Stock price prediction with global incidents using Long Short-Term Memory network

Polawat Srichana*[1] and Benchawan Wiwatanapataphee[2]

Centre of Excellence in Mathematics, Bangkok 10400, Thailand [1]
e-mail: polawat.sri@student.mahidol.edu[1],
School of EECMS, Curtin University, WA 6102, Australia[2]
e-mail: b.wiwatanapataphee@curtin.edu.au[2]

Abstract

This research paper presents a decision-support framework for investment in the stock market with unexpected incidents. The data used for the study are energy resource stocks with incidences that affected the Japan exchange group market in 2017-2020. Multiple attributes of the data were considered in the analysis, including stock price volatility and closing price moving averages. The best relevant features were chosen by using feature engineering techniques. Three low-risk stocks with expected high returns, selected using Markowitz Modern Portfolio theory, were used for the prediction. A multivariate prediction model based on the Long Short-Term Memory network was developed to predict the closing prices of these three energy stocks with incidences. The effectiveness of the proposed prediction model was evaluated using a standard measurement metric, the Root Mean Square error. The results indicate that the proposed framework is an effective decision-support tool for day traders.

1 Introduction

For investors or anyone interested in investing in the stock market, it is crucial to know the price or trend of stock prices in advance. Designing and analyzing stock pricing is important and challenging work. It takes more than just an analysis of the stock price to forecast a stock's present value or trend. The global occurrence, stock market rumours, and foreign exchange rates are some additional elements that make it harder for investors. However, with a built-in complexity, these chaotic dynamics made things difficult. It is challenging to forecast future stock market price movement due to the numerous variables and sources of information and the small signal-to-noise ratio.

Nowadays, academics and the industry have a growing interest in stock forecasting. Many studies have been conducted to satisfy investors' needs, including the portfolio selection model for selecting portfolios, the regression-integrated moving average (ARIMA) model, the conventional regression approach, and machine learning models for forecasting trends and time series movements with time. The Markowitz portfolio selection model developed by Harry Markowitz in the 1950s is a widely used approach for constructing diversified investment portfolios that maximize expected return while minimizing risk. Some research works have discussed the potential benefits and limitations of using big data in portfolio optimization based on the Markowitz model and machine learning approaches [1, 2]. Pradeep and Sugumar [3] demonstrated the effectiveness of a hybrid approach combining data envelopment analysis and genetic algorithms with the Markowitz model for portfolio optimization using empirical data.

Recently, there has been a lot of research on developing machine learning (ML) techniques for stock prediction. Advanced developments in this area are based on Deep learning, Ensemble learning, Hybrid models, and Sentiment analysis.

- Deep learning uses neural networks with multiple layers to recognize patterns in data through training.

- Ensemble learning involves training a group of models and combining their predictions to form a stronger overall model.

- Hybrid models combine different algorithms to improve performance. For example, a hybrid model for stock prediction might combine a neural network with a support vector machine or a random forest with a boosting algorithm.

- Sentiment analysis uses natural language processing and machine learning algorithms to identify and extract subjective information from natural disaster news and major financial news.

Several machine-learning techniques, including Artificial neural networks (ANNs), Support vector machines (SVMs), Random forests and Boosting, have been applied for stock prediction under incidents. The ANNs, inspired by the human brain's structure and function, can predict stock prices by learning to recognize patterns in the data. The SVMs and Random forests are used for classification and regression tasks. The SVMs work by finding the hyperplane in a high-dimensional space that maximally separates different classes or predicts continuous values, while Random forests work by constructing a large number of decision trees and aggregating the predictions made by each tree. Boosting algorithms are also used for classification tasks. They involve training a sequence of weak learners (models that are only slightly better than random guessing) and combining their predictions to form a strong overall model.

A number of prediction models based on the ANNs have been proposed for stock price prediction [4, 5, 6, 7, 8, 9, 10]. It has been confirmed that a combination of sentiment analysis and machine learning techniques can improve the accuracy of stock price predictions [6, 7]. Marmaris & Hatzichristos [4] found that ANNs can provide accurate predictions, but their performance depends on the quality of the input data and the network architecture. Smith [5] used a combination of financial statement data and macroeconomic variables to improve the accuracy of stock price predictions. Chen & Huang [8] proposed a hybrid model that combines the improved grey wolf optimizer and extreme learning machine for stock price prediction and a hybrid model that combines particle swarm optimization and support vector machines for stock price prediction. The authors found that the hybrid model outperformed both individual models. Kamalakannan & Thangavelu [9] predicted stock prices using different ML algorithms with news articles as input and found that a long/short-term memory (LSTM) network performed the best, with an accuracy of around 85%. Sheridan & Paul [10] used various boosting algorithms combined with sentiment analysis for stock price prediction and found that an XGBoost algorithm outperformed other methods. These existing Researches have shown that machine learning algorithms and sentiment analysis can effectively predict stock prices. However, the accuracy of the predictions depends on the quality of the input data and the specific method used.

This paper presents a decision-support framework for investment in the stock market with unexpected incidents. The primary tasks are the data selection using the Markowitz Portfolio Selection Model and the stock price prediction with incidents using the LSTM networks. The rest of the paper is organized as follows. Section 2 describes the methodology. Section 3 concerns the LSTM-stock price prediction with incidents. Results and conclusion and discussion. are presented in Sections 4 and 5.

2 Methodology

The following important processing steps are typically involved in stock prediction under market incidents.

- **Data collection**: The first step in stock prediction is typically to collect data on the relevant market and economic variables that may influence stock prices. This may include historical stock prices, financial statements, news articles, social media posts, and other data types.

- **Data preprocessing**: After the data has been collected, it may need to be cleaned and preprocessed to make it suitable for analysis. This may involve removing missing or corrupted data, correcting errors, and normalizing the data.

- **Feature engineering**: In this step, the relevant features that will be used to train the prediction model are selected and extracted from the data. These features may include technical indicators, sentiment scores, and other derived variables that are thought to predict stock prices.

- **Model selection and training**: The next step is to select an appropriate machine learning algorithm and train it on the data using the selected features. This may involve adjusting the hyperparameters of the model to optimize its performance.

- **Evaluation**: After the model has been trained, it should be evaluated to assess its accuracy and to identify any areas where it may be lacking. This may involve techniques such as cross-validation, where the model is tested on a portion of the data that was not trained on.

