DOI: 10.15849/IJASCA.220720.06

Int. J. Advance Soft Compu. Appl, Vol. 14, No. 2, July 2022
Print ISSN: 2710-1274, Online ISSN: 2074-8523

Candlestick Pattern Classification Using Feedforward Neural Network

Meilona Eurica Karmelia, Moeljono Widjaja, and Seng Hansun

Informatics Department, Universitas Multimedia Nusantara, Indonesia
e-mail: meilona.karmelia@student.umn.ac.id
Informatics Department, Universitas Multimedia Nusantara, Indonesia
e-mail: moeljono.widjaja@umn.ac.id
Informatics Department, Universitas Multimedia Nusantara, Indonesia
e-mail: seng.hansun@lecturer.umn.ac.id

Abstract

Investment in the capital market can help boost a country's economic growth. Without a doubt, in investing, a technical analysis of the condition of the stock is needed at that time. One of the technical analyses that can be done is to look at the historical data of stocks. Candlestick charts can summarize historical data that contain price value for Open, High, Low, and Close (OHLC) in the form of a chart. A group of candlesticks will form a pattern that can help investors to see whether the stock is trending up or down. The number of candlestick patterns and the manual determination of candlestick patterns may take time and effort. Feedforward Neural Network (FNN) is one of the algorithms that can help map the input and output of a given dataset. This study aims to implement FNN to classify candlestick patterns found in historical stock data. The test results show that the accuracy for each model scenario does not guarantee whether all patterns can be properly recognized. This is mainly caused by an imbalanced dataset and the classification process cannot be done properly. Testing with the original data has an accuracy of above 85% on each stock, but the average F1-score is below 45%. Further experiments using random under-sampling and Synthetic Minority Oversampling Technique (SMOTE) result in decreased accuracy value, where the lowest is 59% in PT Bukit Asam Tbk share, and an increased average F1-score, but less than 15%.

Keywords: *Candlestick patterns, feedforward neural network, investment, historical data, OHLC, SMOTE, stocks.*

1 Introduction

Investment is a commitment of some funds for one or more assets owned with the hope of generating positive income in the future [1]. Some examples of investments are precious metals, stocks (capital market), savings, land, property, and others. With current technological advances, investing in the capital market is very easy because many applications can be downloaded and accessed to help carry out the share buying and selling transactions. The existence of the capital market can play a great role in increasing national economic activity because, with the capital market, companies can quickly obtain funds for their operations which in turn increase the national economy of a country [2]. However, it is essential to note that in order to invest in the capital market, prior learning or analysis is required of the current stock conditions.

Technical analysis is a study of how current and past price activities in the capital market can help predict the direction of price movements in the future [3]. Charts can be used as tools to perform technical analysis, one of the charts often used is candlestick charts. On a candlestick chart, each candlestick represents the open, high, low, and close prices within a specified period, for example, for one day or one hour [4]. A collection of several candlesticks in a candlestick chart can form a pattern that could help provide signals for trend reversals. This study focuses on classifying candlestick patterns of historical stock data on five stocks listed on the LQ45 IDX (Indonesia Stock Exchange) in 2021 [5]. LQ45 contains 45 companies in Indonesia that have high liquidity and huge market share with a good financial status [6]. This list has been used in several studies [7]–[10], however in this study we will focus on five stocks, namely ANTM (PT Aneka Tambang Tbk.), ADRO (PT Adaro Energy Tbk.), INCO (PT Vale Indonesia Tbk.), PGAS (PT Perusahaan Gas Negara Tbk.), and PTBA (PT Bukit Asam Tbk).

Candlestick patterns are believed to provide a reversal signal so that they can be a tool for choosing the right entry time in investing. Previous researchers [11], [12] built rules on each pattern by comparing the length of the lower shadows, the length of the upper shadows, and the length of the real body with the previous few days for each type of candlestick. Meanwhile, Kusuma et al. [13] conducted a study to predict future stock market movements with candlestick charts using the Convolutional Neural Network. Stock prediction is also made by Huang et al. [14] by comparing Feedforward Neural Network (FNN) with an adaptive neuro-fuzzy inference system (ANFIS) to predict stock using fundamental financial ratios.

Furthermore, Hu et al. [15] classified candlestick patterns with seven classifiers, namely Bagging, Random-Committee, Random Sub-Space, Partial Decision Tree (PART), Random Forest, Artificial Neural Network (ANN), and Support Vector Machine (SVM). In their research, the researchers described 103 candlestick patterns consisting of several groups. The researchers conducted a classification experiment with 30 pattern representations from each part of the existing group and evaluated the classification with synthetic datasets and real datasets. In short, they used the rules from the described 103 candlestick patterns to generate a synthetic dataset. The results prove that the experiments from synthetic datasets can be used more effectively in choosing the best classifiers to identify candlestick patterns, wherein the experiment Random Forest became the classifier with the best accuracy up to 95.30% and SVM as the worst classifier with an accuracy of 73.49%.

