From Time Series to Images: Revolutionizing Stock Market Predictions with Convolutional Deep Neural Networks

TATANE Khalid[1], SAHIB Mohamed Rida[2], ZAKI Taher[3]
National School of Applied Sciences, ESTIDMA, Ibn Zohr University, Agadir, Morocco[1]
Faculty of Applied Sciences, IMIS Laboratory, Ibn Zohr University, Agadir, Morocco[2, 3]

***Abstract*—Predicting the trend of stock prices is a hard task due to numerous factors and prerequisites that can affect price movement in a specific direction. Various strategies have been proposed to extract relevant features of stock data, which is crucial for this domain. Due to its powerful data processing capabilities, deep learning has demonstrated remarkable results in the financial field among modern tools. This research suggests a convolutional deep neural network model that utilizes a 2D-CNN to process and classify images. The process for creating images involves transforming the top technical indicators from a financial time series, each calculated for 21 different day periods, to create images of specific sizes. The images are labeled Sell, Hold, or Buy based on the original trading data. Compared to the Long Short Time Memory Model and to the one-dimensional Convolutional Neural Network and the model exhibits the best performance.**

Keywords—Technical indicators; convolutional neural networks; stock trend forecasting; deep learning

I. Introduction

Predicting stock trends has been the subject of intensive research for decades, making it a challenging problem [1] [2]. Developing accurate prediction models is challenging due to the many factors that can impact price changes. In recent years, machine learning and text mining models have shown promising results in multiple applications, including stock market prediction [3]. However, deep learning techniques [4] [5] [6] have become a potential alternative due to the limited accuracy of these models. The use of deep learning in financial studies is becoming increasingly popular as its potential is demonstrated in various applications. Despite this, deep neural networks are still not widely used for stock market prediction. The main reason for this is that processing financial data is difficult and the prediction task is complex. To tackle this issue, a unique strategy is suggested that transforms financial data into image-like representations and utilizes a deep convolutional neural network (CNN) to forecast stock trends.

This paper provides a thorough explanation of the approach, encompassing the technical aspects of the model and the data pre-processing procedures. The proposal consists of converting financial data into 2D images and then feeding them into a CNN for training. In particular, 21 technical indicators for a 21-day period are computed and the 225 most pertinent features are selected, which are transformed into 15x15 images. A chronological sequence of these images is used as input to the CNN model.

The approach's effectiveness in predicting stock trends is demonstrated by the presentation of its results. The model's performance is evaluated through various metrics, such as precision, confusion matrix, F1-score, and recall. Moreover, the evaluation of the outcomes of the proposed approach against other conventional machine learning models, proving that the deep learning approach is superior. In summary, the proposed approach is a promising solution for stock market prediction, delivering a more precise and efficient alternative to traditional machine learning models.

The rest of the paper is structured as follows: Section II describes the related work; Section III details the background and the followed methodology in Section IV. Section V covers the results, and Section VI presents general conclusions. Finally future works is given in Section VII.

II. Related Work

Different approaches have been investigated in the field of predicting stock prices to improve the accuracy of forecasts. [7] It suggests employing ARIMA models, for crop price predictions. It recommended using feature selection techniques to enhance their predictive abilities. The research in [8] developed an HNN-based forecasting approach that utilized headline information as well as ARIMA model predictions to drive sales estimation and business strategic decision making for the publishing industry.

For forecasting stock price indices, [9] came up with a method integrating ARIMA model and backward propagation neural network. The resulting statistics were compared against outputs attained via back propagation neural network and a single ARIMA model approach, respectively. Similar to [10], a model was designed to measure time periods optimal for the prediction of the market price of securities within various industries in 2019. The results emphasized the importance of using large datasets while training models to predict with more precision. A hybrid CNN-TLSTM model for predicting the USD/CNF exchange rate by [11] obtained low mean absolute percentage error (MAPE) and mean squared error (MSE) showing its ability of dealing with complex nonlinear data According to [12], CNN-LSTM proved efficient as a quantitative trading strategy analyzer in the stock market capable of forecasting future movement of stocks and revealing their patterns. The study in [13] proposed the CNN-BiLSTM-AM technique for prediction of the following days' closing stock prices.

The study revealed that this model achieved greater predictive accuracy as well as performance when compared to other models using CNNs coupled with an attention mechanism. A study by [14] used an LSTM recurrent neural network to predict stock market trends, highlighting the LSTM model's capacity for accurately predicting stock prices.

III. Background

A. Background

Deep learning has become one of the most relevant mechanisms in the domain of machine learning and has been used generally in image recognition and natural language processing [15]. The most commonly used deep learning algorithms are convolutional neural networks (CNN) for feature extraction [16], recurrent neural networks (RNN) for mining time series data [17], etc. Therefore, for the purpose of making full use of their benefits, a CNN model is utilized to benefit from its power in feature extraction, which will be presented as follows.