- **Deployment**: If the model performs well in evaluation, it can be used in stock prediction. This may involve integrating the model into a trading system or using it to generate predictions on demand.

- **Monitoring and updating**: After deployment, it is important to monitor the model's performance and update it as needed to ensure that it continues to perform well. This may involve retraining the model on new data or adjusting the features or algorithms used.

2.1 Data Selection and the use of the Markowitz Portfolio Selection Model

The historical data of assets from the energy resources division of the Japanese stock exchange market (JPX), as shown in Table 1 and Figure 1(a), are used in this study. Figure 1(b) presents significant incidents affecting stock prices.

Based on the Markowitz portfolio selection model, an optimal investment portfolio is constructed to maximize expected return while minimizing risk. The model involves the following important steps:

- **Estimating expected returns and variances**: The first step in using the Markowitz model is to estimate the expected returns and variances of the assets that will be included in the portfolio. These estimates can be based on historical data or other factors, such as the risk and return characteristics of the assets.

- **Specifying the desired level of risk**: The next step is to specify the desired level of portfolio risk, which determines acceptable variance. This can be done by specifying a target value for the portfolio variance or by using a risk tolerance measure such as the Sharpe ratio.

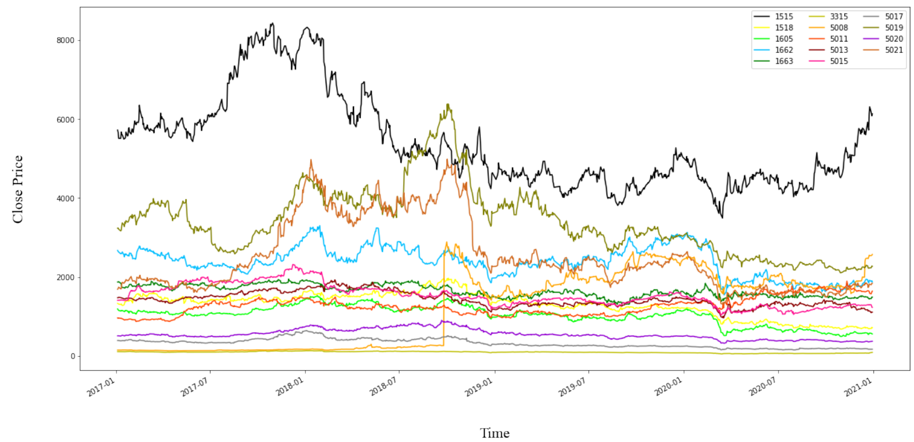
- **Solving the optimization problem**: Once the expected returns and variances of the assets and the desired level of risk have been specified, the optimization problem can be solved using a mathematical algorithm to find the optimal weights of the assets in the portfolio.

- **Implementing the portfolio**: After the optimal portfolio has been determined, it must be implemented by purchasing the appropriate shares of the assets in the portfolio.

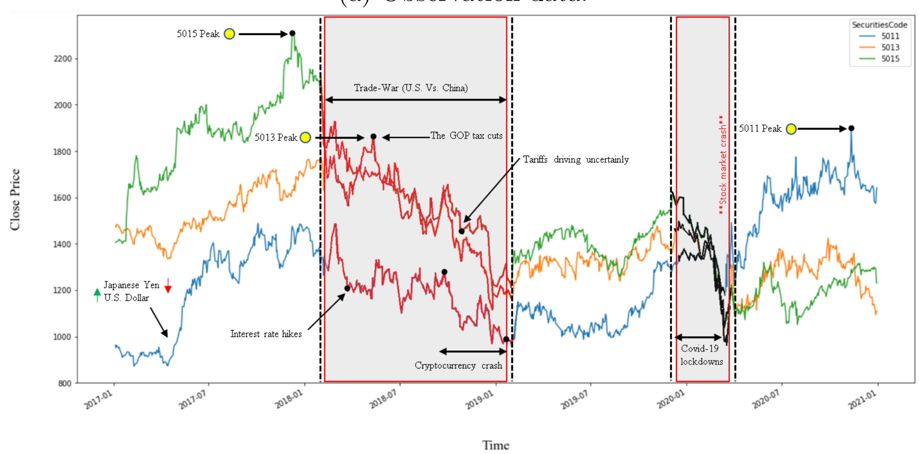
- **Monitoring and rebalancing**: After the portfolio has been implemented, it is important to monitor its performance and to rebalance it as needed to maintain the desired level of risk. This may involve buying or selling assets to adjust the weights in the portfolio.

Table 1: Japanese stock exchange market (JPX): Energy resources

Stock code	Description
1515	Nittetsu Mining CO., Ltd.
1518	Mitsui Matsushima Holdings CO., Ltd.
1605	Inpex corporation
1662	Japan Petroleum Exploration CO., Ltd.
1663	Ko energy group inc.
3315	Nippon Coke Engineering Company, limited
5008	Toa Oil Company, Limited
5011	Nichireki CO., Ltd.
5013	Yushiro Chemical Industry CO., Ltd.
5015	Bp Castrol K.K.
5017	Fuji oil company,ltd.
5019	Idemitsu kosan CO., Ltd.
5020	Eneos Holdings, Inc.
5021	Cosmo energy holdings company, limited



(a) Observation data.



(b) Selected data with incidents

Figure 1: Observation data and selected data with incidents

2.1.1 Expected Returns and Variances of Assets

Several factors can affect the appropriate method for calculating the expected returns and variances of assets. One factor that can influence the choice of method is data availability. For example, if historical data on asset returns is readily available, it may be practical to use that data to estimate the expected return and variance. Other methods may need to be used if the data is unavailable or limited. The type of asset can also influence the choice of method. For example, stocks and bonds may have different characteristics and require different methods for estimating expected returns and variances. The time horizon over which the expected return and variance are estimated can also be a factor. For example, if the asset is being evaluated for a long-term investment, a method that considers long-term trends and risk factors may be more appropriate [11]. Moreover, the complexity of the asset can also affect the choice of method. For example, assets with complex structures or underlying risks may be more difficult to model using simple statistical techniques and require more sophisticated methods such as Monte Carlo simulation [12].

