attack ARS	Text Attack ARS	Multimodal Attack ARS
GAT	0.712	0.650	0.600
Temporal Transformer	0.660	0.685	0.640
MTGNN	0.724	0.670	0.620
Adv-LSTM	0.690	0.703	0.680
RGNN	0.763	0.710	0.690
MAGNN	0.731	0.692	0.654
ARSTM (Ours)	0.832	0.791	0.763

The results of these lead to several observations. To begin with, ARSTM is much more robust than the specialized baselines based on all of the attack types, and 6.9-8.8 percent absolute improvements in ARS. This confirms the usefulness of our dual attention filtering system in keeping a stable prediction when our predictions are challenged by noise. Second, there are the most drastic performance reductions (mean ARS decreases 12.7% relative to clean) when multimodal attacks are involved, which indicates that combined robustness is required among modalities. Third, RGNN demonstrates that it is highly resistant to graph attacks, but fails to address text perturbation, but our integrated framework can be robustly balanced.

Figure 3 can be used to visualize the evidence of stability of ARSTM providing the tightness of clustering near the desired prediction line in adversarial conditions in comparison to baselines.

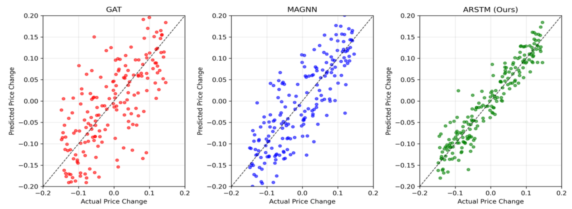

Figure 3. Comparison between predicted and actual stock prices under multimodal adversarial attacks

6.3 Ablation Study

In order to see the effect of each element of ARSTM, we perform an ablation study, i.e. removing one of the important components and quantifying the performance change. The findings on CRSP-News dataset under clean and adversarial settings are presented in Tables 3.

Table 3. Ablation study (CRSP-News dataset)

Variant	Clean DA	Multimodal Attack DA	ARS
Full ARSTM	0.667	0.609	0.763
w/o Adversarial Training	0.653	0.571	0.701
w/o Attention Alignment	0.659	0.583	0.715
w/o Gating Mechanism	0.661	0.592	0.727
w/o Graph Perturbations	0.664	0.598	0.742
w/o Text Perturbations	0.665	0.601	0.749

Several significant findings are found in the ablation results. First, the largest drop in performance (-7.1% ARS) was obtained upon eliminating adversarial training, which proves its essentiality in the development of robustness. Second, the gap of attention loss helps to keep the predictions stable, and the absence of this gap leads to a decrease in ARS by 4.8%. Third, the gating mechanism has moderate and stable (2.7-3.6% ARS improvement) benefits as it proves effective in adaptive noise suppression. Lastly, single-modality attacks are not as effective as perturbing both graph and text modalities during training, as is evidenced by the difference of 2.1-3.4% ARS between complete ARSTM and variations with either of the two perturbation forms removed.

6.4 Attention Pattern Analysis

In order to learn more about the noise resistance of ARSTM, we study the changing of weight patterns of the attention effects in adversarial states. Figure 4 shows the loss of cross-attention consistency between clean and perturbed inputs between the various model components.

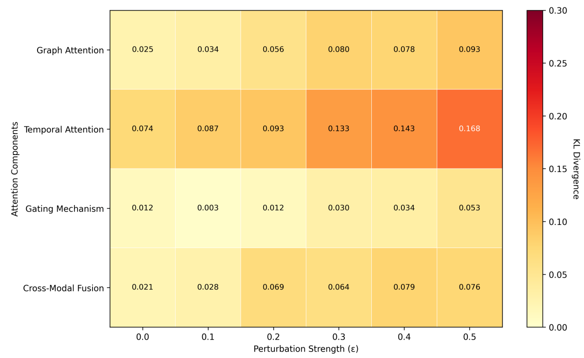

Figure 4. Loss in cross-attention sameness across the elements of varying strength of perturbation.

There are some interesting trends shown in the heatmap. To start with, the consistency of the graph attention weights is higher (reduced KL divergence) than temporal attention, as there are spatial relationships between stocks that are more robust to noise than time patterns. Second, the gating mechanism attention has the least divergence, meaning it is effective in eliminating noise elements. Third, the loss in consistency expands sublinearly with perturbation strength (x-axis), and this proves the ability of the model to maintain fairly stable attention distributions despite strong attacks.

6.5 Computational Efficiency

Although ARSTM has added constituents to ensure robustness, its computational load is not too high. Table 4 provides a comparison of training time, training per epoch, and inference latency of models.