1) Convolutional neural networks: Undoubtedly, convolutional neural networks (CNNs) (see Fig. 1) have established themselves as the most prevalent deep learning model, showcasing various network structures encompassing 1D, 2D, and 3D CNNs [18]. The 1D CNN specializes in processing sequential data [19], while the 2D counterpart excels in image and text recognition. Conversely, the 3D CNN takes the next stage in the identification of video data [20]. A standard convolutional neural network comprises an input layer, an output layer, and multiple hidden layers. These concealed layers typically house numerous convolutional layers that execute convolution through a dot product, employing RELU as the activation function. Following these convolutional layers are additional operations such as pooling, flattening, and fully connected layers [21].

2) Convolutional layer: The convolutional rlayer, a cornerstone of convolutional neural networks, shouldes the bulk of computational burden. It utilizes a set of spatially-restricted filters, each spanning the entirety of the input volume's depth. During the forward pass, these filters are slid across the input's width and height, computing the dot product between their entries and the input at each position. This process, akin to fingerprint identification, generates a 2D activation map, revealing the filter's response at every location. The network learns filters that activate to specific visual features, each producing a distinct map. These maps are then stacked along the depth dimension, forming the layer's output volume.

3) Pooling layer: Convolutional neural networks frequently employ pooling layers interspersed between convolutional layers. These interludes serve to shrink the representation's size, thereby reducing both the network's parameter count and computational demands, while simultaneously mitigating overfitting. The pooling layer independently processes each depth slice of the input, resizing it through a MAX operation. A widely used variant employs 2x2 filters with a stride of two, effectively down sampling each depth slice by a factor of two in both width and height dimensions. This translates to each MAX operation extracting the highest value from a quartet of numbers, leaving the depth dimension unaltered.

4) Fully connected layer: Fully connected layers establish connections between every neuron in one layer and every neuron in another layer. This architecture is similar to that of a traditional multi-layer perceptron neural network (MLP). In a CNN, the flattened matrix is passed through a fully connected layer to enable classification.

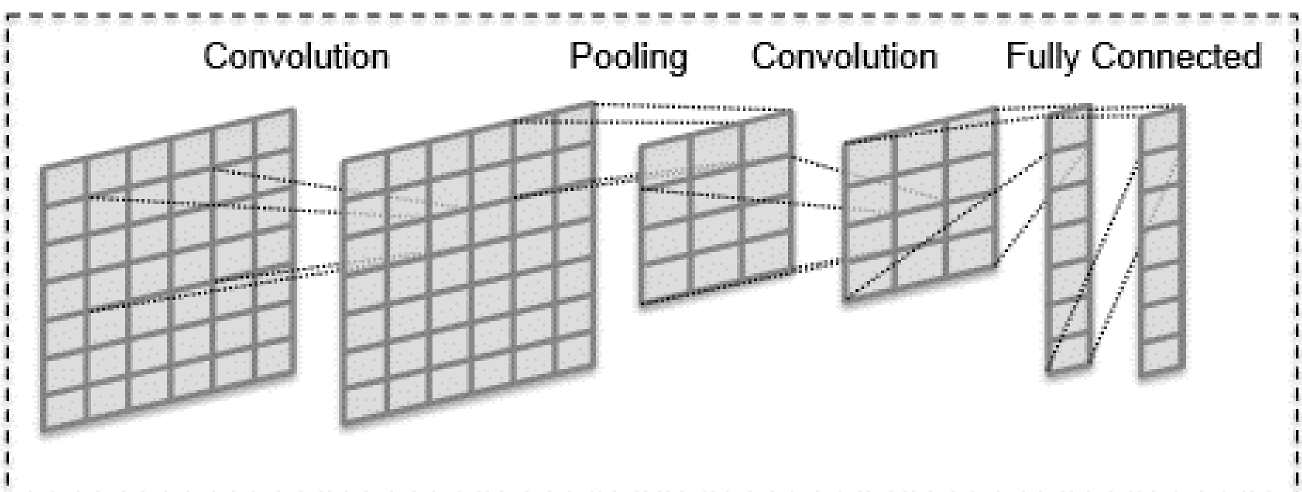

Fig. 1. CNN model architecture.

IV. Methodology

The proposal involves a model that employs Convolutional Neural Networks (CNNs) to detect favorable buying and selling points in stock prices. The model utilizes the top 15 technical indicators out of 21, using varying time intervals to generate images. As mentioned in (see Fig. 4), this approach encompasses four key stages: feature engineering, feature selection, data labeling, class imbalance handling, and image creation. The objective is to identify the most appropriate buy, sell, and hold positions in the time series of related stock prices.

1) Feature engineering: The dataset comprises features such as Date, Open, High, Low, Close, Adj Close, and Volume. To enhance the dataset's informational value, it's necessary to calculate 21 technical indicators for each day. These indicators include: RSI, William %R, MFI, MACD, PPO, ROC, CMFI, CMO, SMA, EMA, WMA, HMA, Triple EMA, CCI, DPO, KST, EOM, IBR, DMI, PSAR, and Volume values for intervals ranging from 6 to 27 days. These indicators are primarily categorized as momentum or oscillator types.

