JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 40, 717-743 (2024)
DOI: 10.6688/JISE.202407_40(4).0003

Stock Trend Prediction Assisted by Automatically Defined Polarity Scores of News Articles

SHUEN-LIN JENG+, CHIAU-HSUAN LIU AND CHIN MIN GUAN
Department of Statistics, Institute of Data Science
Center for Innovative FinTech Business Models
National Cheng Kung University
Tainan City, 701 Taiwan
E-mail: sljeng@mail.ncku.edu.tw+; s11001166@gmail.com; trista.chinmin@gmail.com

This study thoroughly integrated news information into the daily stock price up/down prediction. The predictors included the carefully selected technical features of stock prices, the word level and sentence level sentiment features, and the automatically defined polarity scores of news articles. The main goals of this study are to establish the functional polarity scores of news articles, build models with high prediction accuracy, and identify the important news feature effect for the price up/down prediction. Compared with the Long Short-Term Memory (LSTM) model of Recurrent Neural Networks, we take advantage of Multivariate Adaptive Regression Splines (MARS) as our primary prediction model for its capability in local feature building and its interpretability of selected features. An empirical analysis was carried out on the selected 100 individual stocks in Standard & Poor's 500 from 2019 to 2021. The overall average prediction accuracy for 100 stocks by the proposed MARS model was 0.653 which was significantly higher than 0.505 by the LSTM model.

Keywords: long short-term memory, multivariate adaptive regression splines, news polarity scores, sentiment, stock trend prediction, time series

1. INTRODUCTION

The advancement of financial technology (FinTech) automates various financial services and enhances their efficiency, eventually reducing business operating costs and bringing convenience to the public. One of the significant applications of FinTech is predicting of stock prices to maximize shareholders' profit with the assistance of modern technologies. According to many recent studies, financial news can be utilized to improve the accuracy of stock price predictions by using natural language processing (NLP) technology, a branch of artificial intelligence. Typically, sentiment analysis is performed to extract related information from financial news.

Several novel methods are proposed in this paper to calculate the news polarity scores to manifest the relationship between news and stock price. The specific aims of this study are threefold: (1) Establish the functional polarity scores of news articles; (2) Build high-accuracy models for the price up/down prediction; (3) Identify the important news feature effect for the price up/down predictions. The data processing process of this study can

Received March 31, 2023; revised May 22, 2023; accepted July 8, 2023.
Communicated by Mu-Yen Chen.
+ Corresponding author.

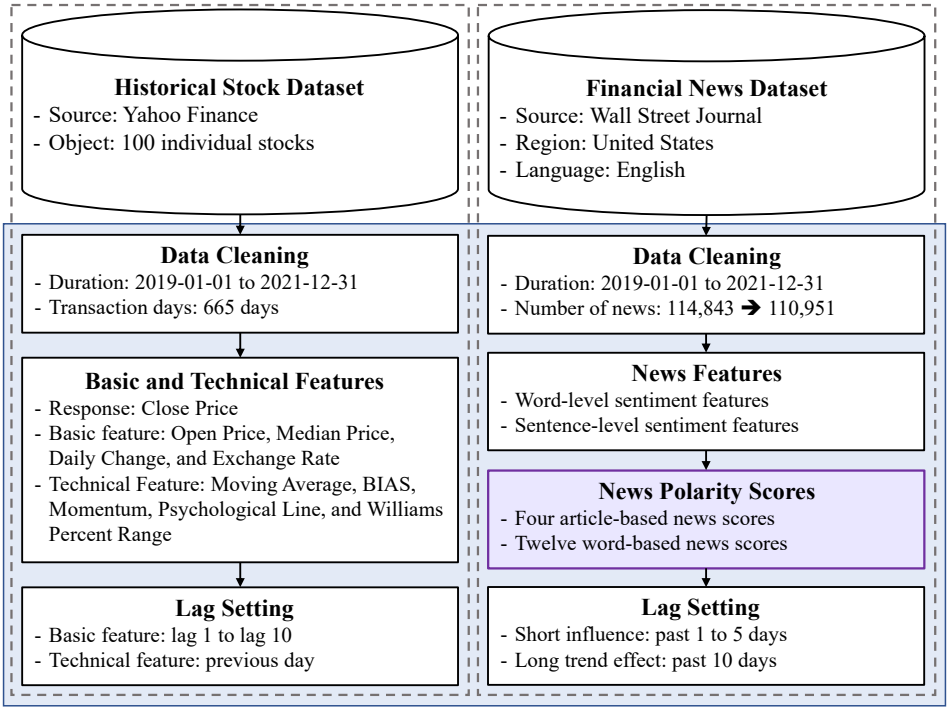

Fig. 1. Data processing procedure.

be schematized as shown in the flowchart in Fig. 1. The left dashed box is for the feature construction of the stock dataset, while the dashed box on the right is for the financial news dataset.

One of the major purposes of this research is to demonstrate the prediction power of the MARS model with extra news features on the large companies in the stock market. Therefore, the historical stock dataset of this study selected 100 stocks in the United States that comprised 100 companies with the highest market values in S&P 500 at the time of data collection. This dataset is used to evaluate the effectiveness of our newly proposed method in forecasting the up/down of the stock price and to calculate the basic and technical features. The detailed calculation formulas and lag setting will be introduced in Section 5.

The financial news dataset was collected from news articles from the Wall Street Journal (WSJ) published between January 1st, 2019, and December 31st, 2021. The number of original news from WSJ is 114,843. These news articles are used to establish news polarity scores, which contain invaluable information on the linkage between news and stock prices, aiming to improve the predicted up/down direction accuracy. The calculation method and lag setting will also be described in detail in Section 5.

Three types of models are built with all of the features above in various combinations. Least Absolute Shrinkage and Selection Operator (LASSO) regression is a linear model which serves as a baseline. We use the Multivariate Adaptive Regression Splines (MARS) model to build the local autoregression modeling to select important features in

various combinations, that also considers their interaction. One of the most competitive advantages of MARS model is that it can provide the coefficient of the feature effects, an essential part of the interpretability of a model, which most deep learning models lack. Since Long Short-Term Memory (LSTM) is often suggested in the literature for stock price prediction, this paper also explores the LSTM model with the parameters indicated in the relevant literature. We compare the predictive power of the three models. The LSTM is a black-box model. It is difficult to obtain a single coefficient for the important feature. The important feature effects in price up/down prediction are identified through the MARS models.

2. RELATED WORKS

2.1 Stock Price Prediction

Roondiwala *et al*. [1] use recurrent neural network (RNN) and LSTM to predict the price of the Indian stock market index NIFTY 50 from the national stock exchange. Idrees *et al*. [2] adjust the parameters of the Autoregressive Integrated Moving Average (ARIMA) model to build a statistical model that can effectively predict the price of stocks. Long *et al*. [3] propose a novel neural network method specifically for financial time series, Multi-Filters Neural Network (MFNN), to predict the price of the Chinese stock market index (CSI 300). The results show that the newly proposed model outperforms various machine learning and statistical models such as RNN, LSTM, *etc*. Kim and Won [4] introduce the method of combining time series and neural network models to build a new model. The new model they proposed combining LSTM and various Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) is called a hybrid LSTM model. It was found that GEW-LSTM, a hybrid model combining the LSTM model with three GARCH-type models, has better results. Mahfooz *et al*. [5] trained an LSTM model to forecast and classify stock trend for 120 different US stocks to automate trading by generating buy/sell signals based on the directional movement of the stock market index, which they referred to as an uptrend when its value increases, and a downtrend when the value decreases. Alkhatib *et al*. [6] suggest that LSTM has the best performance among many models when they include two additional features (High-Low and Open-Close) to the model when predicting the adjusted closing price of Apple, Tesla, Snapchat, and ExxonMobil stock, compared to using only the four features that were used traditionally (High, Low, Volume and Open). We will explain these features in Section 5. Meanwhile, Wang and Liu [7] propose a deep fusion model by utilizing LSTM to learn sequential information and adopt a Hybrid Attention Network, which includes both sentence-level and temporal attention, to find the relative importance of words from news articles, in order to predict the stock trend for the 500 companies in S&P500 index by combining semantic features from news content with historical stock prices. While neural networks have become popular, statistical models are still used in research to avoid the black-box problem of neural networks.

2.2 Sentiment Analysis

Recently, utilizing the sentiment of news and applying it to stock-market prediction have drawn a lot of attention in the research of financial technology [8, 9]. Mohan *et al*. [10] indicate a strong correlation between stock prices and news articles based on

existing sentiment analysis research. They collect S&P 500 stock prices and many companies related news. The results show a correlation between textual information and stock price direction. Agarwal [11] believes that most traders get their information from the news, making news a central factor in predicting changes in the stock market. Using a dictionary-based approach, the authors utilize the Python tool VADER to determine the sentiment value of sentences. The results show that the frequency of sentiment changes reflects stock volatility. Fan *et al*. [12] investigate Factor-Augmented Regularized Model for Prediction (FarmPredict), which can be applied to financial fields such as stock returns. The main calculation steps of FarmPredict are to learn hidden features from the article through principle component analysis, screen the idiosyncratic variables by correlation, and use LASSO to predict the price. Nemes and Kiss [13] give the prediction of stock value changes using sentiment analysis of stock news headlines. Sawale and Rawat [14] study the stock market prediction using sentiment analysis with Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs).