This study aims to implement an FNN with a sampling technique in classifying the candlestick pattern. This method is relatively more straightforward than other methods used in several previous researches explained above and requires lower computational resources. We use a multilayer feedforward neural network with 36 neurons in each hidden layer. The activation functions used are gelu, relu, and softmax in the first hidden layer, second hidden layer, and output layer. The feedforward neural network model is built using the Tensorflow [16] library in Python. Hence, the contributions of this study are 1) a proposed Feedforward Neural Network with under-sampling and over-sampling techniques, 2) three different experimental scenarios in the evaluation phase to represent real-world scenarios, and 3) evaluation of five real stocks listed in the LQ45 indices.

The structure of this paper will be explained in the following series. Section 2 will describe the datasets used, pre-processing step, and the basic concept of FNN. Section 3 will describe the classification results of several scenarios and experimental phases conducted in this study. Lastly, some finishing remarks will be given in Section 4.

2 Research Methods and Data

This section starts by describing the dataset used in this study. Then, the data preprocessing step conducted in this study will be briefly explained, followed by the main algorithm used, namely the Feedforward Neural Network (FNN). Lastly, the confusion matrix as the performance evaluation method will be described.

2.1 Dataset and candlestick patterns

This study uses a dataset from Yahoo! Finance [17] from February 26, 2006, to February 26, 2021. The dataset uses stock data listed on IDX LQ45, with stock codes: ANTM (PT Aneka Tambang Tbk.), ADRO (PT Adaro Energy Tbk.), INCO (PT Vale Indonesia Tbk.), PGAS (PT Perusahaan Gas Negara Tbk.), and PTBA (PT Bukit Asam Tbk). The process of labeling the data on the downloaded dataset is carried out using the Technical Analysis Library (TA-Lib) and re-examining the patterns that have been found manually by visualizing the candlestick chart pieces of the patterns that have been found. This study uses ten (10) types of candlesticks, namely Dragonfly Doji, Gravestone Doji, Bearish Engulfing Pattern, Bullish Engulfing Pattern, Bullish Doji Star, Bearish Doji Star, Hammer, Hanging Man, Morning Star, and Evening Star. Figure 1 is an example of the Dragonfly Doji pattern found in one of the datasets used, and Table 1 is the data distribution in each class.

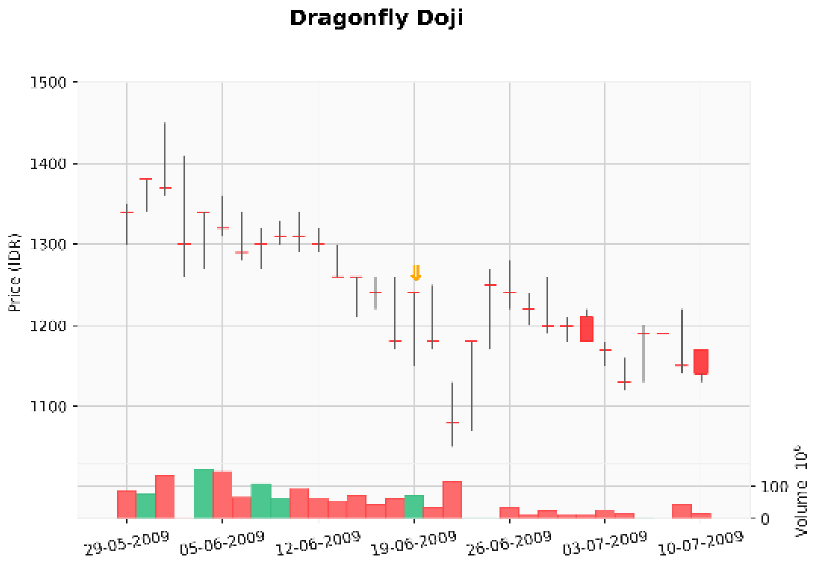

Fig. 1. Example of a Dragonfly Doji pattern

Table 1: Distribution of data in each class

Class	**ADRO**	**ANTM**	**INCO**	**PGAS**	**PTBA**
Bearish Doji Star	24	30	33	41	54
Bearish Engulfing Pattern	18	79	71	77	103
Bullish Doji Star	11	26	37	31	33
Bullish Engulfing Pattern	82	20	25	26	26
Dragonfly Doji	219	284	302	313	265
Evening Star	10	9	10	5	11
Gravestone Doji	177	272	240	172	206
Hammer	15	9	16	20	21
Morning Star	7	2	13	4	12
Unclassified	2526	2957	2928	2992	2945

2.2 Data preprocessing

Figure 2 is a flowchart of the steps carried out in the data preprocessing process. First, the dataset that is still in the form of daily prices will be transformed into a DataFrame with a price scale for three consecutive days. The price value in each dataset will be normalized so that the value is on a scale of 0 to 1. This is done to simplify the model training process. Then, because the label used is a categorical label, categorical encoding is performed to convert the categorical label into a binary vector form with OneHotEncoder in the Scikit-learn library [18]. Furthermore, the dataset is divided into training and testing data with a ratio of 80:20, and 80% of the total training data will be further divided into training and validation data with a ratio of 80:20.