2) Feature selection: The proposed method improved the model's performance by applying a feature selection technique to the technical indicators. It selected 225 high-quality features using a univariate method that chooses variables with a strong relationship to the target outcome. It used two methods for feature selection: ANOVA F-value focusing on statistically significant differences between classes, and mutual information capturing non-linear relationships. It identified the features that both methods agreed on, then sorted the index list from the intersection of ANOVA F-value and mutual information to group similar indicators in the image.

3) Labeling data: The target was labeled by following a methodology [22] that used an 11-day window on the closing price. It checked if the middle number in the window was the highest, and if so, labeled the last day (11th day) as SELL. Otherwise, if the middle number was the lowest, labeled the last day as BUY. If neither condition was satisfied, labeled the last day as HOLD. The window was then slid and the process was repeated. This approach aimed to identify buying and selling opportunities at low and high points within each 11-day interval.

4) Handling class imbalance: The target was labeled and the data was found to be severely imbalanced. The Hold instances outnumbered the Buy and Sell instances by a large margin. The labeling algorithm used produced a relatively high number of Buy and Sell instances. Moreover, the Hold class was difficult to classify, making it hard for the model to learn meaningful insights. To solve this problem, the Sample Weights technique was adopted. This technique involved giving more weight to specific samples, which was a successful method for dealing with class imbalance.

5) Image creation: After completing all the preparatory steps, including downloading the dataset, calculating the technical indicators, performing feature selection, labeling the target, and normalizing the data, tabular data containing 225 features for each day was obtained. Subsequently, this data was converted into an image-like format. Sample 15x15 pixel images generated during this image creation process are presented in (see Fig. 2) to illustrate the visual representation of the data.

Fig. 2. Sample of generated images.

6) CNN model architerture: The proposed model consists of nine layers. The first layer, the input layer, receives images of size 15x15x3. The second layer is a convolutional layer with 25 filters, each of size 2x2. This layer is followed by a dropout layer that randomly drops out nodes during training with a 0.15% probability. Next, another convolutional layer is added, featuring 12 filters of the same size as the previous layer, stacked to a dropout layer with the same percentage (0.15%). Comes next the flatten layer, a dense layer with 100 neurons, a dropout layer with 0.15%, and the last layer which is the output layer with three neurons The model's output layer consists of three neurons, which determine the final action. The neural network uses the Adam optimizer, which is an adaptive gradient-based method that adjusts the learning rate for each parameter based on the previous gradients. It uses a learning rate of 0.001, which is a common value for the optimizer. The neural network trains for 3000 epochs, which means it sees the training data 3000 times with a batch size of 64 sample. EarlyStopping and ReduceLROnPlateau callbacks were utilized during model training. The first one stops the training when a monitored quantity (such as the validation loss) stops improving, and the last one reduces the learning rate when a monitored quantity (such as the validation loss) stops improving. (see Fig. 3) shows a recap for the CNN architecture.

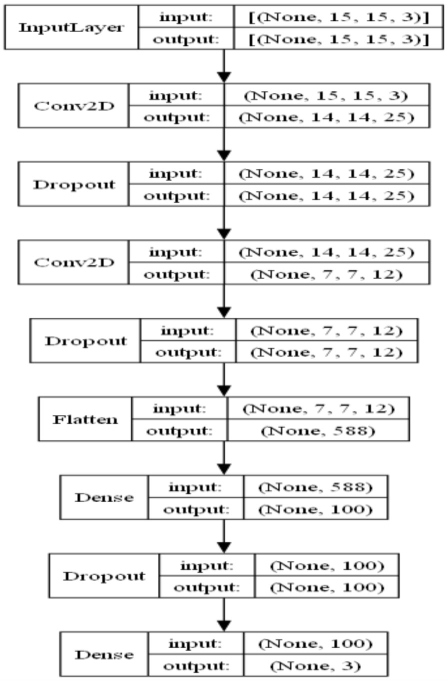

Fig. 3. Model.

a) EarlyStopping: Choosing the optimal number of training epochs is critical when training a neural network. Employing too many epochs can lead to overfitting, where the model becomes overly reliant on the training data and loses generalizability. Conversely, using too few epochs can result in underfitting, where the model fails to grasp the underlying patterns in the data. To address this dilemma, a technique called "early stopping" can be implemented. This involves training the model for a potentially large number of epochs, but proactively stopping the training process once the model's performance on a separate validation dataset ceases to improve.

b) ReduceLROnPlateau: During training, deep neural networks leverage the stochastic gradient descent (SGD) algorithm. This optimization technique utilizes randomly sampled data points from the training set to estimate the error gradient of the model's current state. Subsequently, the backpropagation algorithm is employed to update the model's weights based on this gradient information.

The learning rate governs how rapidly the model adapts to the problem it's facing. Smaller learning rates necessitate more training epochs because the weights are updated with smaller increments in each iteration. Conversely, larger learning rates propel faster adjustments, demanding fewer epochs.

