A self-supervised approach to anomaly detection in Vietnam stock market

Pham Quoc Trung[1] and Nguyen Duy Anh Quan[2]

[1]VinUniversity, 20trung.pq@vinuni.edu.vn
[2]Vinuniversity, 20quan.nda@vinuni.edu.vn

ABSTRACT

The Vietnamese stock market is a dynamic and crucial component of the global economy, necessitating robust methods for anomaly detection to maintain market integrity and investor confidence. This paper presents a self-supervised approach to anomaly detection in the Vietnamese stock market, addressing the challenges posed by multivariate, non-stationary, and noisy data. Traditional supervised methods are limited by the scarcity of labeled anomalies, prompting the need for unsupervised or semi-supervised techniques.

Our methodology combines statistical and machine learning techniques. Initially, the Mann-Whitney U test is employed to identify calendar anomalies such as the January Effect, Weekend Effect, and Turn-of-the-Month Effect. Building on these findings, we implement a self-supervised forecasting model using a Temporal Convolutional Network (TCN) to predict future stock prices. The TCN's large receptive field allows it to analyze extensive time segments, enhancing prediction accuracy. Deviations between forecasted and actual data are used to pinpoint potential anomalies.

Our results demonstrate the efficacy of this hybrid approach in detecting anomalies within the Vietnamese stock market. The Mann-Whitney U test confirmed significant calendar anomalies, while the TCN-based model accurately identified irregular trading patterns. This framework not only advances the field of anomaly detection in emerging markets but also offers a scalable solution adaptable to other financial systems.

Github: https://github.com/tintinanhquan/data_mining_project.git

1 | Introduction

The financial market, characterized by high-speed transactions and high capital flows, is a crucial component of the global economy. Because of its importance, the detection and analysis of anomalies within stock market transactions become essential to protect the integrity of markets and investor interests, particularly in emerging markets like Vietnam. Market anomalies, that deviate from expected normal market behaviors, could potentially indicate manipulative trading activities or other irregularities that could undermine market stability and investor confidence.

Machine learning has become a powerful tool for anomaly detection in recent years. However, significant challenges remain, particularly in the realm of multivariate anomaly detection. A lack of labeled anomalies limits the power of supervised approaches. Real-world systems often exhibit non-stationary behavior, meaning their statistical characteristics can change over time. Additionally, they are highly context-dependent, requiring anomaly detection models to adapt to current conditions. The data itself can be a challenge: heterogeneous, noisy, and high-dimensional. When anomaly detection serves as a diagnostic tool, interpretability of the model's results becomes crucial for understanding the nature of the anomaly. This complexity necessitates the use of unsupervised or semi-supervised learning approaches in this domain.

Contribution Our project tackles the challenge of identifying unusual patterns within Vietnam's stock market. To achieve this, we combined and enhanced methods from various fields to address the limitations commonly encountered in anomaly detection. Specifically, We first employed the Mann-Whitney U test [8], a statistical approach, to identify potential calendar anomalies in Vietnamese stock data. Recognizing the prevalent seasonal patterns in many stocks, we then implemented a self-supervised forecasting model utilizing a Temporal Convolutional Network (TCN) [1]. TCNs are known for their ability to analyze large time segments due to their extensive receptive field, allowing them to make highly accurate future predictions. Lastly, by comparing the forecasted signal with the original data, we can pinpoint significant deviations, indicating potential anomalies. Through this combined approach, we aim to develop a framework specifically designed for anomaly detection in the Vietnamese stock market, enabling the identification of irregular patterns.

2 | Related Work

Stock market anomaly forecasting remains a critical application of data mining. A Study by Mufasta et al. highlights the presence of patterns and seasonality in international stock markets [4]. Consequently, numerous studies have employed statistical models and sequential deep learning models to capture these time-dependent patterns, aiming to generate reasonably accurate stock price predictions. Building upon these forecasts, other research has explored leveraging the patterns and knowledge encoded within the prediction models to identify anomalies in the stock market.