In this research, historical data on the returns of the asset is readily available. Using that data to estimate the expected return and variance may be practical. The return of an asset for a given time period is typically calculated as the change in the price of the asset over that period, expressed as a percentage [13]. For example, if the price of the asset at the beginning of the period is P_0 and the price at the end of the period is P_n, the return, R, over the period is:

$$R = (P_n - P_0)/P_0. \tag{1}$$

Once you have calculated the returns of the asset for each time period in the data set, you can use those returns to estimate the variance of the asset. The variance is a measure of the dispersion or volatility of the returns, and it is calculated as the average squared deviation of the returns from the mean return. The variance of an asset i is given by

$$Var(R_i) = \frac{1}{T-1} \sum_{t=1}^{T} \left(R_t - E(R)\right)^2, \tag{2}$$

where T is the number of returns in the data set and $E(R)$ is the average return over the period. The variance of a portfolio is the weighted sum of individual asset variances, considering the covariance between each pair of assets. Thus, the variance of a portfolio with m assets is determined by

$$Var(R_p) = \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} \sigma_i \sigma_j w_i w_j, \tag{3}$$

where n is the number of assets, ρ_{ij} is the covariance between the returns of the asset i and the asset j, and σ_i and w_i denote the standard deviation of the returns, $\sigma_i = \sqrt{Var(R_i)}$, and the weight of the asset i, respectively. The resulting variance estimate (2) gives a measure of the volatility of the asset's returns over the period of time for which data is available.

2.2 Specifying the desired level of risk

In the Markovic portfolio selection model, the level of risk in the portfolio is determined by the variance or standard deviation of the portfolio [14].

The lower the variance or standard deviation, the less risky the portfolio is considered to be[15]. To specify the desired level of risk in the Markovic model, we set a constraint on the variance of the portfolio.

2.2.1 Markowitz Portfolio Selection Model

Based on the assumption that investors are risk-averse and seek to minimize risk, the objective function of the Markovic portfolio selection model to maximize the expected return of the portfolio while minimizing risk can be expressed mathematically as follows:

$$\min f(\mathbf{w}) = \min \sum_{\mathbf{i=1}}^{\mathbf{n}} \sum_{\mathbf{j=1}}^{\mathbf{n}} \rho_{\mathbf{ij}} \sigma_{\mathbf{i}} \sigma_{\mathbf{j}} \mathbf{w_i} \mathbf{w_j} \tag{4}$$

subject to constraints $h_i(\mathbf{w}),\ (i = 1, \ldots, m)$:

- Expected return of the portfolio is greater than or equal to required return, μ_e, i.e.,
$$\mu_e - \mathbf{w}^T \mu \leq 0 \tag{5}$$
- Minimum weight is less than or equal to the weight of each asset that is less than or equal to maximum weight, i.e.,
$$w_i - \max\{w_i\}_{i=1}^{n} \leq 0 \tag{6}$$
- the standard deviation of the portfolio should be no more than σ_e, i.e.,
$$\sigma_i - \sigma_e \leq 0. \tag{7}$$
- the sum of weights is 1, i.e.,
$$\sum_{i=1}^{n} w_i - 1 = 0. \tag{8}$$
and
$$w_i \geq 0\ (i = 1, \ldots, m). \tag{9}$$

The first constraint ensures that the portfolio meets a minimum required return, while the second constraint specifies any limits on the weights of the assets in the portfolio. The third constraint help to guide the optimization process and ensure that the portfolio is within an acceptable level of risk. The last constraint ensures that the portfolio is fully invested (i.e., the weights of all assets sum to 1).

In this study, the interior Point method was applied using the barrier function $\phi : \Omega \to R$:

$$\phi(\mathbf{w}) = -\sum_{i=1}^{m} \log(-h_i(\mathbf{w})) \tag{10}$$

to obtain the modified objective function

$$\min_{\mathbf{w}} \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij}\sigma_i\sigma_j w_i w_j + \frac{1}{t}\phi(\mathbf{w}). \tag{11}$$

2.2.2 Solution of Markowitz Portfolio Selection Model

The objective function (11) was solved by **Algorithm 1** to find the weights of the assets that minimize variance and meet the constraints. In this study, σ_e = 0.3162 is equivalent to a variance of 0.1). The resulting weights represent the optimal allocation of the portfolio among the assets.

Figures 2 and 3 present profiles of the efficient frontier in each year between 2017 and 2020, in which successful optimization of the return versus risk paradigm places a portfolio along the efficient frontier line.

Algorithm 1: Interior Point Algorithm

Input : Set initial solution w_0, initial parameter t, tolerance TOL
Output: Processed list ws, fs, gap.

```
/* initialize arrays                                          */
w, ws, fs = w0, [ w ], [ f(w) ]
/* duality gap                                                */
gap = [ m/t ]
/* setup parameters                                           */
k = 0, maxI = 1000
/* loop until the stopping criterion is met                   */
while (m / t > TOL) do
    i=0
    d = np.array([[1], [1]])
    while norm(d) > TOL and i < maxI do
        /* Gradient of f(w) + 1/t * φ(w)                      */
        gw = g(w, t)
        /* Hessian of f(w) + 1/t * φ(w)                       */
        Hw = H(w, t)
        d = -inv(Hw) * gw
        w = w + d
        ws.append(w)
        i += 1
    end while
    t = (1 + 1/(13 * √nu)) * t              // update parameter t
    /* update solution                                        */
    gap.append(m/t)
    fs.append(f(w))
    k += 1
end while
ws = np.array(ws)
return ws, fs, gap
```