Bidirectional Encoder Representations from Transformers (BERT) is a robust language representation model introduced by Devlin *et al*. [15]. Unlike other well-known word embeddings extraction methods, such as Embeddings from Language Models (ELMo) and Generative Pre-trained Transformer (OpenAI GPT), Bidirectional Encoder Representations from Transformers (BERT) is designed to pre-train deep bidirectional representations from unlabeled text, with the advantage of capturing the contextual semanteme. Consequently, soon after the release of the approach, many researchers extend the application scope by adapting it to different domains [16–19].

From the financial perspective, Hiew *et al*. [20] construct a BERT-based sentiment index for three popular individual stocks in the Hong Kong market, which are highly discussed on Weibo.com. They use a Long Short Term Memory (LSTM) model henceforth and get convincing predictability for the return of the three stocks. Araci [21] constructs the FinBERT model by pre-training the model with financial data additionally. They attempt to pre-train the model on different sizes of text. Yu *et al*. [22] develop a chatbot based on BERT to cope with client questions in financial investment customer service. Farimani *et al*. [23] calculate the sentiment mood time series via the probability distribution of news embedding generated through a BERT-based transformer language model fine-tuned for financial domain sentiment analysis. They then use a deep recurrent neural network for feature extraction and a dense layer for price regression.

2.3 News Labeling

Yu [24] use 276,701 self-annotated comments on the 349 newspaper articles from *Financial Times* to construct credibility measures by labeling each observation as "bullish", "bearish" or "neutral" to make inference on the investment trend of these social network users. Similarly, Das *et al*. [25] use LM Dictionary to classify financial text into "positive", "negative" and "neutral" to denote the sentence's polarity. This polarity labeling approach has been adopted by several other researchers [26–29] using various NLP tools available, such as VADER, TextBlob, SentiWoordNet, *etc*..

In addition to labeling news with self-defined rules, many datasets are labeled manually, which refers to labeling with the professional domain knowledge by financial personnel, such as Meyer *et al*. [30] and Krishnamoorthy [31]. However, manual labeling is not only labor-intensive but also susceptible due to personal subjective influence.

Yadav *et al*. [32] consider the opening and closing prices in the news labeling. However, the news labels on the same day will be classified into the same category leading to ignoring the information in the news. The method proposed by Wang and Huang [33] take into account not only stock prices but also news-level information. This study suggests several novel news polarity scores by modifying and extending the results of Wang and Huang [33].

3. DATA DESCRIPTION

The 100 selected stocks (Appendix Table A.1) in S&P 500 between January 1, 2019, to December 31, 2021, were downloaded using yahoo financial (Download at `https://finance.yahoo.com/`) through the `quantmod` package in R software. The data contents are the daily opening price, closing price, high price, low price, and volume of each stock. Here we pick 4 of them, namely *XEL*, *DISCA*, *FTI*, and *SLB*, to draw a schematic diagram of their closing price trends. The purpose of Figure 2 is to show the trend of multiple stocks in one plot. The scales of the stock prices were adjusted for this purpose. By this multiple-scale figure, we can easily see the stock prices of *XEL*, *FTI*, and *SLB* dropped significantly around March 2020 due to the outbreak of the Covid-19 pandemic. Daily price fluctuations are very large for each stock, indicating the difficulty in the price up/down prediction.

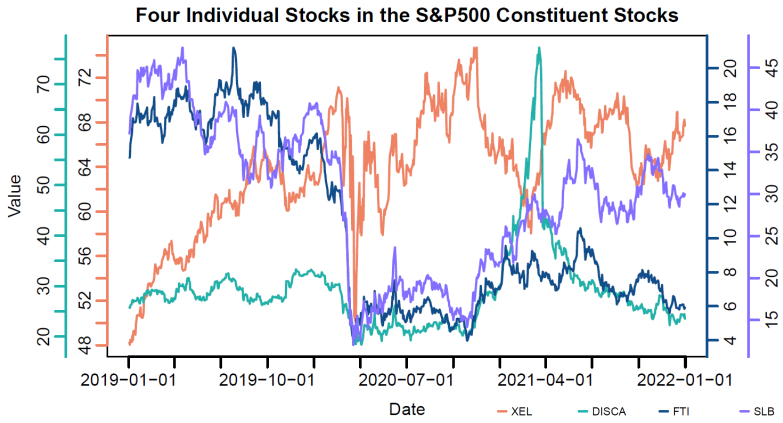

Fig. 2. The trend of stocks from 2019-01-01 to 2021-12-31; The scales of the stock prices are adjusted to fit the figure.

Jordan and Elgazzar [34] point out that due to the high volatility of the stock markets, we will miss out on the opportunities to have a better result if the model only learns from past information. Data quickly becomes stale, creating the need to use web crawlers to collect the newest data. Hence, we use the `selenium` package and `BeautifulSoup` package in python to collect 114,843 original news published between January 1st, 2019, to December 31st, 2021, from Wall Street Journal (`https://www.wsj.com/`) through the university library.

We removed the news that does not contribute to the sentimental information, such as Crossword and picture descriptions. We also found that news with more than 2,000 words were mostly interview dialogues or film transcripts, which were unsuitable for follow-up analysis. Consequently, we keep only the news with 100 to 2000 words in the subsequent analysis. After data cleaning, the final number of news that forms our financial news dataset is 110,951. About 3.39% of the news was removed.

4. PROPOSED NEWS POLARITY SCORES AND MODELS

4.1 News Polarity Scores

We innovated four news polarity scores inspired by Wang and Huang [33]. The novel parts are that the weight for the price rate of change is calculated differently, and the transaction volume is now considered. The central concept of the news polarity score is to link news and stock price using Term Frequency and Inverse Document Frequency (TF-IDF) to calculate the importance of words in a news. The polarity score also contains the adjusted price Rate of Change (ROC) with the trading volume of target stock. The polarity score is defined as follows,

$$Score(w_i, n_j, d, I) = tfidf(w_i, n_j) \times AR(t(n_j), d) \times V(t(n_j), I), \tag{1}$$

where w_i represents the ith word in news corpora, n_j is the jth news in our dataset, $t(n_j)$ denotes the date that the news n_j was published, d represents the lag of days when calculating the ROC, I is a indicator variable of using trading volume, AR and V refer to the adjusted ROC and trading volume. It is noted that the same word w_i will have a different score calculated by other news n_j. The three components of $Score(w_i, n_j, d, I)$ are described as follows,

- $tfidf(w_i, n_j)$ refers to the Term Frequency (TF) and Inverse Document Frequency (IDF) of word w_i for news n_j. TF is the frequency of a specific word appearing in the news, while IDF is the total number of news in the corpora divided by the number of news containing the word w_i. TF-IDF can be expressed as follows,

$$\begin{aligned} tfidf(w,n) &= tf(w,n) \times idf(w) \\ &= \frac{f_n(w)}{\sum_{w \in n} f_n(w)} \times ln\left(\frac{|D|}{|\{n \in D : w \in n\}|}\right), \end{aligned} \tag{2}$$

  where $f_n(w)$ denotes the frequency of the word w in the news n, D is the corpus of news.

- $AR(t(n_j), d)$ is an adjustment of the price Rate of Change (ROC). ROC is a technical momentum indicator, that measures the percentage change in the closing price of the target stock or target indicator. The ROC is defined as,

$$ROC(t,d) = \frac{close(t-1) - close(t-1-d)}{close(t-1-d)}, \tag{3}$$

  where $close(t')$ is the closing price on date t'.

The Adjusted ROC (AR) considers the price fluctuations to highlight the potential news which has a more significant impact on the stock price. A large proportion of fluctuations gives a heavier weight. The adjusted ROC is defined as follows,

$$AR(t,d) = Weight(ROC) \times ROC(t,d), \tag{4}$$

where $Weight(ROC)$ denotes the weight adjusted by the ROC in different ranges as in Table 1. These weights may be changed to increase or decrease the impact of price fluctuation on the resulting score.

Table 1. Selected weight for each ROC range.

ROC Range	Weight
$(-0.005, 0]$, $(0, 0.005]$	1
$(-0.01, -0.005]$, $(0.005, 0.01]$	2
$(-0.015, -0.01]$, $(0.01, 0.015]$	3
$(-0.02, -0.015]$, $(0.015, 0.02]$	4
$(-0.025, -0.02]$, $(0.02, 0.025]$	5
$(-\infty, -0.025]$, $(0.025, \infty)$	6

- $V(t(n_j), I)$ accounts for the trading volume effect. When I is 0, the trading volume is not considered; otherwise, the calculated score will include the trading volume of the previous day of the corresponding news. The formula is:

$$V(t,I) = \begin{cases} 1, & I = 0 \\ Volume(t-1), & I = 1. \end{cases} \tag{5}$$

We now propose four aggregated scores with different semantics for subsequent analysis, namely article-based news scores (ABNS), word-based news direction scores (WBNDS), word-based news positive scores (WBNPS), and word-based news negative scores (WBNNS). The word in the adjusted Loughran-McDonald's Dictionary is called an ALM word. All these aggregated scores are calculated only by the ALM words. These four innovative news polarity scores integrate the word sentiment of news and the volatility in the stock market. We anticipate that adding these scores as extra features will help the model more accurately predict the stock price up/down direction.