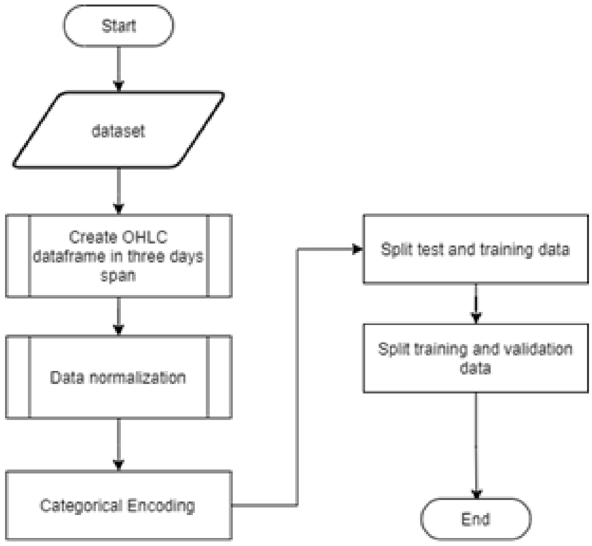

Fig. 2. Data preprocessing flowchart

2.3 Feedforward neural network

FNN is a type of neural network where the connections between neurons do not form a directed cycle [19]. In general, a neural network has at least three layers, namely the input layer, hidden layer, and output layer. The input layer is the first layer that will be passed by the inputted parameters for processing. Furthermore, the hidden layer will be computed between the input and output layers. Finally, the output layer is the layer that will produce the final output. Figure 3 shows a simple architecture of FNN with one hidden layer.

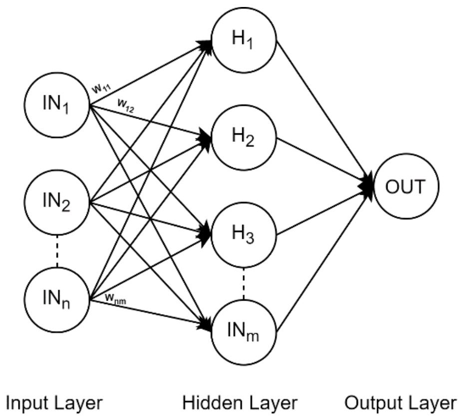

Fig. 3. FNN architecture with one hidden layer

FNN model is built by using several parameters, such as the number of neurons in the hidden layer, the number of hidden layers, and the activation function. In this study, 36 neurons were used in two hidden layers, and the gelu, relu, and softmax activation functions were used in the first, second hidden layers, and the output layer as parameters. The data that has gone through the previous preprocessing process is then used to train and evaluate the model that has been built. For the FNN model, we used the daily price of open, high, low, and close in three consecutive days for the inputs and one out of the eleven classes for the output. First, the training data will be used for the training process of the FNN model that has been built. Then training and validation of loss and accuracy are also displayed in graphical form to show the results of training and data validation on the FNN model. Next, model testing will be carried out on the FNN model that has been trained to determine the performance of the trained FNN model. After that, the evaluation of the model will be carried out by looking at the accuracy value of the results of training, validation, and testing.

2.4 Performance metrics

Finally, the performance evaluation of the Feedforward Neural Network model in this study is displayed with a confusion matrix. A confusion matrix is a table that containing information about the comparison of the model results from the classification trials carried out to the actual classification results. The calculated values are accuracy, precision, recall or specificity, and F1-score [20]. Then, from the values of precision, recall, and F1-score obtained, the average value of each precision, recall, and F1-score for all classes will be calculated as the 'macro' average value to differentiate them from the 'micro' average value of precision, recall, and F1-score for each available class. Equation (1) to Equation (3) represent the formulas for Precision (Prec), Recall (Rec), and F1-score, respectively [21].

$$Prec = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{1}$$

$$Rec = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{2}$$

$$F1 - score = \frac{2 \times Prec \times Rec}{Prec + Rec} \tag{3}$$

3 Results and Discussion

We begin this section by explaining the network architecture and hyperparameters being used in this study. Moreover, the experimental phase is divided into three different scenarios that are explained and discussed later in this section. Lastly, a comparison with several related studies is given in the last part of this section.