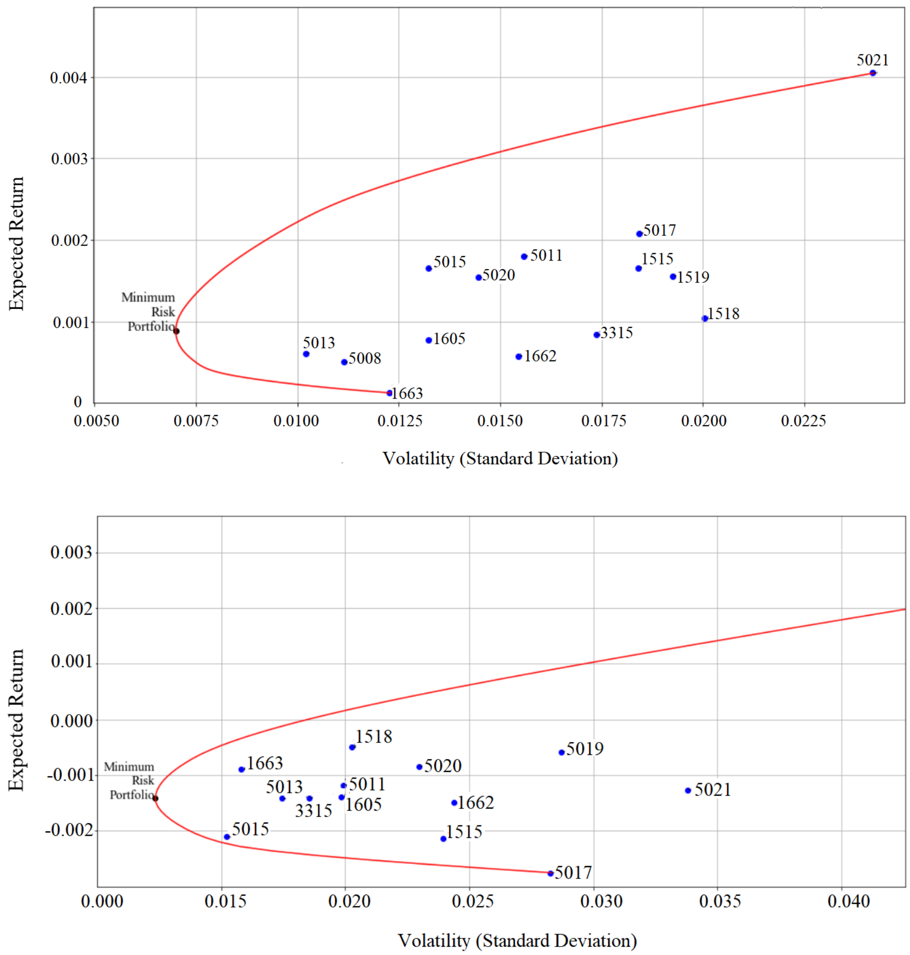
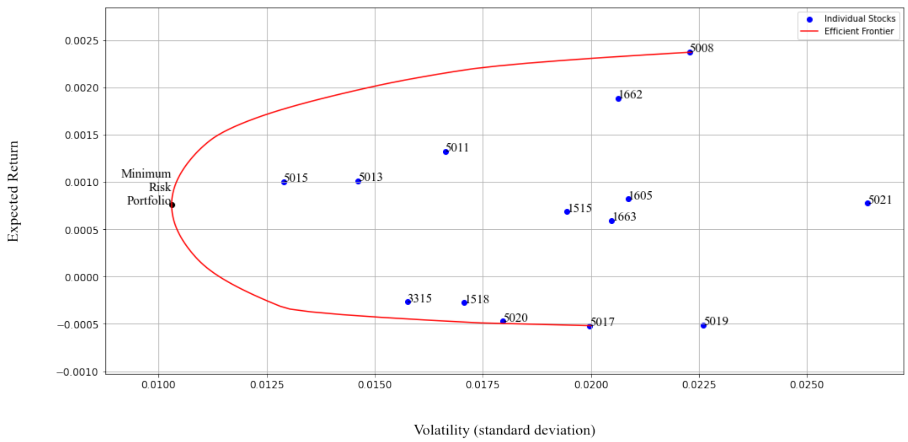
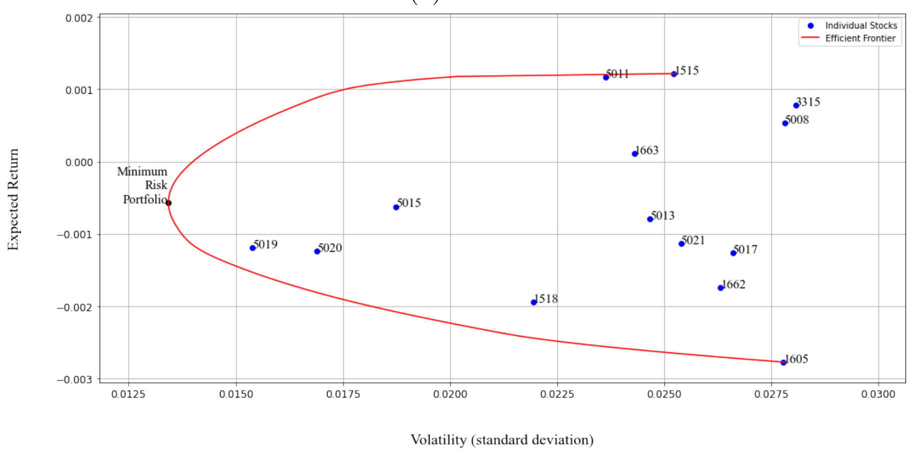


Figure 2: Profiles of efficient Frontier with various optimal portfolios (blue dots) in 2017 (first row) and 2018 (second row).



(a) 2019



(b) 2020

Figure 3: Profiles of efficient Frontier with various optimal portfolios (blue dots) in 2019 (first row) and 2020 (second row).

The minimum risk portfolio is shown in Table 2 and Table 3. It is noted that the 5021-, 5019-, and 5008-stock assets have extremely high volatility, making them more susceptible to long-term profit-taking. These stocks have high expected returns in the years 2019 and 2020. Considering long or high-risk investments with high returns, it indicates that the stocks with the second-highest risk are 5020, 1518, 5011 and 5017, and the 5020-stock asset has the lowest risk and highest return in terms of second-highest risk during this period. Additionally, the 5011-stock asset is relatively close to the minimum risk point, which can also be considered.

For now, we only consider stocks that, despite their modest profits and having an expected return of greater than zero, give the result as 5015, 5013 and 5011. However, it has a low risk and volatility, making it suitable for inclusion in the portfolio for long-term success. So they have been employed to forecast.

Table 2: Minimum Risk Portfolio in years 2017 and 2018.