1. **Article-Based News Score (ABNS)** is the score cumulation of all the words in a news n_j, defined as,

$$ABNS(n_j) = \sum_{w_i \in n_j} Score(w_i, n_j, d, I). \tag{6}$$

2. **Word-Based News Direction Score (WBNDS)**. The word impact score (WIS) is the score cumulation of a word in all news, defined as,

$$WIS(w_i) = \sum_{n_j} Score(w_i, n_j, d, I). \tag{7}$$

The word direction (WD) of a news is determined by the WIS, defined as,

$$WD(w_i) = \begin{cases} 1, & WIS_i \geq 0 \\ -1, & WIS_i < 0. \end{cases} \tag{8}$$

Then, the word-based news direction score (WBNDS) of news n_j is the accumulation of the WD values of all words in a news n_j, defined as,

$$WBNDS(n_j) = \sum_{w_i \in n_j} WD(w_i). \tag{9}$$

3. **Word-Based News Positive Score (WBNPS)** is the score cumulation of words with positive WIS values in a news n_j, defined as,

$$WBNPS(n_j) = \sum_{w_i \in n_j} Score(w_i, n_j, d, I), \text{ where } WIS(w_i) \geq 0. \tag{10}$$

4. **Word-Based News Negative Score (WBNNS)** is the score cumulation of words with negative WIS values in a news n_j, defined as,

$$WBNNS(n_j) = \sum_{w_i \in n_j} Score(w_i, n_j, d, I), \text{ where } WIS(w_i) < 0. \tag{11}$$

4.2 Long Short-Term Memory (LSTM)

Long Short-Term Memory network (LSTM) is an extension of a standard recurrent neural network (RNN) that can store short-term memory for a long time and is able to use contextual information when mapping between input and output sequences. [35] It eliminates the vanishing gradient problem that prevents an RNN model from propagating useful gradient information as the impact of a given input either decays or inflates exponentially with each cycle of iteration. This is addressed by the gating mechanism in LSTM, namely the input gates (ability to write), the output gates (ability to read) and the forget gates (ability to reset), which are the multiplicative units. There are memory blocks with cells that take in different pieces of information, *i.e.*, the current input, the short-term memory from the previous input (hidden state), and the long-term memory (cell state). The self-loop gates preserve and access info for a long time and act as a regulator to decide which information to be retained or scrapped before passing the relevant data to the next cell.

The logistic sigmoid is generally used as the gate activation function to let the activation value falls between 0 (gate close) and 1 (gate open). This produces a clear instruction whether to let everything pass through or block it. Besides, hyperbolic tangent (tanh), rectified linear unit (ReLU), or logistic sigmoid are typically employed as cell activation functions. The self-loop is attached to the most essential component, a state unit $s_i^{(t)}$, where its weight is controlled by a forget gate $f_i^{(t)}$ to update the internal state of an LSTM cell [36]. The mathematical formula of an LSTM structure is as follows,

$$
\begin{aligned}
f_i^{(t)} &= \sigma\left(b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)}\right), h_i^{(t)} = \tanh\left(s_i^{(t)}\right) q_i^{(t)}, \\
s_i^{(t)} &= f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \sigma\left(b_i + \sum_j U_{i,j} x_j^{(t)} + \sum_j W_{i,j} h_j^{(t-1)}\right), \\
g_i^{(t)} &= \sigma\left(b_i^g + \sum_j U_{i,j}^g x_j^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)}\right), \\
q_i^{(t)} &= \sigma\left(b_i^q + \sum_j U_{i,j}^q x_j^{(t)} + \sum_j W_{i,j}^q h_j^{(t-1)}\right),
\end{aligned}
\tag{12}
$$

where $\mathbf{x}^{(\mathbf{t})}$ is the current input vector, $\mathbf{h}^{(\mathbf{t})}$ is the currently hidden layer vector containing all output from the cells, and b, U, W are biases, input weights and recurrent weights respectively, for the three gates. $g_i^{(t)}$is the external input gate, $q_i^{(t)}$ is the output gate, and $h_i^{(t)}$ is the output of the LSTM cell. LSTM dynamically controls the time scale by simultaneously controlling the forgetting factor and the decision to update the state unit, and therefore serves its purpose of reserving and retrieving beneficial information over a long period. The discussion on determining the parameters of the LSTM model will be given in the evaluation session.

4.3 Least Absolute Shrinkage and Selection Operator (LASSO)

The Least Absolute Shrinkage and Selection Operator, more commonly known as LASSO, is a type of linear regression that uses shrinkage or regularization to set coefficients of some wrongly learned variables exactly to zero to avoid over-fitting. Hence, LASSO can select a set of variables that are important for the model and make the generated model more accessible to interpret than a model with all variables included. A simple linear regression model is expressed as,

$$Y = X\beta + e, \tag{13}$$

where Y is the response variable, X is the input feature, β is the coefficient of variable and e is the standard error.

In LASSO, the variable coefficients β are minimized by adding an ℓ_1 penalty equal to the absolute magnitude of coefficients. This penalty forces some of the coefficient estimates to be equal to zero when the tuning parameter λ, which controls the bias-variance tradeoff, is large enough [37]. The minimization objective is to look for the set of variable coefficients that produce the smallest RSS value,

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| = \mathrm{RSS} + \sum_{j=1}^{p}|\beta_j|, \tag{14}$$

$$\hat{\beta} = \arg\min_{\beta}\left\{\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2\right\}, \tag{15}$$

where n is the number of observations, and p is the number of input features.

We will need to select a suitable value for the tuning parameter λ, and this is solved by choosing a grid of λ values and using cross-validation to obtain the one that comes with the smallest error. The model is then rerun with a complete set of data available and the selected λ value, generating an optimal model for the data set. In short, LASSO model is a popular method to use on a dataset with high multicollinearity and high dimensionality.

4.4 Multivariate Adaptive Regression Splines (MARS)

The basic idea of Multivariate Adaptive Regression Splines (MARS) [38, 39] is to establish a local regression model for each interval and fit it by complex linear or nonlinear functions. MARS uses piecewise linear basis $h_(x)$ functions of the form $(x-t)_+$ and $(t-x)_+$, and each contains a reflected pair. When the basis function has a node at value t, the formula is defined as,

$$(x-t)_+ = \begin{cases} x-t, & \text{if} \quad x>t \\ 0, & \text{otherwise} \end{cases} \quad \text{and} \quad (t-x)_+ = \begin{cases} t-x, & \text{if} \quad x<t \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

These basis function products make identifying additive and interactive properties among multiple variables possible. The MARS model can be expressed as follows,

$$f(x) = \beta_0 + \sum_{i=1}^{k} \sum_{m=1}^{m_i} \beta_{im} h_{im}(x_i), \tag{17}$$

where each $h_m(x)$ denotes the m^{th} basis function or a product of multiple basis functions, x is the input feature, k is a number of input features, β_0 and β_m are the intercept and the coefficients corresponding to $h_m(X)$ respectively. The coefficients of the model are estimated by minimizing the residual sum of squares.

The first stage of the MARS algorithm performs forward stepwise search that considers all products of candidate pairs with basis functions in the model each time. The product that reduces the largest residual squared error is added to the current model. This process continues until the number of terms in the model reaches a preset maximum value. For the second stage of the MARS algorithm, the model will take backward deletion, removing the term from the model with the smallest increment of the residual squared error, finding the best combination of basis functions under each lambda (λ) to generate the best model $\hat{f}_\lambda$. The value of λ can be optimized using Generalized Cross-Validation (GCV), defined as

$$GCV(\lambda) = \frac{\sum_{i=1}^{N} (y_i - \hat{f}_\lambda(x_i))^2}{(1-M(\lambda)/N)^2}. \tag{18}$$

$M(\lambda)$ is a valid number of parameters in the model. The lowest $GCV(\lambda)$ model is selected as the final MARS model.

5. FEATURE CONSTRUCTION

The price up/down prediction features include the commonly used basic and technical features, word and sentence level news features, and the proposed news polarity scores. All of them have their corresponding lag settings.