However, selecting an inappropriate learning rate can lead to complications. A learning rate that's too high can cause the model to converge rapidly to a suboptimal solution, missing the optimal one. Conversely, a learning rate that's too low can cause the training process to become sluggish and potentially stall indefinitely. Consequently, selecting the optimal learning rate is crucial for effective deep neural network training.

7) Performance evaluation: The proposed model was assessed mainly by its computational performance, especially its accuracy in classifying buy, hold, and sell categories. These predicted labels guided buy, sell, and hold decisions based on stock prices. The model was trained and tested on WALMART stocks, using train, validation, and test splits. The F1-score was the main metric for performance evaluation on the test data, along with other metrics such as the confusion matrix, weighted F1 score.

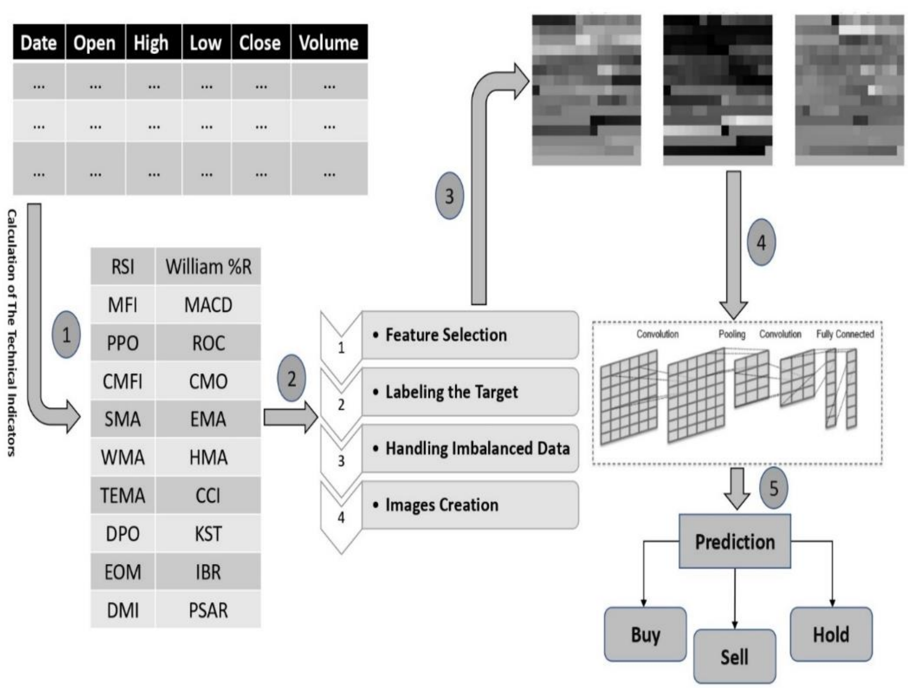

Fig. 4. Proposed method.

V. RESULTS

On WALMART test data the model gave the following result:

The model's predictive abilities were thoroughly evaluated and analyzed. Table I presents the confusion matrix for the WALMART test data, while Table II offers a deeper analysis based on these results. Notably, the precision values for "Buy" and "Sell" classes surpass those for "Hold," indicating the model's strength in capturing key buy and sell points. However, false entry and exit points arose due to the inherent rarity of these instances compared to "Hold" periods. This posed a challenge for the neural network: accurately detecting rare entry and exit points while maintaining the overall "Hold" distribution. As a consequence, the model generated false alarms for non-existent entry and exit points to capture the majority of true ones. Additionally, the "Hold" points lacked the clear definition of "Buy" and "Sell" points, leading to some confusion, particularly near peaks and valleys within sliding windows.

TABLE I. CONFUSION MATRIX OF TEST DATA (WALMART)

Predicted			
Actual	Buy	Sell	Hold
Buy	44	0	20
Sell	0	42	19
Hold	122	140	613

The F1-weighted score was used to comprehensively assess classification performance, which considered class weights for a more refined evaluation. The sample count of each class determined its F1 score weight. Multi-Layer Perceptron (MLP), Long Short-Term Memory (LSTM), and Convolutional Neural Network (CNN) methods were the baselines for stock market prediction, compared with the proposed model. Table III displayed the comparative prediction results.

TABLE II. EVALUATION OF TEST DATA (WALMART)

Total Accuracy: 0.699			
	Buy	Sell	Hold
Recall	0.6875	0.6885	0.7000
Precision	0.2650	0.2307	0.9401
F1-Score	0.3825	0.3455	0.8024
Weighted F1	0.7480		

TABLE III. THE AVERAGE OF F1-SCORE OF TEST DATA (WALMART) ON DIFFERENT MODEL

Model	AVG F1-Score
CNN	0.44
LSTM	0.57
MLP	0.45
CNN 2D	**0.74**

VI. CONCLUSION

In this study, a 2D Deep Convolutional Neural Network model was developed utilizing financial data and technical indicators to predict optimal buy and sell points. The model analyzed financial time series data, converted it into 2D images, and identified "Buy," "Sell," and "Hold" signals for profitable trades. Using WALMART stock prices as the primary data source, the model outperformed other models with the same dataset.