Figure 4 is the snipped code of FNN's architecture and hyperparameters used in this study. As previously described, besides the input layer, there are two hidden layers with 36 neurons for each layer. Gelu and relu activation functions are being used in those hidden layers. For the output layer, 11 neurons that represent each candlestick pattern (including the unclassified group) and softmax activation function are used. In model compilation, we used Nadam optimizer from Keras, categorical cross-entropy as the loss function and categorical accuracy as the metric evaluation.

```
import tensorflow as tf

# Build FNN model with 2 hidden layer arch
def FNN_Model(data):
  model = tf.keras.models.Sequential()
  # Input Layer
  model.add(tf.keras.layers.Flatten(input_shape=(data.shape[1],1)))
  # First Hidden Layer
  model.add(tf.keras.layers.Dense(units=36, activation=tf.nn.gelu,
                                  name='dense-36-gelu'))
  # Second Hidden Layer
  model.add(tf.keras.layers.Dense(units=36, activation=tf.nn.relu,
                                  name='dense-36-relu'))
  # Output Layer
  model.add(tf.keras.layers.Dense(11, activation=tf.nn.softmax,
                                  name='dense-11-softmax'))
  print('Input Shape:', data.shape) # Print input dataset
  print()
  print(model.summary()) # Print model summary
  model.compile(optimizer=tf.keras.optimizers.Nadam(learning_rate=0.001),
                loss=tf.keras.losses.CategoricalCrossentropy(),
                metrics=[tf.keras.metrics.CategoricalAccuracy()])
  return model
```

Fig. 4. Networks' architecture and hyperparameters

The results of this study were tested with three scenarios. Each scenario is distinguished from the content of the dataset used, namely 1) using data without under-sampling or over-sampling techniques in the first scenario, then 2) without using the unclassified class and over-sampling the minority class in the second scenario, and 3) using the entire class with both under-sampling for the majority class and over-sampling (SMOTE) [22] in the minority class. An unclassified class is a class that contains patterns that do not belong to the ten (10) candlestick patterns. The under-sampling technique randomly removes some data from the majority class in the training datasets, while over-sampling adds synthetic data from the minority class in the training datasets.

3.1 Scenario 1: Entire contents of the dataset including the unclassified ones

In the first experiment, the model is trained using the original dataset without any under-sampling or over-sampling techniques. Based on the test results shown in Table 2, it can be seen that the accuracy of the best model training, validation, and testing is owned by the model in the ANTM stock code. However, due to the imbalanced dataset, which makes the test data not evenly distributed between each class, the accuracy of each model cannot be used as the only benchmark in the first scenario test. In the five models that have been built, the unclassified class has high precision, recall, and F1-score values among other classes. This is due to a large number of training and testing data in the unclassified class so each model's accuracy value is also very good.

Of the five models trained with different datasets, the model trained with ANTM stock data can detect Dragonfly Doji and Gravestone Doji candlestick patterns better with F1-score values above 90%. The precision, recall, and F1-score values of the best model can be seen in Table 3. Moreover, based on Table 4, the macro average F1-score of each model looks low, which is less than 40%. Therefore, if we treat all classes equally, the performance of each model for classifying each model is the same. This is due to the relatively small amount of data in several classes, and there are still patterns that cannot be identified which makes the macro average F1-score value low.

Table 2: Accuracy results for the first scenario

Stocks	Accuracy (%)		
	Train	**Validation**	**Test**
ADRO	93	88	87
ANTM	**94**	**94**	**95**
INCO	90	89	90
PGAS	92	91	92
PTBA	89	89	88

Table 3: Candlestick patterns classification results for ANTM on first scenario

Class	Precision	Recall	F1-score	Support
Bearish Doji Star	0.00	0.00	0.00	6
Bearish Engulfing Pattern	0.67	0.13	0.22	15
Bullish Doji Star	0.00	0.00	0.00	1
Bullish Engulfing Pattern	0.00	0.00	0.00	3
Dragonfly Doji	**1.00**	0.93	0.96	58
Evening Star	0.00	0.00	0.00	1
Gravestone Doji	0.96	0.94	0.95	52
Hammer	0.00	0.00	0.00	2
Morning Star	0.00	0.00	0.00	3
Unclassified	0.94	**0.99**	**0.97**	598

Table 4: Macro average of the first scenario results

Stocks	Macro Avg		
	Precision	**Recall**	**F1-score**
ADRO	0.27	0.26	0.24
ANTM	0.36	0.30	0.31
INCO	0.27	0.25	0.25
PGAS	0.40	**0.36**	**0.38**
PTBA	**0.41**	0.27	0.30

Figure 5 shows a loss function plot and an accuracy plot for PGAS during the model development. Other stocks show similar results for the loss function and accuracy plots.