5.1 Basic and Technical Features

We considered 4 basic features and 5 technical features that investors often use to judge stock price volatility in practical analysis. The 4 basic features included the "Closing Price", "Opening Price", "Median Price", "Daily Change" of the particular stock we are working on, The 5 technical features are obtained by further calculation using historical data, and it includes "Moving Average", "BIAS", "Momentum", "Psychological Line" (PSY) and "Williams Percent Range" (WMS) [40, 41] of the stock. To add the related international information with special interests, we also considered 4 weighted indexes, which are S&P500 in the United States, Shanghai Stock Exchange (SSE) in China, Taiwan Weighted Index (TWII) in Taiwan and Hang Seng Index (HSI) in Hong Kong. Each weighted index produced its own 4 basic features and 5 technical features. We also included 3 "Exchange Rates" between the four regions (USD to RMB, USD to TWD and USD to HKD) as extra technical features.

5.2 Word and Sentence News Features

Loughran and McDonald [42] create their financial-specific dictionary called the Loughran-McDonald dictionary (LM dictionary), which consists of words prevalent in 10-K reports and thus is more specific to the finance domain. After adding extra words of the positive and negative wordlist categories (Appendix Table A.2)in the Loughran-McDonald's Dictionary (LMD) and constructing 9 specific wordlist categories (Appendix Table A.3), the 11 word-level sentiment features are established by counting the daily occurrences of keywords in these 11 categories. Our selection of specific categories is to characterize some important themes and events that may affect the movement of stock prices. The 9 specific categories are *Disaster*, *Publish*, *Acquisition*, *Lawsuit*, *President*, *Bear*, *Bull*, *Bankrupt* and *Highlights*. In this paper, we further transform *Bear* and *Bull* into two new categories, namely *Agreement* and *Bear-AG*, according to Oliveira *et al*. [43] by using the following formulation,

$$\begin{aligned} Agreement &= 1 - \sqrt{1 - \left(\frac{NBull_t - NBear_t}{NBull_t + NBear_t}\right)^2}, \\ Bear\text{-}AG &= 1 - \sqrt{\frac{2 * NBear_t}{NBull_t + NBear_t}}. \end{aligned} \tag{19}$$

To extract sentence-level sentiment features, we utilize a fine-tuned BERT model on the news we have collected. After decomposing each news into sentence units and input into the BERT model, each sentence's probability of three emotion categories (positive, neutral and negative) will be obtained. A sentence is defined as negative when its probability is higher than the third quartile of the probabilities in the negative emotion category. A sentence is defined as positive when its probability is higher than the third quartile of the probabilities in the positive emotion category. The remaining sentences are defined as neutral. Then, the number of negative and positive sentences per day is calculated and become the 2 features on sentence-level sentiment.

5.3 News Polarity Scores

Four aggregated scores of a news defined in Section 4 are studied. They are article-based news scores (ABNS), word-based news direction scores (WBNDS), word-based news positive scores (WBNPS), and word-based news scores news negative scores (WBNNS). Each aggregated score produces 4 polarity score features of a news with $d \in 1,2$ and $I \in 0,1$. 4 polarity score features of a day are obtained by the sum, average, maximum and minimum values of the polarity scores for all the news in the day.

5.4 Lag Setting

The lag effect for basic features is set to 10 days. We also account for the short-term effect and long-term trend for the news features and news polarity score features. The short-term effect refers to the difference between the current day and the past 1-5 days, and it is recorded as $Feature.MO1$, $Feature.MO2$, ..., and $Feature.MO5$, where $Feature$ represents one of the news features. The long-term trend is the movement over the past 10 days, defined as $TREND = (Max_t(10) - T)/(Max_t(10) - Min_t(10))$, where $Max_t(10)$ and $Min_t(10)$ are the maximum and minimum values for time t in the past 10 days, and T is the value of the current feature. The adjusted feature for the long trend effect is recorded in the analysis as $Feature.TREND10$. We do not set additional lag for the 5 technical features because the historical information is already included within the model. Other than PSY feature and WMS feature, the rest of the technical features have set the lag to consider n day, where $n = \{2,3,...,10\}$. Based on the suggestions from the literature, The lag of PSY is in a multiple of 6 ($n = \{6,12,24\}$) and the lag of WMS is a multiple of 7 ($n = \{7,14,28\}$).

5.5 The Day-of-the-Week Effect

The impact of holiday and weekday news on the stock market, *i.e.*, the day-of-the-week effect, is also taken into account in our analysis [44, 45]. When a news is released over the weekend, investors have more time to gather information and devise strategies. The company also tends to release important news on weekends, allowing more time to operate and stabilize stock prices. The difference in the influence of news published on weekdays and weekends is the day-of-the-week effect. Assuming the relationship between the day-of-the-week effects is an exponential function, heavier weight is given to recent news. When the stock market is closed on t national holiday or a special day, then the modified news feature on the first day after the day t can be calculated as $M'_{t+1} = e^{-\lambda t}M_1 + e^{-\lambda(t-1)}M_2 + ... + e^{-\lambda*1}M_t + e^{-\lambda*0}M_{t+1}$, where M_1, M_2, ..., M_t are the news features on the first day to the last day during the holiday and M'_{t+1} and M_{t+1} are the modified and original news features on a day after the day t. The best lambda value is searched from interval stock between 0 to 5 and by step 0.1, and we select the one that produces the highest prediction accuracy in the training dataset.

5.6 Feature Groups

Different feature groups used in the model are summarized in Table 2. The numbers in the table represent the features put into the LSTM, LASSO and MARS models. The same time lag effect is considered for all three models. As mentioned in Sections 5.1 and

5.4, we have 4 basic features (Open Price, Close Price, Median, Daily Change) with 10 lags, 3 technical features (Moving Average, BIAS, Momentum) with 9 lags, 2 technical features (PSY, WMS) with 3 lags, of individual stock that we are working on and 4 indexes, and 3 exchange rates with 10 lags ($5 \times [(4 \times 10) + (3 \times 9) + (2 \times 3)] + 3 \times 10 = 395$ features).

The Word-Level Sentiment Features include 11 categories of news features and Sentence-Level Sentiment Features have another 2 categories about the sentence polarity. Both Word-Level and Sentence-Level Sentiment come with short-term effects and long-term trends. Hence, Word-Level Sentiment has 77 features considered (news feature + effect of past 5 days + long-term trend = 7 features for each of the 11 categories), while Sentence-Level Sentiment has 14 features (news feature, past 5 days effect and long-term trend = 7 features for each of the 2 categories).

We have discussed in Section 5.3 that each aggregated score generates 4 polarity score features, and these polarity score features have their own set of short-term effects and long-term trends. So, each of *ABNS*, *WBNDS*, *WBNPS* and *WBNNS* has 28 features (7 features for each of 4 polarity scores). Note that the sum, average, maximum and minimum values of the polarity scores of all the news in each day are only calculated in the *ABNS*4-*WSS* feature group and the *ABNS*4-*BTF* feature group. Therefore these 2 feature groups have 112 features included (4×28 features) instead of 28 features in other feature groups which only considered the sum value of the polarity scores of all the news each day.

6. EVALUATION

6.1 Experimental Structure

The training data period is from 2019-01-01 to 2021-06-30, with 551 standard trading days, while the testing data period is from 2021-07-01 to 2021-12-31 with 114 standard trading days. Table 3 shows the number of days when the close price of the 4 stocks (*XEL*, *DISCA*, *FTI*, and *SLB*) drop or rise compared to the previous day during the test period. This table shows the typical situation of up and down proportions of the stock price.

The actual fluctuating direction and the predicted direction are labeled as *ALabel* and *PLabel*, as defined below,

$$ALabel = \begin{cases} 1 \quad \text{(Up)}, & if \quad C_{t-1} \leq C_t \\ -1 \quad \text{(Down)}, & if \quad C_{t-1} > C_t, \end{cases}$$

$$PLabel = \begin{cases} 1 \quad \text{(Up)}, & if \quad C_{t-1} \leq P_t \\ -1 \quad \text{(Down)}, & if \quad C_{t-1} > P_t, \end{cases} \tag{20}$$

where C_t denotes the value of the closing price on the day t, C_{t-1} denotes the value of the closing price on the previous day of t, and P_t indicates the prediction price on the day t.

Table 2. Summary for feature groups.

Group Name	Basic and Technical Features	News Features						Number of Features
		Word -Level Sentiment Features	Sentence -Level Sentiment Features	News Polarity Scores				
				ABNS	WBNDS	WBNPS	WBNNS	
BTF	395	x	x	x	x	x	x	395
WS	395	77	x	x	x	x	x	472
WSS	395	77	14	x	x	x	x	486
ABNS-WSS	395	77	14	28	x	x	x	514
ABNS4-WSS	395	77	14	112	x	x	x	598
ABNS4-BTF	395	x	x	112	x	x	x	507
WBNDS-WSS	395	77	14	x	28	x	x	514
WBNPS-WBNNS-WSS	395	77	14	x	x	28	28	542
ALLNPS-WSS	395	77	14	28	28	28	28	598
ALLNPS-WS	395	77	x	28	28	28	28	584
ALLNPS-BTF	395	x	x	28	28	28	28	507

Table 3. The day's proportion in the test data that the close price was down or up compared to the previous day.