Islam et al. (2018) investigated the use of LSTMs [10] for anomaly detection in stock trading data [5]. They trained an LSTM model to predict future transaction volume for a 50-day window using historical volume data from US companies implicated in insider trading cases. The mean squared error loss function was used to train the model. Following prediction, they employed Normalized Cross Correlation to identify discrepancies between the predicted and actual volume data, potentially indicating anomalous trading activity.

As an effort to make use of ensemble learning, Wang et al. (2019) introduced a multi-dimensional recurrent neural network (RNN) ensemble learning framework for stock market manipulation detection [9]. Recognizing the multifaceted nature of manipulative activity, they incorporated features from both trading data (liquidity, volatility, returns) and company profiles (market capitalization, leverage ratio). Their hypothesis was that these features would deviate from normal patterns during manipulation attempts. To capture these deviations, they employed an RNN-based classifier for trading features and a supervised model for company characteristics. The final prediction was made by combining these models using bagging and majority voting. This approach yielded a significant improvement (29.8% increase in AUC) compared to conventional machine learning and statistical methods.

3 | Data

3.1 Data Collection

As part of our data collection phase, we have successfully gathered the required fundamental company data and daily time-series data covering approximately 400 stocks from Vietnam's stock market. This data has been sourced from the Vietnam Stock Exchange websites and through APIs provided

by SSI.

- **Time-Series Price Volume Data:** Includes daily data for stocks such as Symbol, Trading Date, Opening Price, Closing Price, Highest and Lowest Daily Price, and Daily Trading Volume.
- **Fundamental Data:** Contains quarterly data like Net Profit After Tax, Price to price-to-earnings ratio, Price-to-Book-Ratio, Number of Shares Outstanding, and Quick Ratio.

3.2 Data pre-processing

Our team has undertaken the following pre-processing steps to prepare the data for analysis:

- **Cleaning:** Addressed missing values, removed duplicates, and corrected data errors, Replace inf values. Missing values were interpolated for stock prices where necessary.
- **Normalization:** Applied standardization techniques to ensure consistency across the data, which facilitates more accurate analysis.

3.3 Feature engineering

Effective feature engineering is crucial for extracting meaningful information from raw data and enhancing the performance of models. In the case of time series forecasting, features are important to capture time-related patterns. Here's a breakdown of our chosen feature:

- Day of the week, Current month: Those are categorical features, indicate the specific period of current time step. Financial markets exhibit cyclical patterns based on weekdays, weeks, and months. Including features like "Day of the Week" and "Month" allows the model to learn these potential seasonalities.
- Lagged Features (ie. Volume of last week Monday): Stock prices often exhibit trends and momentum that persist over time (weekend effect). Lagged features provide targeted historical context for the model.
- Close Price, Volume, High-Low: fundamental price information and gauging market activity

4 | Methodology

4.1 Overview

The methodology employs a two-part approach. The first phase focuses on identifying calendar anomalies commonly observed in stock markets. This includes the January Effect, Weekend Effect, and Turn-of-the-Month Effect. We specifically adapt these established methods to the context of the Vietnamese market.

- **January Effect:** This effect suggests that there is a seasonal anomaly in the financial market where securities' prices increase in the month of January more than in any other month.
- **Weekend Effect:** This phenomenon indicates that stock returns on Mondays are often notably lower than those of the preceding Friday.
- **Turn-of-the-Month Effect:** This refers to the tendency of stock prices to rise on the last trading day of the month and the first three trading days of the next month.

In the second phase, We utilized a forecasting model to capture the underlying patterns within the Vietnamese stock market. This forecasted signal is then used to identify potential anomalies by comparing it to the actual market data. Deviations exceeding a certain threshold can then be flagged as potential irregularities. This section describes the step-by-step approach taken from data preparation through to the analysis phase.