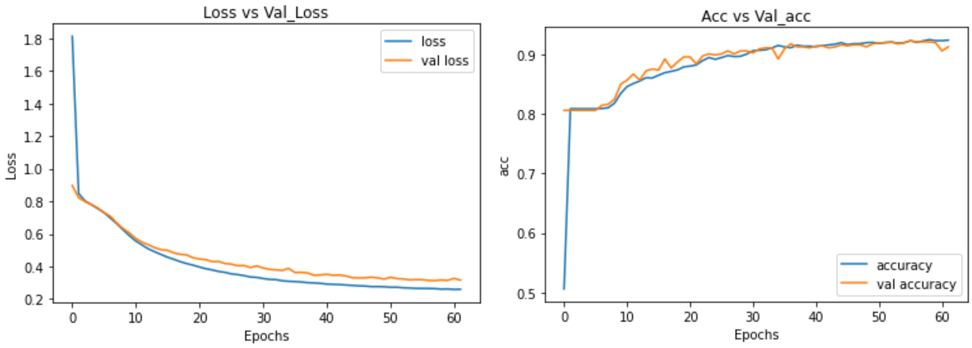

Fig. 5. Loss function and accuracy plots for PGAS

3.2 Scenario 2: Eliminating unclassified ones and performing over-sampling techniques

After evaluating the model trained in the previous scenario, we tried to eliminate the unclassified class to see the effect of biased data and over-sampling SMOTE on the training data. Based on the test results, the accuracy value in the training model for each stock increased several points compared to the accuracy value in the first test scenario. However, it can be seen in Table 5 that the accuracy value of the validation and testing data is not as good as the accuracy value in the first test scenario.

The precision, recall, and F1-score values in Table 6 look better than in the first scenario because eliminating the unclassified class with the highest number and adding training data with the SMOTE over-sampling method can help the model recognize better in other patterns. Based on Table 7, the best macro average F1-score was obtained by ANTM shares with a value of 72%, and the lowest was PGAS shares with a value of 60%. Overall, the performance of each model for classifying each pattern seems to increase with trials without using unclassified class and using SMOTE over-sampling technique.

Table 5: Accuracy results for the second scenario

Stocks	Accuracy (%)		
	Train	Validation	Test
ADRO	95	83	84
ANTM	**98**	**90**	**93**
INCO	95	84	84
PGAS	96	83	83
PTBA	91	82	75

Table 6: Candlestick patterns classification results for ANTM on second scenario

Class	Precision	Recall	F1-score	Support
Bearish Doji Star	0.75	**1.00**	0.86	3
Bearish Engulfing Pattern	**1.00**	0.88	0.94	17

Bullish Doji Star	0.40	0.67	0.50	3
Bullish Engulfing Pattern	**1.00**	0.75	0.86	4
Dragonfly Doji	**1.00**	0.96	**0.98**	71
Evening Star	0.50	**1.00**	0.67	1
Gravestone Doji	0.96	0.94	0.95	49
Hammer	0.00	0.00	0.00	0

Table 7: Macro average of the second scenario results

Stocks	**Macro Avg**		
	Precision	**Recall**	**F1-score**
ADRO	0.65	0.73	0.67
ANTM	0.70	**0.77**	**0.72**
INCO	**0.71**	0.76	0.71
PGAS	0.60	0.64	0.60
PTBA	0.67	0.72	0.68

Figure 6 shows a loss function plot and an accuracy plot for ANTM during the model development. Other stocks show similar results for the loss function and accuracy plots.