	XEL	DISCA	FTI	SLB
Down	53 (46.49%)	65 (57.02%)	61 (53.51%)	60 (52.63%)
Up	61 (53.51%)	49 (42.98%)	53 (46.49%)	54 (47.37%)

The expanding windows method was taken for training and prediction. The start date of the training set is fixed for all cases, and the end date of the training set is assigned to be the day before the prediction date. That is, the window size of the training set will keep growing over time, and it is a one-day ahead prediction. There are 114 transaction days in the test set, so we need to train each model 114 times.

6.2 Determining Parameters of the LSTM Model

Determining a proper number of parameters in a RNN model is always challenging. We have referred to three related articles on stock price prediction using the LSTM model [46–48]. The parameters setting included the number of units in LSTM layer, number of epochs, activation function used and loss function. The parameters settings are summarized in Table 4. Based on these articles, we practiced their suggested parameters in three selected stocks from S&P 500, namely *DISCA*, *MRK* and *XEL*, to search for the

proper parameters. The combination of parameters that gave the best result is listed in Table 5. We use `ts.lstm` function in `TSLSTM` package in `R` to run the LSTM model by setting the parameters as in Table 5.

Table 4. Summary of parameters setting of the LSTM in the related works [46–48].

Data Source	Daily trading data of Apple Inc., Amerisource Bergen Corporation, and Cardinal Health	Daily trading data of Agriculture Development Bank and daily financial news ShareSansar headlines	Daily trading data of Apple transformed to (0,1) scale
Period (Train:Test Ratio)	From 2008-01-01 to 2020-09-15 (80:20)	From 2011-03-20 to 2019-11-14 (60:25, 15% validation)	(Train) from 2010-01-03 to 2020-02-28 (Test) from 2020/02/28 to 2020/04/29
Units	100	120	120
Epochs	150	100	200
Activation	ReLU	tanh	Not stated
Loss	MSE	MAE	MSE
Best Result	(RMSE) 5.816 - 6.120 for the three stocks	(RMSE) 23.07	(RMSE) 0.01246 ((0,1) scale)

Table 5. Finalized parameters setting of the LSTM model.

Number of Units	Number of Epochs	Activation Function	Loss Function
128	100	ReLU	MSE

6.3 Price Up/Down Prediction

First, overall evaluations of the three models are assessed by the accuracy, sensitivity, specificity, and F1-score of the predictions for the 100 stocks. When the stock price is predicted to be going up or down, the predicted direction is labeled as positive or negative, respectively. Furthermore, when the predicted direction is the same as or different from the actual direction of stock price change, the classification result is labeled as true or false, respectively. Hence, under the binary classification, there will be 4 outcomes, namely "true positive" (TP), "true negative" (TN), "false positive" (FP), and "false negative" (FN). Accuracy is the percentage of total sample that is classified correctly ($(TP+TN)/(TP+TN+FP+FN)$), sensitivity calculates the ratio of correctly-identified positive to actual positive ($TP/(FN+TP)$), specificity measures the ratio of correctly-identified negative to actual negative ($TN/(FP+TN)$), and lastly, F1-score is the weighted average of precision (ratio of correctly-identified positive to all predicted positives, $TP/(FP+TP)$) and sensitivity that provides information on the performance of our model's classification ability ($2*(\text{precision}*\text{sensitivity})/(\text{precision}+\text{sensitivity})$).

As we can see from Table 6, out of the 3 prediction models, the MARS model is able to achieve the highest accuracy, sensitivity, and F1-score when predicting the movement of stock price, regardless of whether BTF or $WBNPS$-$WBNNS$-WSS feature group is

Table 6. Confusion matrices and performance measurements for overall evaluation of 100 stock predictions.

		Actual Direction					
		MARS		LASSO		LSTM	
		Up	Down	Up	Down	Up	Down
Up	Predicted direction based on BTF	3789	1996	3186	1481	2738	2566
Down		1986	3629	2589	4144	3037	3059
Up	Predicted direction based on WWW	3812	1990	3085	1420	2786	2652
Down		1963	3635	2690	4205	2989	2973
Accuracy	BTF	0.6507		0.6442		0.5085	
	WWW	0.6532		0.6395		0.5052	
Sensitivity	BTF	0.6561		0.5534		0.4741	
	WWW	0.6601		0.5342		0.4824	
Specificity	BTF	0.6452		0.7367		0.5438	
	WWW	0.6462		0.7476		0.5285	
F1-score	BTF	0.6555		0.6109		0.4943	
	WWW	0.6585		0.6002		0.4969	

Note: WWW stands for WBNPS-WBNNS-WSS feature group.

used, although its specificity is noticeably lower than LASSO model. The results of all 4 assessments (accuracy, sensitivity, specificity, and F1-score) also improved when we use the *WBNPS-WBNNS-WSS* feature group, that include news polarity scores and word and sentence level sentiment scores, to predict the stock trend, especially when using MARS model, which is 0.6532, 0.6601, 0.6462, and 0.6585, respectively, compared to using just BTF, which is 0.6507, 0.6561, 0.6452, and 0.6555, respectively. This result is aligned with the goal of our study, which is to establish functional news polarity scores for better prediction accuracy. Interestingly, LASSO model has the higher specificity among the 3 models, which is 0.7367 when using the BTF feature group and 0.7476 when using the *WBNPS-WBNNS-WSS* feature group.

In Fig. 3, 20 stocks were selected to show the improvement of stock price prediction accuracy using news features and BTF. The stock order from left to right in the figure is based on the difference in prediction accuracy between the use of baseline BTF and the best news feature group in the MARS model. The points with circle, cross and triangle shapes denote the accuracies from the MARS, LASSO, and LSTM, respectively. The colors of the points represent the feature groups. The MARS model's price up/down prediction accuracies with BTF are connected by a black line in the figure. To demonstrate the effectiveness of the MARS model, we have also included the performance of the LASSO model and LSTM model by connecting their BTF values using dashed line and dotted line respectively. Each stock has 11 feature groups used in the MARS and LASSO models. Some of the prediction accuracies are the same and those points are overlapped in the figure. To reduce the complexity of the figure, we only include two prediction accuracies of the LSTM corresponding to the BTF and *WBNPS-WBNNS-WSS* feature groups.

The stock with the most improvement in prediction accuracy under the MARS model was XEL, also known as Xcel Energy Incorporation, an electricity and natural gas transmission company in the United States. The MARS model with *WBNPS-WBNNS-WSS*

feature group increased 7.89% prediction accuracy than that of BTF feature group from 60.53% to 68.42%. As seen in Fig. 3, although the LASSO indeed has better accuracy in a few stocks, it does not achieve the same level of improvement in accuracy as a result from MARS. To our surprise, a deep learning model as formidable as LSTM does not seem to perform well in our research as its prediction accuracy is the worst compared to the result of the MARS and LASSO models. We will discuss the possible reasons in the conclusion section.

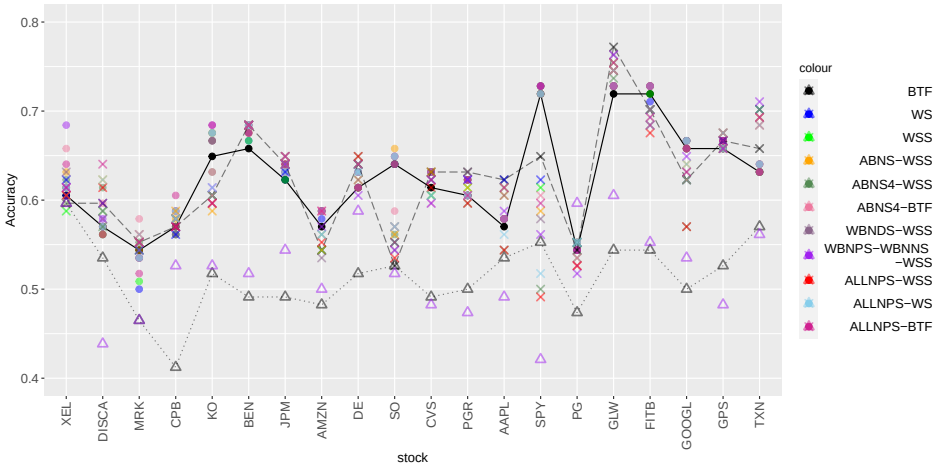

Fig. 3. The price up/down prediction accuracy of 20 selected stocks of the MARS (circle), LASSO (cross) and LSTM (triangle) models using expanding windows method. The colors are corresponding to the feature groups listed in the legend. The stock order from left to right is based on the difference in prediction accuracy between the use of baseline BTF and the best news feature group in the MARS model.

Out of the 100 stocks, 52 stocks have an accuracy higher than 65% in MARS model and their prediction accuracies are drawn in Fig. 4. The order of the stocks in the figure follows the decreasing trend of best prediction accuracies obtained by the MARS model among all 11 feature groups. Among them, the four stocks with prediction accuracies above 75% are *FTI* (TechnipFMC plc, a multinational oil and gas company providing technical solutions to the global energy industry), *SLB* (Schlumberger Limited, the world's largest oilfield services company), *RF* (Regions Financial Corp, a bank holding company), and *GE* (General Electric Company, an American multinational conglomerate operating in the electronics industry, energy industry, *etc*.).