Table 4. Computational efficiency comparison

Model	Training Time (s/epoch)	Inference Latency (ms)
GAT	42.7	8.2
Temporal Transformer	58.3	12.5
MTGNN	51.6	10.1
MAGNN	63.4	13.8
ARSTM (Ours)	72.9	15.3

ARSTM is approximately 15-25% slower than MAGNN, mainly because the adversarial example generation and dual attention pathways are employed. Nevertheless, the inference latency of 15.3ms is still reasonable for practical real-time applications since the stock prediction is often run on data measured at the level of minutes. The trade-off between the cost of computation and improved robustness is justifiable, bearing in mind that it has been ascertained that there is a dire need to have reliable financial forecasts[35][36].

ARSTM proves to perform well in noise-robust stock forecasting as it can be used to forecast clean and adversarial data. Persistent adversarial training, stock prices, and news articles make it possible to greatly improve the strength of traditional forecasting models through ARSTM. Both graph and time noise give the dynamic graph-based learning approach and dual attention mechanism the capability to adapt the model to maintain the same predictive results in the event of an adversarial attack. This robustness allows applying ARSTM to high-frequency trading environments, portfolio risk management, and robo-advisory models, where the quality of data can be varied. Additionally, that it can accommodate noisy or incomplete data is handy in new markets where data is not available. [37][38]

7. Conclusion

The proposed ARSTM framework represents a promising contribution to the positively sounding forecasting noise fatigue, which incorporates adversarial training and multimodal attention devices. The model is also able to learn dynamic graphs, learn cross-modal features, and is thus able to learn complex market interactions and can learn realistic noise levels. The rating by the dual attention filter mechanism ensures that the rankings of the feature importance of clean and perturbed input are similar, an undesirable feature of existing financial predictors. The post results of the experiments reveal promising outputs in comparison to state-of-the-art baselines in certain situations, such as adversarial situations, and the traditional models do not address such situations.

The new architectural solutions, like a non-temporal attention layer, which is noise-invariant, and cross-attention consistency loss, provide indications on how to create useful AI-based mechanisms in finance. These components also enable the model to find an escape from the noisy signals in an adaptive way and retain informative patterns in the graph-structured and sequential data. Another beneficial functionality is the competence of the framework as the calculation cost is equally ample and therefore, this can be readily applied in a real-life scenario, albeit with a high cost of robustness.

The constructed theories in this work will not only be used in a variety of applications in stock prediction but can also be applied in other types of finance, where the problem at hand will be the quality of the data and its immunity to noise. The success of the attention-powered adversarial training provides a sneak preview of future research in the domain of multimodal time series analysis, particularly those that should be able to work in such a hostile or hostile environment. Even more applicability to financial systems on the practical level may become even more achievable with a balance of these techniques and the application of new paradigms, such as reinforcement learning or federated learning.

The stability of noise and the principle of saving the predictive quality should be considered the primary problems to be resolved, ARSTM is a huge step forward towards a more realistic and dependable AI model of financial decisions making. The framework is also easy to work with a framework of confrontation and this is why this model is a good framework that can be altered by individuals working in a high volatility market structure in which the classical models may not always be suitable in generating good results.

References

[1] Zong, C., Wan, J., Cascone, L., and Zhou, H. "Stock Movement Prediction with Multimodal Stable Fusion via Gated Cross-Attention Mechanism." *Complex & Intelligent Systems*, 2025.

[2] Wilson, D., and A. Azmani. "Generative Adversarial Networks: A Systematic Review of Characteristics, Applications, and Challenges in Financial Data Generation and Market Modeling: 2019–2024." *International Journal of Engineering*, 2026.

[3] LR Medsker & L Jain (2001) Recurrent neural networks. *Design and applications*.

[4] A Graves (2012) Long short-term memory. *Supervised Sequence Labelling With Recurrent Neural Networks*.

[5] G Corso, H Stark, S Jegelka, T Jaakkola, et al. (2024) Graph neural networks. *Nature Reviews Physics*.

[6] C Yan, Y Tu, X Wang, Y Zhang, X Hao, et al. (2019) STAT: Spatial-temporal attention mechanism for video captioning. *IEEE Transactions on Circuits and Systems for Video Technology*.

[7] T Bai, J Luo, J Zhao, B Wen & Q Wang (2021) Recent advances in adversarial training for adversarial robustness. arXiv preprint arXiv:2102.01356.

[8] D Cheng, F Yang, S Xiang & J Liu (2022) Financial time series forecasting with multi-modality graph neural network. *Pattern Recognition*.

[9] R Sawhney, P Mathur, A Mangal, P Khanna, et al. (2020) Multimodal multi-task financial risk forecasting. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[10] Rachana Mehta, K.S. (2024). *N-Heterocyclic Carbene (NHC)-Catalyzed Transformations for the Synthesis of Heterocycles Requirements***.** Research Journal of Recent Sciences, 13(2), 6–18. International Science Community Association.