2017		**2018**	
Stock code	**Expected Return**	**Stock code**	**Expected Return**
1515	1.19660e-02	1515	8.42000e-06
1518	2.83370e-02	1518	6.39000e-02
1605	4.98390e-02	1605	1.06000e-01
1662	1.10000e-05	1662	1.89000e-06
1663	1.33713e-01	1663	2.59000e-01
3315	4.16210e-02	3315	3.46000e-02
5008	2.38881e-01	5008	7.14000e-05
5011	1.13716e-01	5011	7.25000e-02
5013	2.00836e-01	5013	1.54000e-01
5015	1.31525e-01	5015	3.10000e-01
5017	9.65500e-03	5017	6.87000e-07
5019	5.10000e-04	5019	4.06000e-07
5020	3.93890e-02	5020	1.82000e-06
5021	2.00000e-06	5021	1.55000e-07
Volatility	7.00900e-03	**Volatility**	1.23000e-02
Expected Return	8.84000e-04	**Expected Return**	-1.42000e-03

Table 3: Minimum Risk Portfolio in years 2019 and 2020

2019		**2020**	
Stock Code	**Expected Return**	**Stock Code**	**Expected Return**
1515	4.68680E-02	1515	2.088323e-02
1518	1.48245E-01	1518	1.309240e-07
1605	1.06140E-02	1605	3.948939e-08
1662	2.58000E-04	1662	4.931597e-08
1663	9.00000E-06	1663	9.742284e-02
3315	1.44574E-01	3315	5.794596e-02
5008	9.57860E-02	5008	5.410598e-02
5011	7.23400E-02	5011	7.315603e-02
5013	1.02188E-01	5013	1.597349e-04
5015	3.74137E-01	5015	1.339377e-01
5017	4.00000E-06	5017	3.741674e-08
5019	2.00000E-06	5019	4.255557e-01
5020	4.97200E-03	5020	1.368325e-01
5021	2.00000E-06	5021	3.027491e-08
Volatility	1.03010E-02	**Volatility**	1.342969e-02
Expected Return	7.62000E-04	**Expected Return**	-5.663481e-04

It is noted that many stocks in 2018 are moving higher profit from the previous year, thus making the stocks to choose from during this period have to be the same stocks selected in the previous period. The stocks with expected returns above zero are 5015, 5013 and 5011. The 5011-Nichireki stock was more profitable during the previous period, but had higher volatility. On the other hand, stocks 5013 and 5015 remained. Close to the same position from the previous period, including a slight increase in Volatility. It is, therefore, appropriate to take into account trading and prediction. From the description, it is possible to select three stocks, namely 5011, 5013 and 5015, for prediction.

3 LSTM - Stock Price Prediction under Incidents

Long-short-term memory (LSTM) can distinguish between recent and early samples by giving each a distinct weight while erasing memory. It considers useless to forecast the next output; this particular type of recurrent network has demonstrated excellent performance on several issues. As a result, compared to other recurrent neural networks that can only memorize short sequences, it is better equipped to handle long input sequences.

3.1 Pre-Processing

The Data pre-processing includes the following four stages.

- Incidence is converted to Boolean data, where True means an event that happened and affects the stock market, while False means no event occurred, then transform True and False to 1 and 0, respectively. Merging two datasets of stock prices and financial news using timestamp matching provides input data with n observations and five features. These features are high, low, close, open prices, and incidence, i.e., $\mathbf{X} = (x_i^1, x_i^2, x_i^3, x_i^4, x_i^5)$, $(i = 1, \ldots, n)$.

- Splitting the dataset into two portions with an 80:20 ratio. The training dataset comprises 80% of the data and the test dataset 20%.

- Feature scaling was performed because the scales of the data attributes varied, preventing training with inconsistent units and accounting for the different values in each attribute. Next, the Min-Max scaling technique normalized data to the 0 and 1 range.

$$y_i^c = \frac{x_i^c - \min_c}{\max_c - \min_c}, \quad (i = 1, \ldots, n;\ c = 1, \ldots, 5), \tag{12}$$

  where $\min_c$ and $\max_c$ are the minimum and maximum values in $\{x_i^c\}_{i=1}^n$.

- The dataset is reshaped into a three-dimensional array as the final preprocessing step. In this study, the window sizes of 5, 10, and 30 are chosen for the model to calculate data in 5 days (Short-term prediction), 10 days (Short-term prediction), and 30 days (Long-term prediction).

3.2 Model Architectures

With the use of hyperparameter tuning, the intended model was performed. The Grid search technique was used to find the optimal hyperparameters with a set of initial parameters. The grid search technique indicates that the optimizer should use "Adam" and a dropout rate of 0.2 on all models and the number of neurons, as shown in Figures 4, 5 and 6 in our simulation.

3.2.1 Training-Prediction Process

The model will be trained with a training data set, while the test will occur after that. The model's architecture was created based on the hyperparameters received from the grid-searching algorithm. The training time and mean square errors (MSE) were recorded, and Root mean square errors (RMSE) were computed.

$$\mathrm{MSE} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}; \quad \mathrm{RMSE} = \sqrt{\mathrm{MSE}}, \tag{13}$$

where y_i and $\hat{y}_i$ represent observed and predicted data, respectively, and n is the number of observations. Stock price predictions are given after the model fitting and Root Mean Square errors (RMSE) were computed.

4 Stock Price Prediction

This section demonstrates the performance of LSTM-prediction models with five-class input data and one-class prediction output. The accuracy of the prediction models with different window sizes is investigated for each stock, including 5015-Bp Castrol K.K., 5013-Yushiro Chemical Industry CO. Ltd. and 5011-Nichireki CO., Ltd. using the Root Mean Square errors (RMSEs).

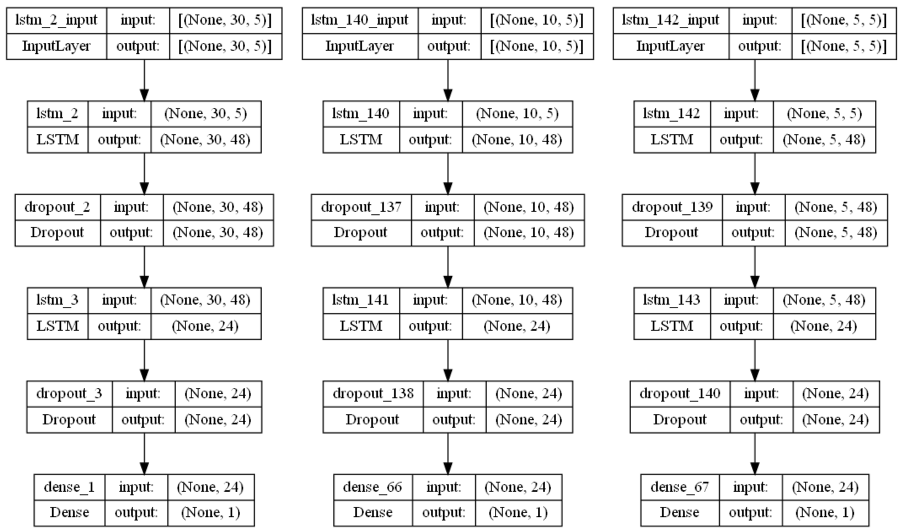


Figure 4: 5015-Model Architectures with a 5-feature input, a closing price output and three window sizes of 30, 10, and 5 days (from left to right).