While the model's performance is encouraging, there's still room for improvement. Experimenting with various techniques, such as hyperparameter tuning, exploring different image sizes, alternative target labeling strategies, and utilizing additional technical indicators, could yield further gains. This research also contributes to the growing trend of applying 2D CNNs to non-image data, particularly in time series forecasting.

VII. FUTURE WORKS

The model was trained on the historical data of WALMART stocks. For future work, more stock data from different sources will be incorporated to enhance the deep learning models. Moreover, other types of deep neural networks and their combinations will be explored to achieve better performance. Furthermore, since the news has an impact on the stock price trends, the model will be augmented with sentiment analysis features to increase its accuracy. Additionally, alternative labeling methods will be experimented to improve the model outcome.

REFERENCES

[1] Kumbure, Mahinda Mailagaha, Christoph Lohrmann, Pasi Luukka, and Jari Porras. (2022). “Machine Learning Techniques and Data for Stock Market Forecasting: A Literature Review.” Expert Systems with Applications 197: 116659.

[2] Guo, Y., He, F., Liang, C., & Ma, F. (2022). Oil price volatility predictability: new evidence from a scaled PCA approach. Energy Economics, 105, 105714.

[3] Henríquez, J., & Kristjanpoller, W. (2019). A combined Independent Component Analysis–Neural Network model for forecasting exchange rate variation. Applied Soft Computing, 83, 105654.

[4] Baek, Y., & Kim, H. Y. (2018). ModAugNet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module and a prediction LSTM module. Expert Systems with Applications, 113, 457-480.

[5] Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PloS one, 12(7), e0180944.

[6] Ma, C., & Yan, S. (2022). Deep learning in the Chinese stock market: The role of technical indicators. Finance Research Letters, 49, 103025.

[7] Shao, Y. E., & Dai, J.-T. (2018). Integrated Feature Selection of ARIMA with Computational Intelligence Approaches for Food Crop Price Prediction. 2018, 1-17. https://doi.org/10.1155/2018/1910520.

[8] Omar, H., Hoang, V. H., & Liu, D. R. (2016). A hybrid neural network model for sales forecasting based on ARIMA and search popularity of article titles. Computational Intelligence and Neuroscience, 2016. https://doi.org/10.1155/2016/9656453

[9] Du, Y. (2018). Application and analysis of forecasting stock price index based on combination of ARIMA model and BP neural network. In Proceedings of the 30th Chinese control and decision conference, CCDC 2018 (pp. 2854–2857). https://doi.org/10.1109/CCDC.2018.8407611

[10] Ghosh, A., Bose, S., Maji, G., Debnath, N. C., & Sen, S. (2019). Stock price prediction using lstm on Indian share market. EPiC Series in Computing, 63, 101–110. https://doi.org/10.29007/qgcz

[11] Wang, J., Wang, X., Li, J., & Wang, H. (2021). A prediction model of CNN-TLSTM for USD/CNY exchange rate prediction. IEEE Access, 9, 73346–73354. https://doi.org/10.1109/ACCESS.2021.3080459

[12] Liu, S., Zhang, C., & Ma, J. (2017). CNN-LSTM neural network model for quantitative strategy analysis in stock markets. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10635(LNCS), 198–206. https://doi.org/10.1007/978-3-319-70096-0_21

[13] Lu, W., Li, J., Li, Y., Sun, A., & Wang, J. (2020). A CNN-LSTM-based model to forecast stock prices. Complexity, 2020. https://doi.org/10.1155/2020/6622927 Lussange, J., Palminteri, S., Bourgeois-Gironde, S., & Gutkin, B. (2019). Mesoscale impact of trader psychology on stock markets: A multi-agent AI approach.

[14] Moghar, A., & Hamiche, M. (2020). Stock market prediction using LSTM recurrent neural network. Procedia Computer Science, 170, 1168–1173. https://doi.org/10.1016/j. procs.2020.03.049

[15] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," nature, vol. 521, no. 7553, pp. 436–444, 2015

[16] Amziane, A., Losson, O., Mathon, B., & Macaire, L. (2023). MSFA-Net: A convolutional neural network based on multispectral filter arrays for texture feature extraction. Pattern Recognition Letters, 168, 93-99.

[17] Ullah, F., Bilal, M., & Yoon, S. K. (2023). Intelligent time-series forecasting framework for non-linear dynamic workload and resource prediction in cloud. Computer Networks, 225, 109653.

[18] J. Zhao, X. Mao, and L. Chen, "Speech emotion recognition using deep 1d & 2d cnn lstm networks," Biomedical Signal Processing and Control, vol. 47, pp. 312–323, 2019

[19] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, "Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks," Journal of Sound and Vibration, vol. 388, pp. 154–170, 2017.