Under the MARS model with the BTF feature group, the prediction accuracy of these four stocks achieves 78.95%, 77.19%, 76.32% and 76.32%, respectively. We also include the results from the LASSO and LSTM models in this figure for comparison. Although there are few occasions where the LASSO model gave a higher accuracy, most stock prices are being predicted more accurately using the MARS model. It is also apparent that the prediction accuracy of the LSTM model is almost always lower than both MARS and LASSO, as the dotted line that connects LSTM's *BTF* value lies below that of MARS and LASSO. To emphasize the effect of news polarity scores in price prediction, Table 7 shows the MARS model's prediction accuracy of the stock price up/down fluctuation of

XEL, *DISCA*, *FTI* and *SLB* under the 11 feature groups. The up/down predictions of the two stocks, *XEL* and *DISCA*, benefited from the features of news polarity scores.

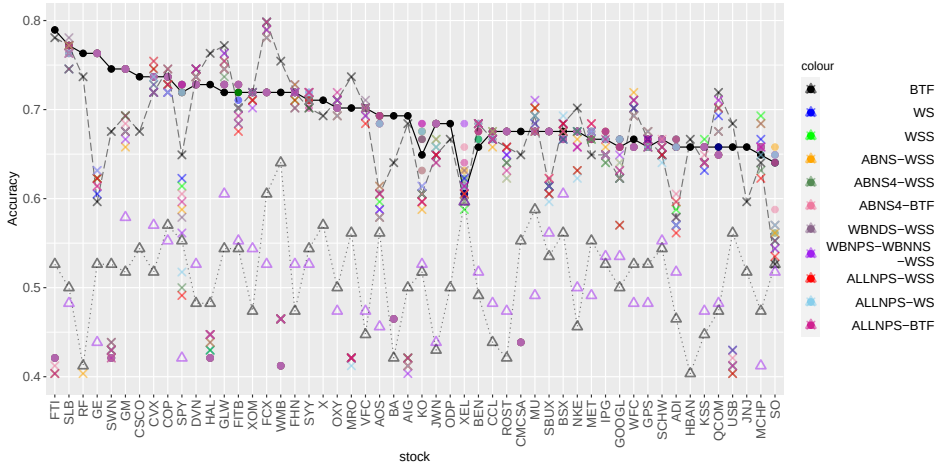

Fig. 4. The price up/down prediction accuracy of 52 selected stocks of the MARS (circle), LASSO (cross) and LSTM (triangle) models using expanding windows method. The colors are corresponding to the feature groups listed in the legend. The order of the stocks follows the decreasing trend of the best prediction accuracies obtained by the MARS model among all 11 feature groups.

Table 7. The MARS's price up/down prediction accuracies of four stocks using expanding windows method with different feature groups.

Group	XEL	DISCA	FTI	SLB
BTF	0.6053	0.5702	0.7895	0.7719
WS	0.6053	0.5614	0.4211	0.7719
WSS	0.6140	0.5614	0.4211	0.7719
ABNS-WSS	**0.6316**	0.5614	0.4211	0.7719
ABNS4-WSS	**0.6228**	**0.6140**	0.4211	0.7632
ABNS4-BTF	**0.6579**	**0.5965**	0.4211	0.7719
WBNDS-WSS	0.6140	0.5702	0.4211	0.7719
WWW	**0.6842**	**0.5789**	0.4211	0.7632
ALLNPS-WSS	0.6140	0.5702	0.4211	0.7632
ALLNPS-WS	0.6140	0.5702	0.4211	0.7632
ALLNPS-BTF	**0.6404**	0.5614	0.4211	0.7719

6.4 Important Features

Due to the better performance in price Up/Down prediction, the MARS model is used for the feature selection. The importance of the feature was based on the GCV and the mean GCV value was calculated over the 114 models corresponding to the prediction of each test day. Here, we take *XEL* using the *WBNPS-WBNNS-WSS* feature group and *DISCA* using the *ABNS*4-*WSS* feature group as examples to illustrate the benefits of using

news polarity scores.

For the stock *XEL* using the *WBNPS-WBNNS-WSS* feature group, the most important feature is unquestionably the daily open price of the stock. Of the top 30 important features selected by the MARS model, 9 are the proposed news polarity scores. In the order of importance, they as listed below:

1.*WBNNS_score*10.*MO*4, 2.*WBNPS_score*11.*MO*2, 3.*WBNNS_score*21.*MO*1,
4.*WBNPS_score*10.*MO*3, 5.*WBNPS_score*11, 6.*WBNNS_score*11,
7.*WBNNS_score*10.*MO*3, 8.*WBNPS_score*10.*MO*1, 9.*WBNPS_score*21.*MO*2.

The first three news polarity scores in the list take the 4^{th}, 6^{th} and 8^{th} place in the order of importance, which demonstrated our proposed scores are relatively important in price up/down prediction.

The MARS model for the stock *DISCA* using the *ABNS*4-*WSS* feature group only selected 11 important variables. The most important feature is, again, the opening price of the day. 8 out of the 11 important variables are the news polarity scores, and they occupy the 4^{th} to the 11^{th} places. They are

1.*ABNS_score*21_*min*.*MO*2, 2.*ABNS_score*21_*sum*.*MO*1, 3.*ABNS_score*11_*mean*,
4.*ABNS_score*21_*mean*.*MO*2, 5.*ABNS_score*21_*min*.*MO*5, 6.*ABNS_score*21_*min*,
7.*ABNS_score*21_*min*.*MO*4, 8.*ABNS_score*11_*min*.*MO*5.

Moreover, unlike the deep learning model LSTM, the MARS model has coefficients that help to clarify the relationship between features and stock prices, which is important for the interpretability of the model.

6.5 Feature Effect for Price Up/Down Prediction

We now discuss the feature effect for the price up/down prediction in the MARS model. The examples are the *XEL* and *DISCA* price up/down prediction using the *WBNPS-WBNNS-WSS* feature group and the *ABNS*4-*WSS* feature group, respectively.

This study uses one-day ahead prediction, 1 model was built in each of 114 test days for every stock. Among all 114 models built for the stock *XEL* and a total of 896 terms were selected in all the MARS models with the *WBNPS-WBNNS-WSS* feature group. Table 8 presents the terms selected at least 10 times by the MARS models. The most frequently chosen main factor was *XEL.Open*, which amounted to 228 times in total. The MARS model is a local regression model, where the value of the cut point of a variable is called the node cut value, as demonstrated in the table below. The other three features selected more than 10 times are our newly proposed scores.

It is worth noting that the feature with the second highest frequency is *WBNNS_score*10.*MO*4, was selected as frequently as 227 times, with the mean and standard deviation of the coefficients being 0.273 and 0.185, respectively. If there is a huge difference between the *WBNNS_score*10 on the day $t-1$ and the *WBNNS_score*10 of the previous 4 days from day $t-1$, we may be able to predict the movement of the stock price with higher accuracy. As long as the coefficient is greater than the cut value –4.054, this feature positively affects the stock price prediction.

In addition, we are also able to capture the feature interaction effect in this model, as shown by the combination of *HSI.PSY*12 and *WBNNS_score*10.*MO*4 which has appeared 86 times, suggesting a potential relationship between them. Based on Table 8, it proposes that if the percentage of the number of days in which the stock price has risen in the past 12 days is greater than 33.333, and if the difference between the value of *WBNNS_score*10

on the day $t-1$ and value of $WBNNS\_score10$ of previous 4 days from day $t-1$ is more than –3.777, the product of these two feature values will exert a negative impact on the prediction of stock price.

Table 8. Feature effect in the MARS model for the stock *XEL*.

Feature Name	Frequency	Mean of coefficient	Standard deviation of coefficient	Mean of node cut value t1 or (t1, t2)	Standard deviation of node cut value t1 or (t1, t2)
XEL.Open - t1	228	–0.050	0.939	68.672	0.176
WBNNS_score10.MO4 - t1	227	0.273	0.185	–4.054	0.341
WBNNS_score21.MO1 - t1	104	–2.25E-08	1.09E-08	–2.62E+07	1.10E+06
(HSI.PSY12 - t1)* (WBNNS_score10.MO4 - t2)	86	–0.013	0.010	(33.333, -3.777)	(0.000, 0.253)
(XEL.Open - t1)* (WBNPS_score11.MO2 - t2)	70	–6.22E-09	4.06E-09	(68.596, -5.49E+06)	(0.077, 1.73E+05)
(TWII.Dailychange.Lag8 - t1)* (XEL.Open - t2)	30	6.555	0.091	(-0.009, 68.583)	(3.80E-04, 0.057)
(S&P500.MO6 - t1)* (WBNNS_score21.MO1 - t2)	19	–7.06E-11	4.68E-11	(–543.300, –2.63E+07)	(0.000, 1.24E+06)

Table 9. Feature effect in the MARS model for the stock *DISCA*.