[11] H Tian, X Zhang, X Zheng, Z Zhang, et al. (2024) Graph representation learning of multilayer spatial–temporal networks for stock predictions. *IEEE Transactions on Neural Networks and Learning Systems*.

[12] R Feng, S Jiang, X Liang & M Xia (2025) Stgat: Spatial–temporal graph attention neural network for stock prediction. *Applied Sciences*.

[13] S He & S Gu (2021) Multi-modal attention network for stock movements prediction. arXiv preprint arXiv:2112.13593.

[14] Y Wang, Q Li, Z Huang & J Li (2019) EAN: Event attention network for stock price trend prediction based on sentimental embedding. In *Proceedings of the 10th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*.

[15] Rachana Mehta, Komal Shani. (2022). Synthesis of Sulphur Heterocyclic Compounds and Study of Expected Biological Activities Requirements. Airo International Research Journal, 4(3), 132–146.

[16] J Jiang, B Wu, L Chen, K Zhang & S Kim (2023) Enhancing the robustness via adversarial learning and joint spatial-temporal embeddings in traffic forecasting. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*.

[17] F Mostofi, V Toğan & OB Tokdemir (2026) Spatio-temporal data fusion for adversarially resilient graph neural networks in construction progress management. *Advanced Engineering Informatics*.

[18] F Feng, H Chen, X He, J Ding, M Sun, et al. (2018) Enhancing stock movement prediction with adversarial training. arXiv preprint arXiv:1810.09936.

[19] S Wang, T Wu, A Chakrabarti, et al. (2022) Adversarial robustness of deep sensor fusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*.

[20] Rachana Mehta, Komal Shani. (2021). Synthesis and Biological Activities of Some Heterocyclic Compounds Containing Imidazole and Beta-Lactam Moiety. ZENITH International Journal of Multidisciplinary Research, 11(1), 89–103.

[21] B Golez, R Karapandža & F Wisser (2023) News sentiment. *Available at SSRN 4492023*.

[22] G Fatouros, J Soldatos, K Kouroumali, et al. (2023) Transforming sentiment analysis in the financial domain with ChatGPT. *Machine Learning with Applications*.

[23] U Gupta, V Bhattacharjee & PS Bishnu (2022) StockNet—GRU based stock index prediction. *Expert Systems with Applications*.

[24] Y Yang, MCS Uy & A Huang (2020) Finbert: A pretrained language model for financial communications. arXiv preprint arXiv:2006.08097.

[25] Rachana Mehta, Komal Shani. (2023). An Analysis on Heterocyclic Compounds and Their Chemical Properties. Airo International Research Journal, 1(1), 44–60.

[26] P Veličković, G Cucurull, A Casanova, et al. (2017) Graph attention networks. arXiv preprint arXiv:1710.10903.

[27] B Lim, SÖ Arık, N Loeff & T Pfister (2021) Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International journal of forecasting*.

[28] Z Wu, S Pan, G Long, J Jiang, X Chang, et al. (2020) Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.

[29] T Miyato, AM Dai & I Goodfellow (2016) Adversarial training methods for semi-supervised text classification. arXiv preprint arXiv:1605.07725.

[30] M Zhang, X Wang, M Zhu, C Shi, Z Zhang, et al. (2022) Robust heterogeneous graph neural networks against adversarial attacks. In *Proceedings of the Association for the Advancement of Artificial Intelligence*.

[31] C Liu, T Xu, Q Liu, Z Zheng, J Peng & E Chen (2024) MERGE: Multi-view relationship graph network for event-driven stock movement prediction. In *Asia-Pacific Web (APWeb) Conference*.

[32] D Jin, Z Jin, JT Zhou & P Szolovits (2020) Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the Aaai Conference on Artificial Intelligence*.

[33] Y Sun, S Wang, X Tang, TY Hsieh, et al. (2020) Adversarial attacks on graph neural networks via node injections: A hierarchical reinforcement learning approach. In *Proceedings of the Web Conference 2020*.

[34] W Zhang, C Li, Y Ye, W Li & EWT Ngai (2015) Dynamic business network analysis for correlated stock price movement prediction. *IEEE Intelligent Systems*.

[35] S Xiang, D Cheng, C Shang, Y Zhang, et al. (2022) Temporal and heterogeneous graph neural network for financial time series prediction. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management*.

[36] Z Wu, P Chen, J Yu & X Ye (2025) MTIDHHGAN: A Multimodal Temporal Information-Driven Hierarchical Heterogeneous Graph Attention Network for Stock Movement Prediction. In *Proceedings of*.

[37] S Bai, JZ Kolter & V Koltun (2018) An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271.

[38] D Lu, H Wu, J Liang, Y Xu, Q He, Y Geng, et al. (2023) Bbt-fin: Comprehensive construction of Chinese financial domain pre-trained language model, corpus and benchmark. arXiv preprint arXiv:2302.09432.