[20] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," IEEE transactions on medical imaging, vol. 35, no. 5, pp. 1285–1298, 2016.

[21] Wikipedia contributors, "Convolutional neural network — Wikipedia, the free encyclopedia," https://en.wikipedia.org/w/index.php?title=Convolutional_neural_network&oldid=962591055, 2023.

[22] [Sezer, O.B., Ozbayoglu, A.M., 2018. Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach. Appl. Soft Computing. 70, 525-538.

APPENDIX

1) Relative Strength Index (RSI): The Relative Strength Index (RSI), conceived by J.W. Wilder in 1978, quantifies the relative intensity of upward price movements against downward ones. This technical indicator, employed by traders to assess trend strength and identify potential reversals, offers valuable insights into momentum and potential trading signals. As price fluctuations occur, RSI values oscillate between 0% and 100%, signifying whether prices reside in the overbought (above 70%) or oversold (below 30%) territory. Equation for calculating the RSI value is provided in Eq. (1):

$$RSI = 100 - \frac{100}{1+(averagegain/averageloss)} \tag{1}$$

2) Williams %R: The Williams %R indicator, a stalwart among momentum-based technical tools, assists traders in identifying overbought and oversold conditions in stock prices. Its domain spans a spectrum from -100 to 0, serving as a window into market sentiment. A well-established interpretation of Williams %R values posits those readings below -80 signal an "oversold" market, ripe for potential upward reversals. Conversely, values exceeding -20 suggest an "overbought" market, susceptible to downward corrections. Eq. (2) details the calculation of Williams %R.

$$R = \frac{\max(high)-close}{\max(high)-\min(low)} * -100 \tag{2}$$

3) Money Flow Index (MFI): The Money Flow Index (MFI), a technical analysis stalwart, leverages both price and volume data to gauge overbought and oversold conditions in an asset, pinpointing potential trend reversals. Its values range from 0 to 100, offering a window into market sentiment. Unlike the Relative Strength Index (RSI), which focuses solely on price, the MFI encompasses both price and volume information, painting a more complete picture of market sentiment. Eq. (3) to Eq. (5) show the calculation of MFI:

$$RMI = TP \times Volume \tag{3}$$

$$MFR = \frac{14\,PeriodPositiveMoneyFlow}{14\,PeriodNegativeMoneyFlow} \tag{4}$$

$$MFI = 100 - \frac{100}{1+MFR} \tag{5}$$

4) Moving Average Convergence and Divergence (MACD): The Moving Average Convergence Divergence (MACD) indicator is a technical analysis tool that gauges the momentum and direction of stock prices. It measures the relationship between two moving averages of a security's price to identify potential turning points in the market.

When the MACD line crosses above the signal line in an upward direction, it suggests that bullish momentum is increasing, and stock prices are likely to rise. Conversely, if the MACD line crosses below the signal line in a downward direction, it indicates that bearish momentum is gaining strength, and stock prices are likely to fall.

The MACD indicator is a versatile tool that can be used to identify potential buy and sell signals, assess the strength of a trend, and anticipate trend reversals. However, it's important to note that the MACD indicator should not be used in isolation and should be considered in conjunction with other technical indicators and fundamental analysis. Eq. (6) and Eq. (7) show the calculations of MACD and Signal Lines:

$$MACD_{Line} = EMA_{12} - EMA_{26} \tag{6}$$

$$Signal_{Line} = EMA_9(MACD) \tag{7}$$

5) Percentage Price Oscillator (PPO): While the Percentage Price Oscillator (PPO) might appear similar to the MACD, a closer look reveals a subtle difference. While both rely on moving averages to assess momentum and trend direction, their calculation methodologies diverge. Eq. (8) and Eq. (9) illustrate the specific computations employed by the PPO to arrive at its main line and signal line:

$$PPO = \frac{EMA_{12} - EMA_{26}}{EMA_{26}} \times 100 \tag{8}$$

$$Signal_{Line} = EMA_9(PPO) \tag{9}$$

6) Rate of Change (ROC): The Rate of Change (ROC) indicator shines a light on the velocity of price movements over a defined timeframe. Eq. (10) unveils its computational heart:

$$ROC = \frac{Lasted_{close} - Previous_{close}}{Previous_{close}} \times 100 \tag{10}$$

7) Chaikin Money Flow Indicator (CMFI): The Chaikin Money Flow (CMF) indicator, a technical analysis stalwart, gauges the volume of capital flowing into or out of a security over time. Its values dance between 1 and -1, with proximity to 1 signifying robust buying pressure and proximity to -1 indicating potent selling pressure.

A CMF value hovering near 1 whispers of a substantial inflow of money into the security, suggesting buyers firmly grip the market reins. Conversely, a

CMF value teetering around -1 hints at a considerable outflow of capital, indicating sellers are asserting control.