Feature name	Frequency	Mean of coefficient	Standard deviation of coefficient	Mean of node cut value t1 or (t1, t2)	Standard deviation of node cut value t1 or (t1, t2)
DISCA.Open - t1	228	–7.068	9.156	55.651	2.267
ABNS_score21_sum.MO1 - t1	48	–2.58E-09	1.53E-09	–1.76E+08	3.07E+06
(HSI.O.Lag1 - t1)* (DISCA.Open - t2)	89	6.25E-04	2.39E-04	(2.17E+04, 55.087)	(0.000, 2.267)
(DISCA.Open - t1)* (ABNS_score21_min.MO2 - t2)	52	3.09E-08	4.51E-09	(53.230, –2.78E+08)	(0.596, 0.000)
(DISCA.Dailychange.Lag6 - t1)* (ABNS_score21_sum.MO1 - t2)	21	–8.46E-08	9.93E-09	(–0.137, –1.75E+08)	(0.000, 3.11E+06)

A total of 449 terms selected by MARS model in the prediction of *DISCA* using the *ABNS*4-*WSS* feature group. Table 9 presents the terms that were selected at least 10 times. $DISCA.Open$, the open price of *DISCA* on a day, is the most frequently selected main factor under this model, totaling up to 228 times. The second most frequent feature is $ABNS\_score21\_sum.MO1$ that has been chosen 48 times. This feature takes the fluctuation of the stock price in two days and its transaction volume into consideration when calculating the Article-Based News Scores. The variable magnitude is at the level of 10^8. Hence, due to the mean coefficient being –2.58E-09, which is much smaller than the cut value –1.76E+08, this feature negatively affects the stock price prediction at least half of the time.

Besides, the combination of $HSI.O.Lag1$ and $DISCA.Open$ was picked 89 times, indicating an interaction effect between these two features on stock price and a potential relationship between them. As stated in Table 9, if the open price of HSI on a preceding day is larger than 21700, and if the open price of *DISCA* is more than 55.087, the product of these two feature values will have a positive effect on stock price's prediction.

7. CONCLUDING REMARKS

The proposed news polarity scores, article-based and word-based news scores, had helped the stock price' up/down prediction for more than 20 stocks out of the 100 stocks in the study. The main advantage of these features is that they don't require human labeling and are automatically defined by a function of price, transaction volume, and news word importance. The MARS model has higher prediction accuracy than the LASSO and LSTM models for most stocks. The MARS model with the feature group of word-based positive and negative scores increased about 8% accuracy in stock XEL price up/down prediction than with the basic and technical feature group.

Although the LSTM model is anticipated to be more potent in many stock price prediction studies, this is not the case here. We followed the parameter setting of the LSTM model from two papers [46, 47] in stock prediction. In only one stock, Procter (PG), the price up/down prediction by the LSTM model was better than the results of the LASSO and MARS models. One may explore other possible parameter settings for the LSTM model. The best parameter search will be very time-consuming and may depend on the target stocks.

Another critical benefit of the MARS model is the capability to provide the coefficients of the important features. Then the feature effect can be interpreted by the sign and magnitude of the coefficient. This interpretability of the model coefficient is not available in the LSTM model, which is a black-box recurrent neural network. The MARS model also identified the interaction effect between important features in the price' up/down prediction.

Our approach included many news articles that may not be directly or indirectly related to the selected 100 companies. The constructed word-level and sentence-level sentiment features could have many noises, which could impact the accuracy of the evaluation performance model. The proposed MARS models performed well in feature selection and prediction accuracy which did not suffer the noise from the sentiment features for the news in the overall evaluation compared to the results using the basic technical features. However, the effect of word-level and sentence-level sentiment features constructed from carefully selected news which mentions specific company names may be further explored.

REFERENCES

1. M. Roondiwala, H. Patel, and S. Varma, "Predicting stock prices using LSTM," *International Journal of Science and Research*, Vol. 6, 2017, pp. 1754-1756.
2. S. M. Idrees, M. A. Alam, and P. Agarwal, "A prediction approach for stock market volatility based on time series data," *IEEE Access*, Vol. 7, 2019, pp. 17287-17298.

3. W. Long, Z. Lu, and L. Cui, "Deep learning-based feature engineering for stock price movement prediction," *Knowledge-Based Systems*, Vol. 164, 2019, pp. 163-173.
4. H. Y. Kim and C. H. Won, "Forecasting the volatility of stock price index: A hybrid model integrating lstm with multiple garch-type models," *Expert Systems with Applications*, Vol. 103, 2018, pp. 25-37.
5. S. Mahfooz, I. Ali, and M. N. Khan, "Improving stock trend prediction using lstm neural network trained on a complex trading strategy," *International Journal for Research in Applied Science and Engineering Technology*, Vol. 10, 2022, pp. 4361-4371.
6. K. Alkhatib, H. Khazaleh, H. A. Alkhazaleh, A. R. Alsoud, and L. Abualigah, "A new stock price forecasting method using active deep learning approach," *Journal of Open Innovation: Technology, Market, and Complexity*, Vol. 8, 2022, p. 96.
7. J.-H. Wang and H.-W. Liu, "A deep fusion model combining news content and historical prices for stock trend prediction," in *Proceedings of Annual Conference of the Japanese Society for Artificial Intelligence*, 2021, pp. 13-26.
8. W. Gu, L. Zhang, H. Xi, and S. Zheng, "Stock prediction based on news text analysis," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol. 25, 2021, pp. 581-591.
9. S. Wu, Y. Liu, Z. Zou, and T.-H. Weng, "S_i_lstm: stock price prediction based on multiple data sources and sentiment analysis," *Connection Science*, Vol. 34, 2022, pp. 44-62.
10. S. Mohan, S. Mullapudi, S. Sammeta, P. Vijayvergia, and D. C. Anastasiu, "Stock price prediction using news sentiment analysis," in *Proceedings of IEEE 5th International Conference on Big Data Computing Service and Applications*, 2019, pp. 205-208.
11. A. Agarwal, "Sentiment analysis of financial news," in *Proceedings of IEEE 12th International Conference on Computational Intelligence and Communication Networks*, 2020, pp. 312-315.
12. J. Fan, L. Xue, and Y. Zhou, "How much can machines learn finance from Chinese text data?" *SSRN*, 2021, No. 3765862.
13. L. Nemes and A. Kiss, "Prediction of stock values changes using sentiment analysis of stock news headlines," *Journal of Information and Telecommunication*, Vol. 5, 2021, pp. 375-394.
14. G. J. Sawale and M. K. Rawat, "Stock market prediction using sentiment analysis and machine learning approach," in *Proceedings of the 4th International Conference on Smart Systems and Inventive Technology*, 2022, pp. 1-6.
15. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *arXiv Preprint*, 2018, arXiv: 1810.04805.
16. Y. Liu and M. Lapata, "Text summarization with pretrained encoders," in *Proceedings of International Conference on Empirical Methods in Natural Language Processing*, 2019, pp. 3730-3740.
17. H. Jwa, D. Oh, K. Park, J. M. Kang, and H. Lim, "exBAKE: Automatic fake news detection model based on bidirectional encoder representations from transformers (BERT)," *Applied Sciences*, Vol. 9, 2019, pp. 35-65.

18. M. Hu, S. Zhao, H. Guo, R. Cheng, and Z. Su, "Learning to detect opinion snippet for aspect-based sentiment analysis," *arXiv Preprint*, 2019, arXiv:1909.11297.
19. F. Chen, Z. Yuan, and Y. Huang, "Multi-source data fusion for aspect-level sentiment classification," *Knowledge-Based Systems*, Vol. 187, 2020, p. 104831.
20. J. Z. G. Hiew, X. Huang, H. Mou, D. Li, Q. Wu, and Y. Xu, "Bert-based financial sentiment index and lstm-based stock return predictability," *arXiv Preprint*, 2019, arXiv:1906.09024.
21. D. Araci, "FinBERT: Financial sentiment analysis with pre-trained language models," *arXiv Preprint*, 2019, arXiv:1908.10063.
22. S. Yu, Y. Chen, and H. Zaidi, "A financial service chatbot based on deep bidirectional transformers," *arXiv Preprint*, 2020, arXiv:2003.04987.
23. S. Anbaee Farimani, M. Vafaei Jahan, A. Milani Fard, and S. R. K. Tabbakh, "Investigating the informativeness of technical indicators and news sentiment in financial market price prediction," *Knowledge-Based Systems*, Vol. 247, 2022, p. 108742.
24. Z. Yu, M. D. Wang, X. Wei, and J. Lou, "News credibility and influence within the financial markets," *Journal of Behavioral Finance*, Vol. 24, 2023, pp. 238-257.
25. N. Das, B. Sadhukhan, T. Chatterjee, and S. Chakrabarti, "Effect of public sentiment on stock market movement prediction during the covid-19 outbreak," *Social Network Analysis and Mining*, Vol. 12, 2022, p. 92.
26. M. S. Sivri, A. Ustundag, and B. S. Korkmaz, "Ensemble learning based stock market prediction enhanced with sentiment analysis," in *Proceedings of International Conference on Intelligent and Fuzzy Techniques for Emerging Conditions and Digital Transformation*, Vol. 2, 2022, pp. 446-454.
27. C.-H. Chen, P.-Y. Chen, and J. C.-W. Lin, "An ensemble classifier for stock trend prediction using sentence-level chinese news sentiment and technical indicators," *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 7, 2022, pp. 53-64.
28. Y. Li and Y. Pan, "A novel ensemble deep learning model for stock prediction based on stock prices and news," *International Journal of Data Science and Analytics*, 2022, pp. 1-11.
29. M. N. Ashtiani and B. Raahmei, "News-based intelligent prediction of financial markets using text mining and machine learning: A systematic literature review," *Expert Systems with Applications*, Vol. 217, 2023, p. 119509.
30. B. Meyer, M. Bikdash, and X. Dai, "Fine-grained financial news sentiment analysis," in *Proceedings of IEEE SoutheastCon*, 2017, pp. 1-8.
31. S. Krishnamoorthy, "Sentiment analysis of financial news articles using performance indicators," *Knowledge and Information Systems*, Vol. 56, 2018, pp. 373-394.
32. A. Yadav, C. Jha, A. Sharan, and V. Vaish, "Sentiment analysis of financial news using unsupervised and supervised approach," in *Proceedings of International Conference on Pattern Recognition and Machine Intelligence*, 2019, pp. 311-319.
33. J.-H. Wang and S. Huang, "Improving sentiment classification from high volatility financial news," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, 2018, pp. 1790-1797.
34. T. Jordan and H. Elgazzar, "Stock market prediction using text-based machine learning," in *Proceedings of IEEE International IOT, Electronics and Mechatronics Conference*, 2020, pp. 1-5.