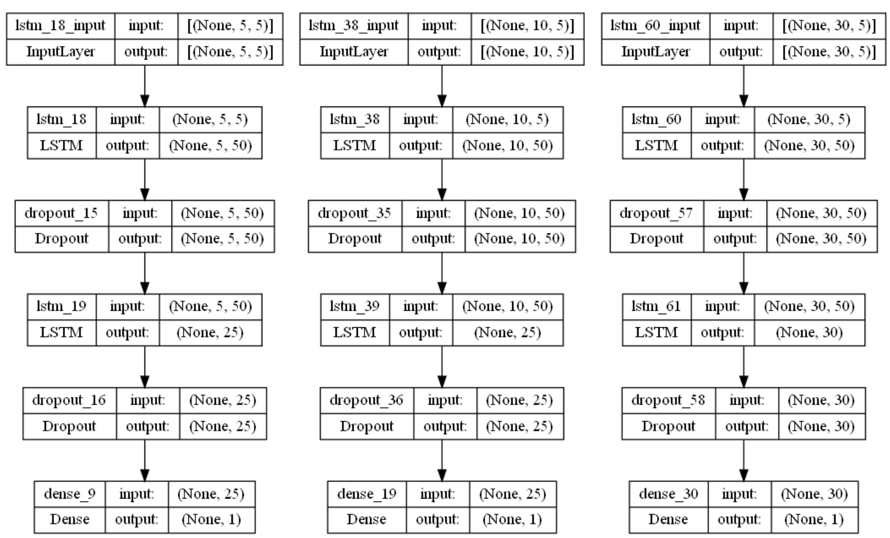


Figure 5: 5013-Model Architectures with a 5-feature input, a closing price output and three window sizes of 30, 10, and 5 days (from left to right).

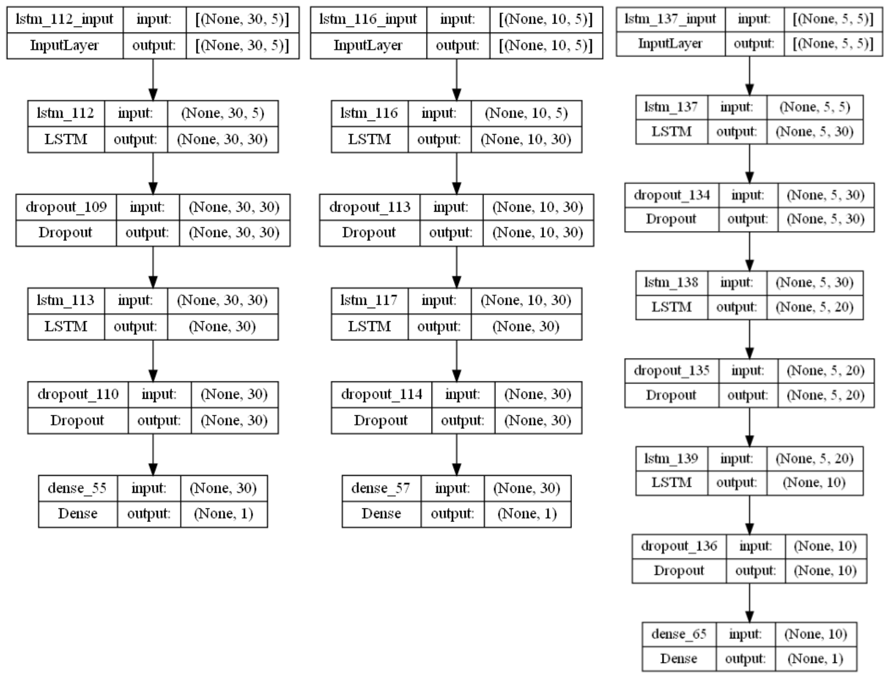


Figure 6: 5011-Model Architectures with a 5-feature input, a closing price output and three window sizes of 30, 10, and 5 days (from left to right).

4.1 Bp Castrol K.K.

Stock prices are moving higher, with the predicted stock price having a similar pattern to the observed stock prices, as shown in Figure 7. It is noted that the long-term prediction (30 days) is less accurate than the short-term prediction, but still has a pattern closer to the actual closing price. For short-term prediction, they are very similar in terms of prediction, only in window size five days it gives a closer to reality than window size ten days. This makes it possible to infer that Bp Castrol K.K. (5015) stocks are suitable for trading and still have an accurate 5-day forecast.

4.2 Yushiro Chemical Industry CO. Ltd.

5013-stock predictions are depicted in Figure 8. Although the stock dropped in the early forecast period, the price remained steady, indicating low volatility. Note that the Long-term prediction (30 days) is less accurate. There is also a large discrepancy in the middle and end of the close price line. Therefore, it is not suitable to use Long-term prediction to predict Yushiro Chemical Industry

CO. Ltd.'s stock (5013). On the other hand, Short-term prediction has higher accuracy, especially for a Window size of five days which is very accurate. It has a pattern that is similar to the actual stock price line. The small jitter is also predicted. Compared to the Window size of 10 days, there is a slight deviation in the middle of the prediction. It can be inferred that the short-term prediction for the Yushiro Chemical Industry CO. Ltd.'s stock (5013) stock is very suitable, especially for predicting stock prices within five days.

4.3 Nichireki CO., Ltd.

For the 5011-Nichireki stock, long-term prediction with extremely high tolerances is presented in Figure 9, although there are some similar patterns, small jitter is not predicted. Also, there was a little discrepancy at the beginning of the prediction, while the actual stock price line soared. Nevertheless, the model predicts a stock price drop ahead of the actual price.

Table 4: RMSEs of the LSTM models with different window sizes.