However, it's crucial to acknowledge that the CMF indicator should not be a solitary oracle. Its effectiveness is amplified when combined with other technical indicators and fundamental analysis, creating a well-rounded investment perspective. For the mathematically inclined, the CMF's inner workings are unveiled in Eq. (11) to Eq. (13):

$$Multiplier = \frac{((Close-Low)-(High-Close))}{(High-Low)} \quad (11)$$

$$MFV(MoneyFlowVolume) = Volume \times Multiplier \quad (12)$$

$$CMF_{12period} = \frac{21periodSumOfMFV}{21periodSumOfVolume} \quad (13)$$

8) Chande momentum oscillator (CMO): The Chande Momentum Oscillator (CMO) joins the ranks of the RSI and the stochastic oscillator as a fellow momentum indicator. It too dances between -100 and 100, but with its own interpretive twist. When the CMO value waltzes past 50, stock prices are deemed to be frolicking in "overbought" territory. Conversely, a plunge below -50 suggests they're wallowing in the "oversold" zone. Eq. (14) unveils the CMO's inner workings:

$$CMO = 100 \times \frac{S_u - S_d}{S_u + S_d} \quad (14)$$

9) Simple moving average (SMA): The humble Simple Moving Average (SMA) reigns supreme as a fundamental indicator, capturing the essence of a security's price movement over a chosen period. Its power lies in its simplicity – it's just the average of prices during that timeframe. Its true magic unfolds when it's crossed by the actual stock price. Eq. (15) unveils the unassuming yet powerful formula behind the SMA:

$$SMA(M,n) = \sum_{k=a+1}^{a+n} \frac{M(k)}{n} \quad (15)$$

10) Exponential moving average (EMA): The exponential moving average (EMA) is a type of moving average indicator that places greater weight on recent price data, making it more responsive to current trends. Traders employ the EMA on trading charts to identify potential entry and exit points for trades based on the relative position of price action to the EMA. When the EMA is high relative to price, traders may consider selling, while when it is low relative to price, they may consider buying. Eq. (16) illustrates the calculation of EMA:

$$EMA(t) = (M(t) - x) \times \frac{2}{\tau+1} + x \quad (16)$$

$$\text{With } x = EMA(M, t-1, \tau) \quad (17)$$

11) Weighted Moving Average (WMA): While both the weighted moving average (WMA) and exponential moving average (EMA) aim to capture trends by assigning decreasing weights to past closing prices, their weighting mechanisms differ. The EMA employs an exponential decay, where the weights decline rapidly as we move back in time. In contrast, the WMA utilizes a linear decay, resulting in a more gradual decrease in weights assigned to older prices. This distinction highlights the potential for the WMA to offer a different perspective on market trends compared to the EMA, particularly in situations where recent price movements hold significant importance. Eq. (17) shows how WMA is calculated:

$$WMA(M,n) = \frac{SumOfWeighedAverages}{SumOfWeight} \quad (17)$$

12) Hull Moving Average (HMA): The Hull moving average (HMA) stands out among moving average indicators for its superior ability to minimize lag, a common drawback of traditional SMAs. While EMAs and WMAs attempt to address this issue by emphasizing recent data, HMA takes it a step further, achieving even better lag reduction. Eq. (19) details the HMA's calculation, highlighting its distinct approach.:

$$HMA(M,n) = WMA\left(2 \times WMA\left(\frac{n}{2}\right) - WMA(n), \sqrt{n}\right) \quad (18)$$

13) Triple Exponential Moving Average (TEMA): The triple exponential moving average (TEMA) tackles a persistent challenge in technical analysis: how to filter out minor price fluctuations while capturing the underlying trend. Unlike traditional EMAs, TEMA employs a layered approach, essentially applying multiple exponential weightings to price data. This sophisticated strategy, as shown in the subsequent formula, effectively smooths out short-term volatility, revealing the true market direction with greater clarity:

$$TEMA = (3 \times EMA) - \left(3 \times EMA(EMA)\right) + \left(EMA\left(EMA(EMA)\right)\right) \quad (19)$$

14) Commodity Channel Index (CCI): Standing out as a versatile tool for traders, the Commodity Channel Index (CCI) assesses how much a current price deviates from its typical range over a specified period. Essentially, it compares the present price to the average price within that timeframe, translating the difference into an easily interpretable value. Although fluctuating between -100 and 100, the CCI spends a quarter of its time outside this range, signifying potentially significant deviations from normalcy. Eq. (20) and Eq. (21) unveil the mathematical magic behind this indicator, revealing how it transforms raw price data into a powerful market sentiment gauge:

$$CCI = \frac{TP - 20PeriodsSMAofTP}{0.015 \times meanDeviation} \quad (20)$$

$$TypicalPrice(TP) = \frac{High + Low + Close}{3} \quad (21)$$

15) Detrended Price Oscillator (DPO): Unlike most oscillators chasing momentum, the Detrended Price Oscillator (DPO) takes a different path. Its unique strength lies in stripping away trend direction from price data, allowing it to focus solely on the underlying cyclical patterns. By filtering out trend noise, the DPO exposes the true rhythm of the market, highlighting peaks and troughs that hint at potential turning points. Eq. (22) unveils the secret behind this transformation, revealing how the DPO isolates cyclical swings for informed buy and sell decisions based on historical echoes:

$$DPO = \frac{PricefromX}{2} + 1periodAgo - XperiodSMA \quad (22)$$

Where X is the number of periods used for the look-back period.