4.2 Statistical Analysis

For our study of calendar anomalies such as the January Effect, Weekend Effect, and Turn-of-the-Month Effect, we focused on calculating returns for specific dates relevant to each anomaly. The analysis involved:

Return Calculations: We calculated the stock returns specifically for dates associated with each calendar anomaly:

- **January Effect:** Returns of small-cap stocks in January compared to other months.
- **Weekend Effect:** Returns from Friday's close to Monday's close.
- **Turn-of-the-Month Effect:** Returns for the last and first three days of each month.

Given the non-normal distribution of our data, traditional parametric tests (like the t-test) were not suitable for our analysis. Therefore, we opted to use:

Mann-Whitney U Test: [8] This non-parametric test was employed to compare the distributions of returns between two independent samples associated with each anomaly. The Mann-Whitney U test is particularly useful in our context as it does not assume a normal distribution of the data, making it ideal for financial data that often exhibits skewness and kurtosis. This test helped us determine if the differences in returns during the anomaly periods were statistically significant compared to other times.

By using the Mann-Whitney U test, we were able to robustly assess the impact of these calendar anomalies on stock returns, providing a reliable statistical foundation for our findings.

4.3 Self-Supervised Approach

Our statistical results, along with research by Mustafa et al. [4], confirm the existence of seasonality in stock markets, including the Vietnamese market. This seasonality reflects a tendency for securities to perform better at certain times compared to others. These timeframes can range from days of the week to months, six-month stretches, or even multi-year cycles. Notably, patterns observed in recent weeks or months can even influence the most recent day's stock prices.

To capitalize on this seasonality, we considered the Temporal Convolutional Network (TCN) architecture, which isspecifically designed to capture patterns within large time segments, and explored its capability in anomaly detection. This section will delve into the details of our approach.

4.3.1 Temporal Convolutional Network

1D-Convolution Convolution is a mathematical operation that merges two functions to produce a third function, reflecting how one function modifies the shape of the other. In the context of 1D sequences, convolution involves a kernel (or filter) that slides over the input sequence. At each position, the kernel performs element-wise multiplication with the corresponding elements of the input sequence and then sums these products to produce a single value. This value is an element of the resulting feature map. The operation can be described mathematically by the following equation:

$$(f * g)(t) = \sum_{i=0}^{k-1} f(i) \cdot g(t-i) \tag{1}$$

where:

- f is the input sequence.
- g is the kernel (or filter).
- k is the length of the kernel.

1D Convolutional Neural Network applies the concept of convolution to extract features from the input sequence. In a 1D convolutional layer, the convolution operations are applied on the whole input sequence, performing element-wise multiplication with the kernels and summing the results to produce a feature map. This process allows the network to capture local patterns and dependencies within the data. By stacking multiple convolutional layers, the network can learn increasingly complex and abstract features, enhancing its ability to recognize patterns and make predictions. Additionally, As more layers are added, the network's receptive field—the range of the input sequence it can consider for each output—expands, allowing it to capture long-range dependencies and contextual information. This large receptive field is crucial for understanding and interpreting the broader context in sequential data.

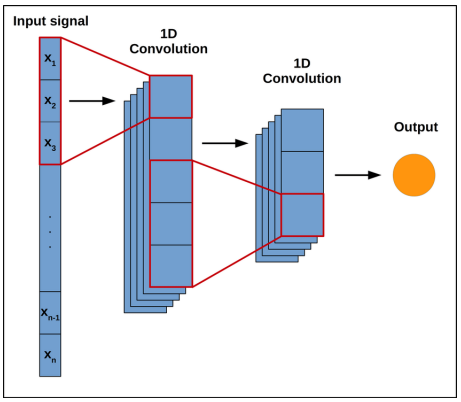

Figure 1: Visualization of 1D CNN [13]

Temporal Convolutional Network (TCN) integrates 1D convolutional neural network architecture with several key enhancements that tailor it for sequential data processing. Central to its design is the concept of causal convolution, where the convolutional filters are restricted to only consider past and current input values, effectively "masking out" future data. Moreover, TCNs leverage dilation, a

technique where the gaps between the elements of the convolutional filters increase exponentially. This dilation mechanism expands the network's receptive field exponentially without increasing the number of parameters excessively. By allowing the model to capture long-range dependencies in the input sequence efficiently, dilation enhances TCNs' capability to extract intricate temporal patterns over extended time spans.