Setting		Stock Code		
Window size	Dataset	5015	5013	5011
30 days	Train	6.28254	6.19627	15.95085
	Test	6.79850	6.96528	8.34224
10 days	Train	5.36566	6.18760	10.57096
	Test	5.84678	6.38242	8.02197
5 days	Train	3.60345	2.9617	9.4177
	Test	3.65349	3.06127	7.8572

Table 4 shows the mean RMSE of the prediction models obtained in each window size. It is noted that for the 5015- and 5013-stock prices, the RMSEs for the training sets in all window sizes are lower than those for the test sets. This is reasonable, as the larger the data used for training, the better the model learns. In contrast to the 5011-stock price prediction, the RMSEs for the training sets in all window sizes are higher than those for the test sets. This may be due to the high volatility of the training data. The RMSE value of the 5011-stock prediction is extremely high, relative to the predicting patterns with inaccuracies.

Compared to the 5015-stock and 5013-stock short-term predictions, the 5011-stock prediction is less accurate. Its price prediction does not get much of a targeted accuracy value, although, during the five days, there is a close pattern, and a small jitter is predicted in the last phase of the prediction. However, at the beginning of the prediction was a huge discrepancy. This also happens for the 10-day and 30-day predictions. Even testing for another number of neurons or a reduction in the hyperparameter could not make the predictions more accurate. Perhaps, it is because Nichireki CO., Ltd. (5011) shares have higher volatility than Yushiro Chemical Industry CO. Ltd. (5013) and Bp Castrol K.K. (5015), causing the prediction to be misleading based on the volatility.

5 Conclusion and Discussion

This paper presents a decision-support framework for investment in the energy-stock market with incidences that affected the Japan exchange group market in 2017-2020. The decision-support framework includes two primary tasks, portfolio optimization and multivariate stock price prediction. Markowitz Modern Portfolio Theory and the long short-term memory neural network were applied for the portfolio section model and the stock price prediction. The findings are summarized as follows:

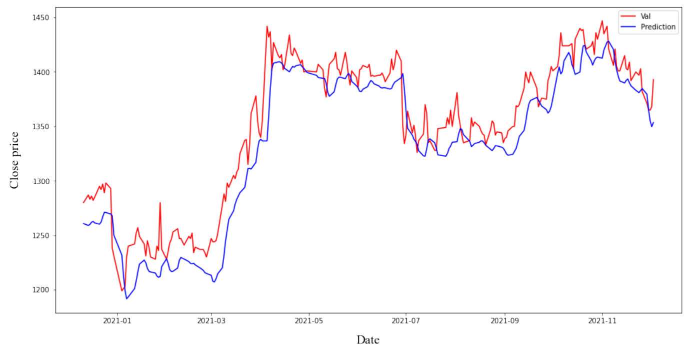
- Markowitz Portfolio Selection Model:

  - Factors that affect the model for calculating the expected returns and variances of assets include data availability, asset type, time horizon, and complexity of the asset. In this study, historical data on the returns of the asset is readily available, we then use that data to estimate the expected return and variance. The time horizon over which the expected return and variance are being estimated can also be a factor. Assets underlying risks may be more difficult to model and may require more sophisticated methods, such as Monte Carlo simulation.

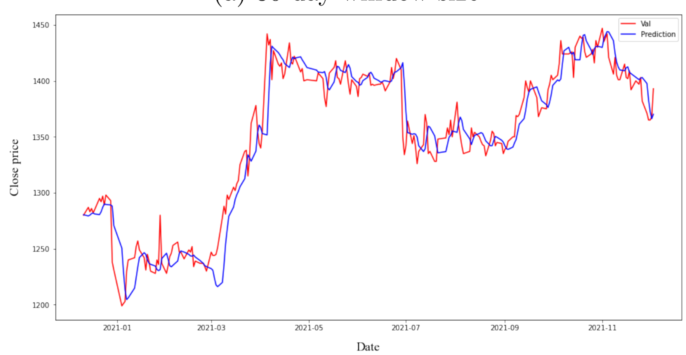
  - The model can help investors to identify portfolios that offer a good balance of risk and return, and can be used to compare different portfolio options in order to make informed investment decisions.

- Multivariate prediction Model: The long short-term memory neural network has been applied to stock prediction and has been found to be effective. Small window size has the potential to improve the accuracy of predictions.

Research has shown that the proposed investment tool, including a portfolio selection model and a multivariate prediction model based on LSTM neural network, is efficient for stock market prediction. However, its performance depends on the quality of the input data and the specific method used.



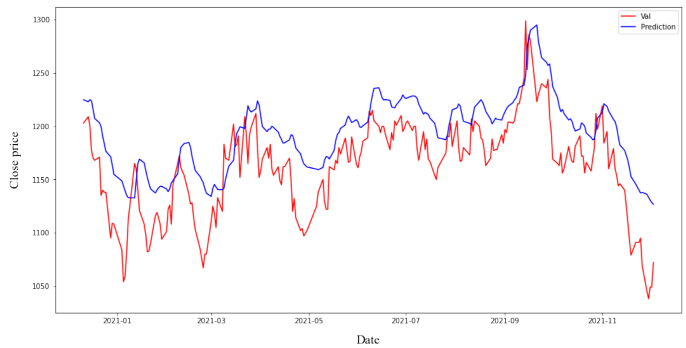
(a) 30-day window size



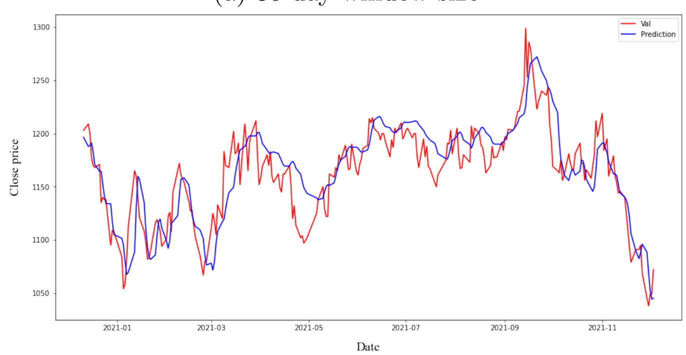
(b) 15-day window size

(c) 5-day window size

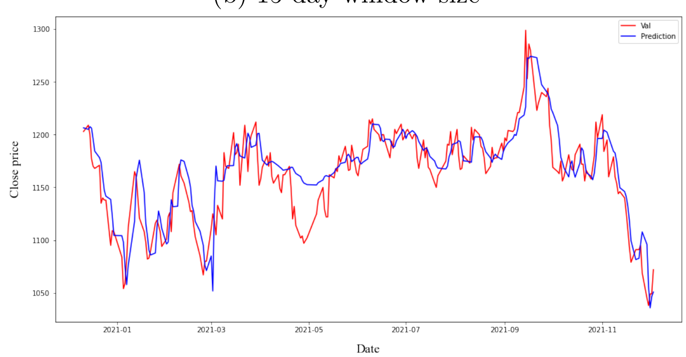
Figure 7: Predicted (blue colour) and observed profiles (red colour) of 5015-Stock prices obtained from the LSTM models with three window sizes: (a) 30 days; (b) 10 days; (c) 5 days.