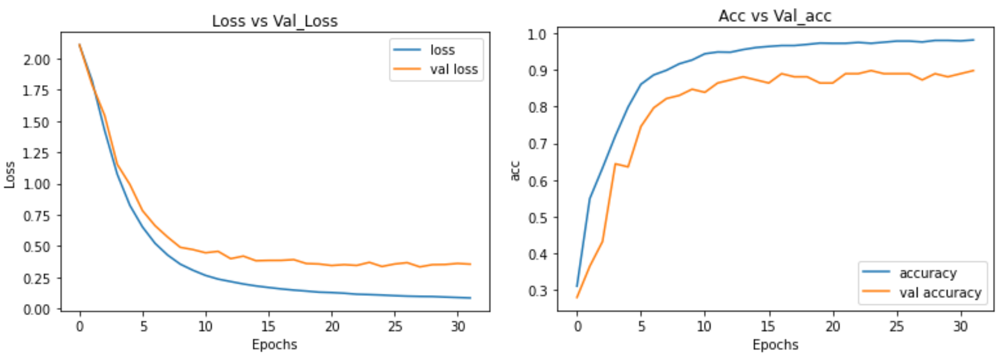

Fig. 6. Loss function and accuracy plots for ANTM

3.3 Scenario 3: Entire dataset and perform random under-sampling and over-sampling

After evaluating the model trained in the two previous scenarios, we then tried to use the entire data (including the unclassified pattern class) and perform both under-sampling and over-sampling techniques on the training data. Based on the test results (Table 8), the best accuracy of the model trained was achieved by PGAS stock data. Furthermore, as shown in Table 9, the precision value when using the random under-sampling method in the unclassified class looks quite good, which is above 90%, which means the value is false. The positive value of the test results obtained is less than the true positive value. However, the recall value in the unclassified class from the results of the third scenario trial seems to have decreased compared to the first scenario. This proves that reducing the amount of data in the unclassified class as the major class and adding synthetic data to the minority classes could increase the false-negative value of the prediction results in the unclassified class, which in turn resulted in a poor recall and F1-score values in the unclassified class.

Table 8: Accuracy results for the third scenario

Stocks	Accuracy (%)		
	Train	**Validation**	**Test**
ADRO	96	74	73
ANTM	97	74	74
INCO	96	71	70
PGAS	**98**	**78**	**83**
PTBA	94	61	59

Table 9: Candlestick patterns classification results for PGAS on third scenario

Class	Precision	Recall	F1-score	Support
Bearish Doji Star	0.28	**1.00**	0.43	8
Bearish Engulfing Pattern	0.48	0.94	0.64	17
Bullish Doji Star	0.20	0.67	0.31	3
Bullish Engulfing Pattern	0.17	0.50	0.25	4
Dragonfly Doji	0.75	0.98	0.85	59
Evening Star	0.00	0.00	0.00	0
Gravestone Doji	0.56	0.78	0.65	36
Hammer	0.18	0.33	0.24	6
Hanging Man	0.00	0.00	0.00	2
Morning Star	0.00	0.00	0.00	0
Unclassified	**0.98**	0.82	**0.89**	604

Table 10: Macro average of the third scenario results

Stocks	Macro Avg		
	Precision	**Recall**	**F1-score**
ADRO	0.29	0.51	0.34
ANTM	0.35	0.73	**0.41**
INCO	0.30	0.68	0.37
PGAS	0.33	0.55	0.39
PTBA	0.28	0.77	0.34

Based on Table 10, the macro average F1-score value of each model still looks low, which is less than 40%, except for the ANTM model with a value of 41%. Until the third scenario, the performance of each model for classifying each pattern still looks less than good. The imbalanced amount of data in several classes that have an impact on the performance of pattern classification is the main reason for the low macro average value.

Figure 7 shows a loss function plot and an accuracy plot for ANTM during the model development. Other stocks show similar results for the loss function and accuracy plots.

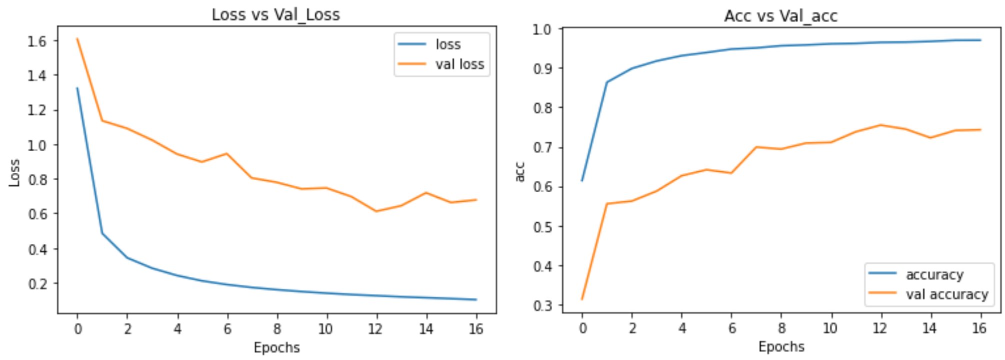

Fig. 7. Loss function and accuracy plots for ANTM

We have successfully implemented the proposed FNN in classifying the candlestick patterns on five stocks. The experimental results show that the best result was obtained from scenario 2, where the over-sampling (SMOTE) technique was applied and the unclassified class was dropped. Moreover, we also compare the result of this study with other competing studies. The summary is shown in Table 11. It is clearly seen that our approach could achieve similar results with other more advanced and complicated techniques.