The combination of causal convolution and dilation makes TCNs particularly suitable for tasks requiring the prediction of future states based on historical data, such as our task of predicting stock prices. This architecture not only improves the model's ability to handle temporal sequences with varying lengths but also contributes to its robustness in capturing complex patterns that evolve over time.

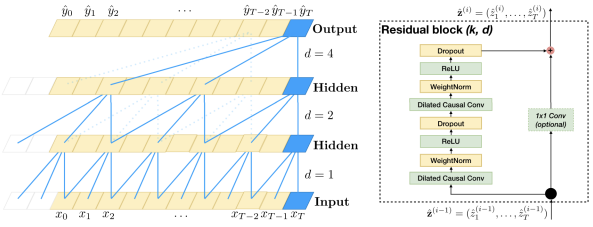

Figure 2: The architecture of TCN [1]

4.3.2 TCN as a forecasting model

Leveraging the TCN's ability to capture long-term dependencies, we repurposed it as a short-range forecasting model. To identify local anomaly patterns within the Vietnamese stock market time series, we designed the TCN to slide across the data. At each segment it processed, we defined a "look-back" window and masked out future information. The features described in Section 3.3 for this look-back window (e.g., the last 30 days) were then fed as input to the model. Based on this historical data, the TCN predicted short-term future stock closing prices (defaulting to 7 days). Finally, to account for overlapping segments, we averaged the predictions for each day to obtain the final results.

4.3.3 Training TCN

To ensure the TCN effectively captures temporal patterns within each stock's entire time series, we split the data for training and testing purposes. For each stock, we employed a 70:30 train-test split, ensuring the order of the data remained intact. This is crucial

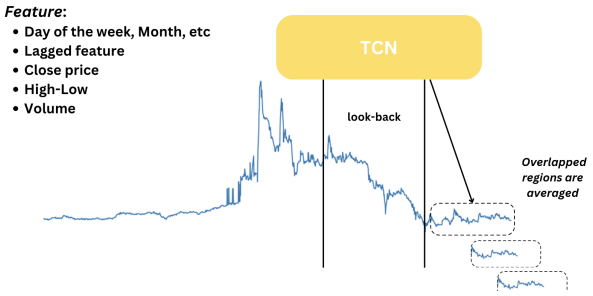

Figure 3: Our TCN forecasting model

for time series data, where the order holds temporal significance. Splitting by date accomplishes this.

During both training and evaluation stages, the closing price plays a dual role: feature and label. For each time step, the model receives the closing price within the look-back window (alongside other features in this window) as input. While the closing price within the prediction range then serves as the target label. The model's performance is evaluated using Mean Squared Error (MSE) as the loss function.

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

where:

- n is the number of data points we predict.
- y_i is the actual close price at time i.
- $\hat{y}_i$ is the predicted close price at time i.

We opted to train and evaluate the TCN independently for each stock in the dataset. While we experimented with pre-training the model on stocks from the same sector, the results weren't as promising as training each stock individually. We'll delve deeper into this in the discussion section.

4.4 Anomaly Detection

Having established a forecasting model, we can leverage the forecasted signal to identify potential anomaly signals. Following established practices, we define anomaly points as those that deviate significantly from the average deviation. To achieve this, we define the residuals as follows:

$$residuals = |actuals - predictions| \tag{3}$$

where:

- **actuals:** the vector of actual closing price values.
- **predictions:** the vector the predicted closing price values.