(a) 30-day window size

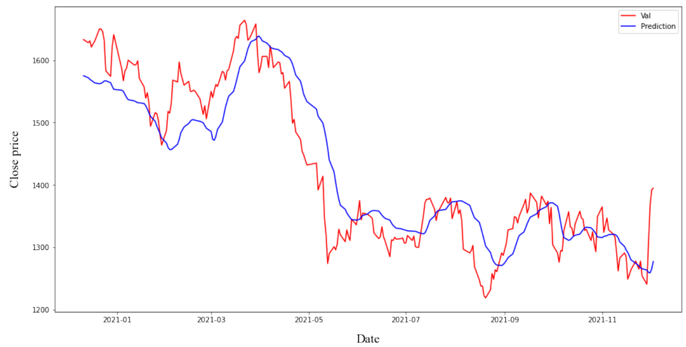


(b) 15-day window size

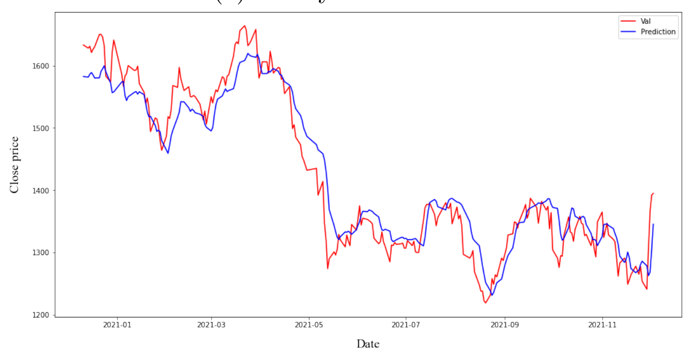


(c) 5-day window size

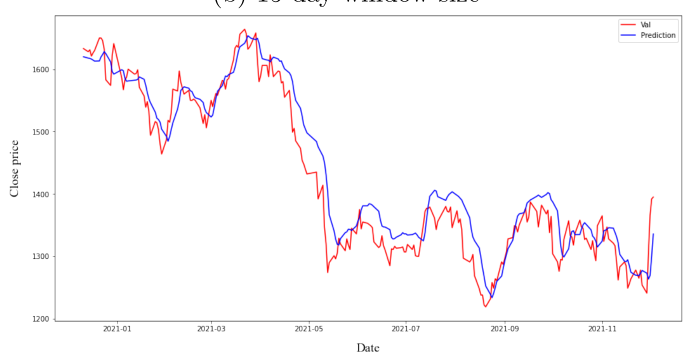
Figure 8: Predicted (blue colour) and observed profiles (red colour) of the 5013-Stock prices obtained from the LSTM models with three window sizes: (a) 30 days; (b) 10 days; (c) 5 days.



(a) 30-day window size



(b) 15-day window size



(c) 5-day window size

Figure 9: Predicted (blue colour) and observed (red colour) profiles of the 5011-Nichireki stock prices obtained from the LSTM models with three window sizes: (a) 30 days; (b) 10 days; (c) 5 days.

References

[1] A. Tsouknidas, V. Stathopoulos, and S. Tziralis, "Data-driven portfolio optimization: A survey," *Expert Systems with Applications*, vol. 134, pp. 109–131, 2019.

[2] K. Kamalakannan and P. Thangavelu, "Big data and portfolio optimization: A review," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 8, no. 4, pp. 93–98, 2018.

[3] J. Pradeep and J. Sugumar, "Portfolio optimization using data envelopment analysis and genetic algorithms," *International Journal of.*

[4] H. Marmanis and D. Hatzichristos, "Stock market prediction using artificial neural networks," *Expert Systems with Applications*, vol. 39, no. 2, pp. 2096–2102, 2012.

[5] J. A. Smith, "Predicting stock prices using financial statements and macroeconomic variables," *Journal of Economics and Business*, vol. 75, pp. 1–9, 2014.

[6] M. B. Alhammadi and S. S. Al-Dharrab, "Predicting stock prices using twitter sentiment analysis," *International Journal of Computer Applications*, vol. 112, no. 16, pp. 18–22, 2015.

[7] J. C. Pradeep and J. A. Sugumar, "Stock price prediction using sentiment analysis and machine learning techniques," *International Journal of Computer Applications*, vol. 168, no. 7, pp. 18–22, 2017.

[8] Y. J. Chen and Y. F. Huang, "Stock price prediction using a hybrid model of improved grey wolf optimizer and extreme learning machine," *Expert Systems with Applications*, vol. 64, pp. 135–144, 2017.

[9] K. Kamalakannan and P. Thangavelu, "Predicting stock prices using deep learning: A comparative study," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 8, no. 6, pp. 23–29, 2018.

[10] T. B. Sheridan and J. F. Paul, "Predicting stock prices with ensemble learning," *International Journal of Information Technology and Computer Science*, vol. 9, no. 6, pp. 1–7, 2018.

[11] D. A. Aaker and B. Mascarenhas, "The need for strategic flexibility," *The Journal of Business Strategy*, vol. 5, no. 2, p. 74, 1984.

[12] P. Brandimarte, *Handbook in Monte Carlo simulation: applications in financial engineering, risk management, and economics.* John Wiley & Sons, 2014.

[13] H. A. Latane and D. L. Tuttle, "Criteria for portfolio building," *The Journal of Finance*, vol. 22, no. 3, p. 359, 1967.

[14] R. Campbell, R. Huisman, and K. Koedijk, "Optimal portfolio selection in a value-at-risk framework," *Journal of Banking & Finance*, vol. 25, no. 9, pp. 1789–1804, 2001.

[15] K. L. Fisher and M. Statman, "The mean–variance-optimization puzzle: Security portfolios and food portfolios," *Financial Analysts Journal*, vol. 53, no. 4, pp. 41–49, 1997.