Table 11: Summary of studies' results

Authors (Year)	**Method(s)**	**Best Result**
Jearanaitanakij and Passaya - 2019 [23]	Convolutional Neural Network	Accuracy: 65.62%
Kusuma et al. - 2019 [13]	Convolutional Neural Network	Accuracy: 92.2%
Hu et al. - 2019 [15]	Bagging, Random Committee, Random Sub Space, PART, Random Forest, Artificial Neural Network, Support Vector Machine	Accuracy: 95.3% (Random Forest)
Xu - 2021 [24]	AdaBoost, Random Forest, XGBoost, Multi Layer Perceptron, Convolutional Neural Network	Accuracy: 90.4% (XGBoost)
Lin et al. - 2021 [25]	Ensemble of Machine Learning methods (Random Forest, Gradient Boosting Decision Tree, Logistic Regression, k-Nearest Neighbors, Support Vector Machine, Long Short-Term Memory)	Accuracy: 91% (k-Nearest Neighbors)

Hung and Chen - 2021 [26]	Convolutional Neural Network – Autoencoder, Recurrent Neural Network	Accuracy: 82.78% (TX dataset) 67.08% (NI225 dataset)
This study	Feedforward Neural Network	Accuracy: 93% F1-score: 72%

4 Conclusion

The implementation of the feedforward neural network algorithm to classify candlestick patterns on stock charts has been completed. The results of the experiments that have been carried out show that the accuracy value generated by each model scenario does not guarantee whether all patterns can be properly recognized because the dataset is not balanced, and it is not easy to carry out the classification process. We use 36 neurons in two hidden layers and different activation functions of gelu, relu, and softmax in the first hidden layer, second hidden layer, and output layer. A good accuracy result was obtained in the first test scenario with an accuracy value above 85% for each stock, and the best accuracy was being owned by ANTM stock (95%). However, the F1-score value in each pattern was not good, so the macro average F1-score in the first scenario is below 40%. Meanwhile, experiments using random under-sampling and SMOTE over-sampling caused the accuracy value to decrease. The lowest value was in PTBA shares at 59%, and the highest was PGAS at 83%. Moreover, the macro average F1-score was slightly increased by less than 15% in averages.

The best result was obtained in Scenario 2 by removing the 'unclassified' class and performing SMOTE over-sampling technique in the dataset. The best accuracy was reached by ANTM (93%) with an overall F1-score of 72%. For future research, more advanced Machine Learning or Deep Learning methods could be implemented to solve this problem, such as Multinomial Logistic Regression [27], Fuzzy Classifier [28], Support Vector Machine [29], Recurrent Neural Networks [30], [31], or even ensemble method [32]. Another more interpretable Machine Learning method, namely the Decision Tree [33], also could be applied shortly.

ACKNOWLEDGEMENTS
The authors would like to acknowledge the support given by Universitas Multimedia Nusantara during this study.

References

[1] McCallum, S. & Viviers, S. (2021). What constitutes impact? Definition, motives, measurement and reporting considerations in an African impact

investment market. *African J. Bus. Ethics*, 15(1), 10–27.

[2] Al-Fadhat, F. & Nadhir, M.R. (2019). Foreign investment and the political economy of Indonesian capital market in 2015-2016. *Humanit. Soc. Sci. Rev.*, 7(6), 340–348, doi: 10.18510/hssr.2019.7659.

[3] Chen, J. (2010). *Essentials of Technical Analysis for Financial Markets*, 1st edition. Wiley.

[4] Martinssson, F. & Liljeqvist, I. (2017). Short-Term Stock Market Prediction Based on Candlestick Pattern Analysis. KTH.

[5] IDX. (2021). Indonesia Stock Exchange. https://www.idx.co.id/ (accessed Feb. 27, 2021).

[6] Tanuwijaya, J. & Hansun, S. (2019). LQ45 Stock Index Prediction using k-Nearest Neighbors Regression. *Int. J. Recent Technol. Eng.*, 8(3), 2388–2391, doi: 10.35940/ijrte.C4663.098319.

[7] Hansun, S. & Young, J.C. (2021). Predicting LQ45 Financial Sector Indices using RNN-LSTM. *J. Big Data*, 8(1), 104, doi: 10.1186/s40537-021-00495-x.

[8] Pantagama, M. & Rikumahu, B. (2020). Indonesia Financial Sector Stock Prediction using Long Short-Term Memory Network Algorithm and Modeling (Study of Banking in August 2018 LQ45 Index). In *Digital Economy for Customer Benefit and Business Fairness*, Anggadwita, G. & Martini, E. (Eds). Routledge, (pp. 159–164).

[9] Pramanaswari, A.S.I. & Yasa, G.W. (2018). Graham & Dodd Theory in Stock Portfolio Performance in LQ 45 Index at Indonesia Stock Exchange. *Int. Res. J. Manag. IT Soc. Sci.*, 5(6), 52–59, doi: 10.21744/irjmis.v5n6.338.