35. A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Springer, 2012, Chapter 4, pp. 36-45.
36. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016, Chapter 10, pp. 373-420.
37. J. Gareth, W. Daniela, H. Trevor, and T. Robert, *An Introduction to Statistical Learning: With Applications in R*, Spinger, Berlin, 2013.
38. J. H. Friedman, "Multivariate adaptive regression splines," *The Annals of Statistics*, Vol. 19, 1991, pp. 1-67.
39. T. Hastie, R. Tibshirani, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer, 2009, Vol. 2.
40. J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques," *Expert Systems with Applications*, Vol. 42, 2015, pp. 259-268.
41. Y. Shynkevich, T. M. McGinnity, S. A. Coleman, A. Belatreche, and Y. Li, "Forecasting price movements using technical indicators: Investigating the impact of varying input window length," *Neurocomputing*, Vol. 264, 2017, pp. 71-88.
42. T. Loughran and B. McDonald, "When is a liability not a liability? Textual analysis, dictionaries, and 10-ks," *The Journal of Finance*, Vol. 66, 2011, pp. 35-65.
43. N. Oliveira, P. Cortez, and N. Areal, "The impact of microblogging data for stock market prediction: Using twitter to predict returns, volatility, trading volume and survey sentiment indices," *Expert Systems with applications*, Vol. 73, 2017, pp. 125-144.
44. J. Zhang, Y. Lai, and J. Lin, "The day-of-the-week effects of stock markets in different countries," *Finance Research Letters*, Vol. 20, 2017, pp. 47-62.
45. R. Ren, D. D. Wu, and T. Liu, "Forecasting stock market movement direction using sentiment analysis and support vector machine," *IEEE Systems Journal*, 2018, pp. 1-11.
46. J. Rasheed, A. Jamil, A. A. Hameed, M. Ilyas, A. Özyavaş, and N. Ajlouni, "Improving stock prediction accuracy using cnn and lstm," in *Proceedings of IEEE International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy*, 2020, pp. 1-5.
47. T. B. Shahi, A. Shrestha, A. Neupane, and W. Guo, "Stock price forecasting with deep learning: A comparative study," *Mathematics*, Vol. 8, 2020, p. 1441.
48. T. H. Aldhyani and A. Alzahrani, "Framework for predicting and modeling stock market prices based on deep learning algorithms," *Electronics*, Vol. 11, 2022, p. 3149.

Shuen-Lin Jeng received the MS degree in Applied Mathematics from Fujen University, Taiwan in 1989, and Ph.D. in Statistics from Iowa State University, USA in 1998. He is currently an Associate Professor in the Department of Statistics and Institute of Data Science at National Cheng Kung University, Taiwan. His research interests include product reliability, software reliability, statistical computing, data mining, and machine learning.

Chiau-Hsuan Liu graduated with a bachelor's degree in Statistics from Tamkang University, Taiwan, and received a master's degree from National Cheng Kung University, Taiwan.

Chin Min Guan graduated with a bachelor's degree in Economics from University of Malaya in Malaysia and currently is pursuing for a Master of Statistics at National Cheng Kung University in Taiwan.

APPENDIX

Table A.1. List of 100 selected individual stocks.

Stock	**Symbol**	**Stock**	Symbol	**Stock**	Symbol
Altria	MO	Freeport-Mcmoran	FCX	Pfizer	PFE
Amazon	AMZN	Gap	GPS	Procter	PG
American International Group	AIG	General Electric	GE	Progressive	PGR
Analog Devices	ADI	General Motors	GM	Qualcomm	QCOM
Apache	APA	Gilead	GILD	Regions Financial	RF
Apple	AAPL	Google	GOOG	Ross Stores	ROST
Applied Materials	AMAT	Google	GOOGL	Salesforce	CRM
At&t	T	Halliburton	HAL	Schlumberger	SLB
Boeing	BA	Host Hotels	HST	Southern	SO
Boston Scientific	BSX	Huntington	HBAN	Southwestern	SWN
Bristol-Myers	BMY	Ibm	IBM	Spdr	SPY
Campbell	CPB	Intel	INTC	Starbucks	SBUX
Carnival	CCL	Interpublic	IPG	Sysco	SYY
Charles Schwab	SCHW	Johnson	JNJ	T.j. Maxx	TJX
Chevron	CVX	Johnson Controls	JCI	Target	TGT
Cisco	CSCO	Jpmorgan	JPM	Technipfmc	FTI
Citigroup	C	Keycorp	KEY	Texas Instruments	TXN
Cleveland-Cliffs	CLF	Kimco Realty	KIM	United States Steel	X
Coca-Cola	KO	Kohl'S	KSS	Unitedhealth	UNH
Comcast	CMCSA	Kroger	KR	Us Bancorp	USB
Conocophillips	COP	Marathon	MRO	Verizon	VZ
Corning	GLW	Merck	MRK	Vf	VFC
Csx	CSX	Meta	FB	Visa	V
Cvs	CVS	Metlife	MET	Walmart	WMT
Deere	DE	Microchip	MCHP	Wells Fargo	WFC
Devon	DVN	Micron	MU	Williams	WMB
Discovery	DISCA	Microsoft	MSFT	Xcel	XEL
Disney	DIS	Netflix	NFLX		
Ebay	EBAY	Nike	NKE		

Table A.2. Extra words for positive and negative wordlists in ALMD.

Positive Words	bull, advance, advanced, boomed, booms, climb, climbed, climbing, climbs, grew, grow, growing, grown, heavy, increase, increased, increasing, jump, jumped, jumping, jumps, raise, raised, raises, raising, rallied, rallies, rally, rallying, rise, risen, rises, rising, rose, skyrocket, skyrocketed, skyrocketing, soar, soared, soaring, surge, surged, surges, surging
Negative Words	bear, bearish, depreciate, depreciated, depreciating, dive, dives, diving, dove, down, drop, dropping, drops, fall, fallen, falling, falls, fell, low, lower, lowest, lows, nosedive, plummet, plummeted, plummeting, plunge, plunged, plunges, plunging, reduced, rout, routed, routing, routs, sank, selloff, shrank, shrink, shrinken, shrinking, shrinks, shrunk, sink, sinking, sinks, slid, slidden, slide, slides, sliding, slip, slipped, slipping, slips, slump, slumped, slumping, slumps, sunk, sunken, tumble, tumbled, tumbles, tumbling, wound, wounded, wounding, wounds

Table A.3. Wordlists for specific categories.

Category	**Wordlists**
Disaster	blizzard, cyclone, earthquake, flood, hurricane, mudslide,
	natural disaster, storm, typhoon
Publish	launch, publish, present, release, releasing, rollout, showcase,
	unveil
Acquisition	acquire, acquiring, acquisition, bargain, buyout,
	hostile takeover, merger, merging, purchase
Lawsuit	defendant, extortionist, infringe, injunct, lawsuit, litigious
President	bush, inauguration, obama, president, trump, white house
Bear	bear, bearish, decline, declining, descend, drop, fall, fell,
	plunge, plunging, slip, slump, tumble, tumbling
Bull	bull, bullish, climb, gain, increase, increasing, jump, rally,
	rallied, rallies, rebound, rise, rose, rising, soar, surge, surging
Bankrupt	bankrupt, bankruptcy, clash, conflict, dispute, feud,
	insolvency, negotiation, spat, tussle
Highlights	trade war, trump, tariff, tax