We establish a threshold of 3 standard deviations away from the mean of the residuals. This threshold captures approximately 99.7% of the data under normal conditions, assuming a normal distribution. Any data point with a residual exceeding this threshold is flagged as a potential anomaly, this condition is defined as follow:

$$|r_i - \mu_r| > 3\sigma_r \tag{4}$$

- r_i represents the residual for the i-th observation.
- μ_r represents the mean of the residuals.
- σ_r represents the standard deviation of the residuals.

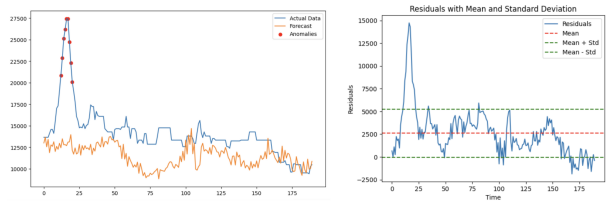

Figure 4: Example of our anomaly detection.
A visualization of anomaly(left), and the residual graph (right)

5 | Experiments

5.1 January Effect Analysis

Histogram Analysis

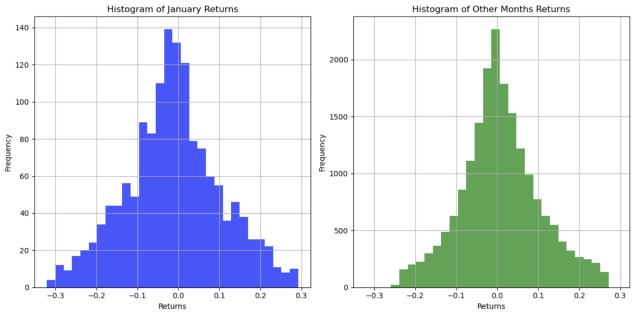

Figure 5: January Effect Return Histogram

The histogram of January returns shows a distribution that appears to be more positively skewed compared to the returns in other months. This visual inspection suggests that January might exhibit unique characteristics in terms of stock returns, potentially supporting the hypothesis of a January Effect in the Vietnamese stock market.

Mann-Whitney U Test Results

To test for the January Effect, we conducted a Mann-Whitney U test comparing the returns of stocks in January to those in other months. The results were as follows:

- Mann-Whitney U Statistic: 12,610,113.5
- P-value: 1.6e-11

These results indicate a highly significant difference in the distribution of returns between January and other months. The extremely low p-value (far below the standard alpha level of 0.05) strongly rejects the null hypothesis that there is no difference in the distributions of returns between January and other months.

5.2 Forecasting

The model's forecasting ability directly influences the sensitivity of anomaly detection. In this section, we will design experiments to evaluate the forecasting capability of the TCN model compared to other baseline models.

5.2.1 Benchmark Model

TCN: We configured a compact Temporal Convolutional Network (TCN) model with the following parameters:

- **Number of Layers:** 4
- **Filter Sizes:** [64, 32, 18, 7]
- **Look-back Segment:** 30 days
- **Prediction Segment:** 7 days
- **Residual Connections:** Enabled

We hypothesize that a 30-day look-back window is sufficient for the TCN model to capture seasonal patterns within a month. These seasonal patterns can be helpful in identifying sudden fluctuations, such as the rapid increase in closing prices observed in the 2022 monthly segments coinciding with the COVID-19 outbreak. To capture additional temporal information beyond, we will introduce lagged features during data preprocessing.

Long Short-Term Memory Network (LSTM) a member of the Recurrent Neural Network (RNN)

family, have been widely applied to sequential tasks, including stock market forecasting [7]. LSTM models excels at processing sequential data and capturing historical patterns within a compressed vector representation, making them well-suited for this task. Following current study [12] [6], We implemented a basic 2 stacked LSTM model following established practices

- **Architecture:** 2-Stacked Unidirectional
- **Hidden Unit:** 50
- **Inference Layer:** 2-layer MLP

Exponential Smoothing (ES) [2] applies weighted averages of past observations to forecast future values, where the weights decrease exponentially as observations move further back in time. This property allows the model to give more importance to recent data, assuming that more recent behavior is a better predictor of the future.