16) Know Sure Thing (KST): he Know Sure Thing (KST) indicator, conceived by Martin Pring, addresses a fundamental challenge in technical analysis: deciphering the complexities of rate-of-change (ROC) data. KST tackles this challenge by employing a novel multi-period approach. It calculates simple moving averages (SMAs) of four distinct ROC periods, effectively smoothing out volatility and capturing momentum across various timeframes. These tamed ROC values then converge into a single, comprehensive indicator: the KST. To further enhance its interpretability, a 9-period SMA of the KST itself is employed as a signal line, generating crossover alerts for potential trading opportunities. Eq. (23) formalizes this innovative approach, unveiling the intricate interplay of SMAs and ROCs within the KST framework:

$$KST = RCMA_1 + (RCMA_2 \times 2) + (RCMA_3 \times 3) + (RCMA_4 \times 4) \quad (23)$$

where:

$RCMA_1$ = 10-period SMA of 10-period ROC

$RCMA_2$ = 10-period SMA of 15-period ROC

$RCMA_3$ = 10-period SMA of 20-period ROC

$RCMA_4$ = 15-period SMA of 30-period ROC

17) Ease of Movement (EOM): In the technical analyst's toolbox, the Ease of Movement (EOM) indicator stands out as a bridge between the ephemeral world of momentum and the tangible realm of volume. EOM seeks to capture this crucial intersection, condensing the combined influence of price movement and volume into a single, quantifiable value. This value, at its core, aims to answer a critical question: are prices moving "easily" in their current trend? EOM posits that effortless price movements, reflected in high EOM readings, hold the potential to persist and present lucrative trading opportunities. Eq. (24) to Eq. (26) unveil the mathematical alchemy behind EOM, revealing how it transforms raw price and volume data into this insightful blend of momentum and volume analysis:

$$DistanceMoved = (\frac{High+Low}{2} - \frac{PHigh+PLow}{2}) \quad (24)$$

$$BoxRation = \frac{\frac{Volume}{scale}}{High-Low} \quad (25)$$

$$1 - PeriodEOM = \frac{DistanceMoved}{BoxRation} \quad (26)$$

18) Interbank Rate (IBR): The interbank rate, often overlooked yet crucial to the financial system's smooth functioning, operates like an invisible plumbing system connecting banks. This rate dictates the interest charged when banks borrow short-term funds from each other to meet immediate liquidity needs or conversely lend out excess reserves. These interbank loans, captured in Eq. (27), typically have very short durations, often overnight and rarely exceeding a week. Understanding the interbank rate is akin to comprehending the heartbeat of the financial system, as it regulates the flow of liquidity and influences borrowing costs for all institutions within the network:

$$IBR = \frac{SumOfInterestRates}{NbrOfInterestRates} \quad (27)$$

19) Directional Movement Indicator (DMI): The Directional Movement Indicator (DMI) stands apart from conventional trend indicators by offering a multifaceted perspective. Instead of relying on a single metric, DMI deploys a three-pronged attack, harnessing the strengths of the Average Directional Index (ADX), the plus directional indicator (+DI), and the minus directional indicator (-DI). This powerful triumvirate delves deep into the trend's soul, simultaneously revealing its strength, direction, and potential turning points. DMI, with its values ranging neatly between 0 and 100, paints a clear picture of the market's underlying momentum, empowering traders to identify and capitalize on trending opportunities with greater confidence.

$$+DI = 100 \times EMA(\frac{+DMI}{ATR}) \quad (28)$$

$$-DI = 100 \times EMA(\frac{-DMI}{ATR}) \quad (29)$$

$$ADX = 100 \times EMA(\left|\frac{+DI - -DI}{+DI + -DI}\right|) \quad (30)$$

20) Parabolic SAR: The Parabolic SAR (SAR) acts as a vigilant sentinel, constantly scanning the market landscape for potential ambush points – where trends might stall or reverse. This dynamic indicator relies on a trio of key components: The previous SAR value, the extreme point (EP) marking the trend's peak or trough, and the ever-adapting acceleration factor (AF) reflecting the trend's sensitivity. As Eq. (31) and Eq. (32) illustrate, rising EPs fuel AF's ascent, amplifying the SAR's response to potential reversals. Conversely, falling EPs trigger AF's descent, making the SAR more cautious in a weakening trend. This intricate interplay between past, present, and future trends empowers traders to identify critical points of potential stops and reverses, navigating the market with newfound foresight.

$$PSAS + PresviousAF(PreviousEP + PSAR) \quad (31)$$

$$PSAS - PreviousAF(PSAR - PreviousEP) \quad (32)$$