The Exponential Smoothing model that we are using as benchmark is Holt-Winters Seasonal Model [11]. It incorporates seasonal variations in addition to level and trend. It uses three smoothing constants: α, β (trend), and γ (seasonality), making it ideal for data with patterns that repeat over a fixed period.

5.2.2 Experiment setup

In the experiment, We utilize a time series dataset encompassing over 400,000 data points from over 200 individual stocks from the Vietnam stock market. As mentioned, for each of the stock we will train and evaluate independently in a 70:30 ratio on train and test. The result will be the average result on the dataset

For metric use, we utilized the MSE, which is used in the training phase, and mean average percentage error [3], which is widely used for regression and forecasting tasks. It is defined as follows:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \times 100\% \qquad (5)$$

where:

- A_t represents the actual value for observation t.
- y_i represents the forecasted or predicted value for observation t.
- n is the total number of observations.

Table 1: Comparison of TCN and LSTM

Model	Average MAPE	Average MSE
TCN (ours)	~0.025	~0.007
LSTM	~0.038	~0.01
ES	~0.08	~0.13

5.2.3 Result

Table 1 indicates that the TCN model exhibits a slight edge over the other benchmarks in terms of both average MAPE and MSE. This finding supports our initial hypothesis that the TCN's architecture, specifically its ability to capture seasonality and long-term dependencies within the data, might be better suited for stock market forecasting.

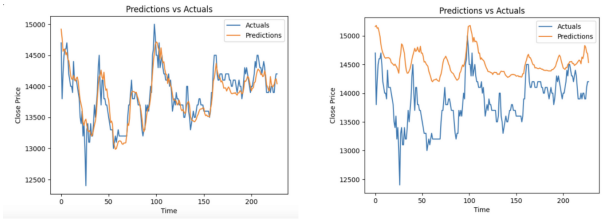

Figure 6: Comparation between TCN (left) and LSTM (right)

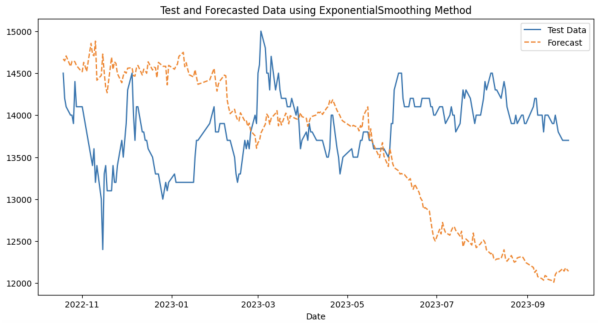

Figure 7: Example of forecasting result using Exponential Smoothing

6 | Discussion

Our findings suggest that using a TCN model for anomaly detection within stock market forecasting holds significant promise for further research. Notably, despite being a deep learning architecture, our compact 4-layer TCN demonstrated efficient training and usability. This efficiency raises the question of whether pre-training the model could enhance its effectiveness.

To explore this, we experimented with pre-training on a group of 18 banking sector stocks, hypothesizing that stocks within the same sector might exhibit similar characteristics. We trained the 4-layer TCN model with identical settings (as described in the experiment section) on the time series data of these 18 stocks (approximately 20,000 entries). We then evaluated the model's performance on a separate set of 3 banking stocks from the same market. Disappointingly, the pre-trained model's MAPE significantly increased compared to the individually trained models (0.04% to 0.15% average MAPE). This suggests that while common sector traits might exist, company-specific performance overshadows these shared characteristics.

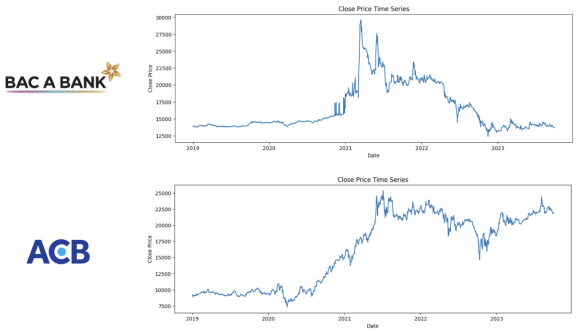

Figure 8: Difference in performance between Bac A Bank (upper) and ACBank (lower)

7 | Conclusion

This study, by utilizing statistical methods and a self-supervised technique, has presented an alternative approach to anomaly detection within the Vietnamese stock market. Firstly, by using the Mann-Whitney U test, we identified calendar anomalies which laid the groundwork for deeper investigation into irregular trading patterns. For the next step, by integrating a self-supervised Temporal Convolutional Network (TCN) model, we are able to effectively analyze stock data trends over and detect anomalies.

From the results above, we can observe the significance of using advanced predictive models in financial markets where anomalies can often go undetected due to less sophisticated market surveillance technologies. The TCN model, in particular, demonstrated superior performance in forecasting and anomaly detection compared to traditional methods such as Exponential Smoothing and LSTM models. This superiority is attributed to the TCN's efficient handling of sequential data and its ability to learn from unlabelled data.

Further research could explore the integration of real-time data feeds into the TCN model to enhance its predictive accuracy and adaptability. Additionally, we could expand the model's application to other sectors within the Vietnamese market to provide a more comprehensive understanding of the Vietnam stock market.

8 | Contribution

Pham Quoc Trung: Research, model implementation, experiment setup.

Nguyen Duy Anh Quan: Research, statistical Tests, experiment setup

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks, 2017.

[2] Robert G. Brown. Exponential smoothing for predicting demand. *Journal of Operations Management*, 4(3):103–119, 1983.

[3] Arnaud De Myttenaere, Boris Golden, Bénédicte Le Grand, and Fabrice Rossi. Mean absolute percentage error for regression models. *arXiv preprint arXiv:1605.02541*, 2016.

[4] Mustafa N. Gultekin and N. Bulent Gultekin. Stock market seasonality: International evidence. *Journal of Financial and Quantitative Analysis*, 18(1):147–182, 1983.

[5] Sheikh Rabiul Islam, Sheikh Ghafoor, and William Eberle. Mining illegal insider trading of stocks: A proactive approach. *ResearchGate*, 2018.

[6] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, and Puneet Agarwal. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. *arXiv preprint arXiv:1802.04431*, 2018.

[7] Sidra Mehtab, Jaydip Sen, and Abhishek Dutta. Stock price prediction using machine learning and lstm-based deep learning models. *arXiv preprint arXiv:2009.10819*, 2020.

[8] Nadia Nachar. The mann-whitney u: A test for assessing whether two independent samples come from the same distribution. *Tutorials in Quantitative Methods for Psychology*, 4(1):13–20, 2008.

[9] Qili Wang, Wei Xu, Xinting Huang, and Kunlin Yang. Enhancing intraday stock price manipulation detection by leveraging recurrent neural networks with ensemble learning. *Neurocomputing*, 2019.

[10] Peter R. Winters. Forecasting sales by exponentially weighted moving averages. *Management Science*, 6(3):324–342, 1960.

[11] Peter R. Winters. Forecasting sales by exponentially weighted moving averages. *Management Science*, 6(3):324–342, 1960.

[12] Frank Xiao. Time series forecasting with stacked long short-term memory networks, 2020. arXiv:2011.00697 [cs.LG].

[13] Bing Yan, Yicheng Wang, Yijun Lai, and Haifeng Fu. The impact of social media influencer endorsements on consumers' purchase intentions: The moderating roles of self-construal and product type. *Computers in Human Behavior*, 117:106613, 2021.