[10] Nurmalitasari, Sumarlinda, S., Supriyanto, N., & Putri, D.K. (2020). LQ45 Stock Price Predictions Using The Deep Learning Method. *Int. J. Adv. Res. Publ.*, 4(4), 20–23.

[11] Lee, C.-H. L., Liaw, Y.-C., & Hsu, L. (2011). Investment Decision Making by Using Fuzzy Candlestick Pattern and Genetic Algorithm. In *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)* (pp. 2696–2701), IEEE, doi: 10.1109/FUZZY.2011.6007707.

[12] do Prado, H.A., Ferneda, E., Morais, L.C.R., Luiz, A.J.B, & Matsura, E. (2013). On the Effectiveness of Candlestick Chart Analysis for the Brazilian Stock Market. *Procedia Comput. Sci.*, 22, 1136–1145, doi: 10.1016/j.procs.2013.09.200.

[13] Kusuma, R.M.I., Ho, T.-T., Kao, W.-C., Ou, Y.-Y., & Hua, K.-L. (2019). Using Deep Learning Neural Networks and Candlestick Chart Representation to Predict Stock Market. Available: http://arxiv.org/abs/1903.12258.

[14] Huang, Y., Capretz, L.F., & Ho, D. (2019). Neural Network Models for Stock Selection Based on Fundamental Analysis. In *32nd Canadian Conference on Electrical & Computer Engineering* (pp. 1–4). Available: http://arxiv.org/abs/1906.05327.

[15] Hu, W., Si, Y.-W., Fong, S., & Lau, R.Y.K. (2019). A Formal Approach to Candlestick Pattern Classification in Financial Time Series. *Appl. Soft Comput.*, 84, 105700, doi: 10.1016/j.asoc.2019.105700.

[16] Abadi, M., *et al.* (2016). TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. Available: http://arxiv.org/abs/1603.04467.

[17] Yahoo! Finance. (2021). Quotes. https://finance.yahoo.com/lookup (accessed May 01, 2021).

[18] Pedregosa, F., *et al.* (2011). Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.*, 12(85), 2825–2830,

[19] Fuangkhon, P. (2014). An Incremental Learning Preprocessor for Feed-forward Neural Network. *Artif. Intell. Rev.*, 41(2), 183–210, doi: 10.1007/s10462-011-9304-0.

[20] Vernanda, Y., Hansun, S., & Kristanda, M.B. (2020). Indonesian Language Email Spam Detection using N-gram and Naïve Bayes Algorithm. *Bull. Electr. Eng. Informatics*, 9(5), 2012–2019, doi: 10.11591/eei.v9i5.2444.

[21] Ferdina, V., Kristanda, M.B.,& Hansun, S. (2019). Automated Complaints Classification using Modified Nazief-Adriani Stemming Algorithm and Naive Bayes Classifier. *J. Theor. Appl. Inf. Technol.*, 97(5), 1604–1614.

[22] Rachburee, N. & Punlumjeak, W. (2021). Oversampling Technique in Student Performance Classification from Engineering Course. *Int. J. Electr. Comput. Eng.*, 11(4), 3567–3574, doi: 10.11591/ijece.v11i4.pp3567-3574.

[23] Jearanaitanakij, K. & Passaya, B. (2019). Predicting short trend of stocks by using convolutional neural network and candlestick patterns. In *2019 4th International Conference on Information Technology (InCIT)* (pp. 159–162), IEEE, doi: 10.1109/INCIT.2019.8912115.

[24] Xu, C. (2021). Image-based candlestick pattern classification with machine learning. In *2021 6th International Conference on Machine Learning Technologies* (pp. 26–33), doi: 10.1145/3468891.3468896.

[25] Lin, Y., Liu, S., Yang, H., & Wu, H. (2021). Stock trend prediction using candlestick charting and ensemble machine learning techniques with a novelty feature engineering scheme. *IEEE Access*, 9, 101433–101446, doi: 10.1109/ACCESS.2021.3096825.

[26] Hung, C.-C. & Chen, Y.-J. (2021). DPP: Deep predictor for price movement from candlestick charts. *PLoS One*, 16(6), e0252404, doi:

10.1371/journal.pone.0252404.

[27] Munandar, T.A., Sumiati, S., & Rosalina, V. (2021). Predictive model for heart disease diagnosis based on multinomial logistic regression. *Inf. Technol. Control*, 50(2), 308–318, doi: 10.5755/j01.itc.50.2.27672.

[28] Widjaja, M., Darmawan, A., & Mulyono, S. (2012). Fuzzy Classifier of Paddy Growth Stages Based on Synthetic MODIS Data. In *2012 International Conference on Advanced Computer Science and Information Systems (ICACSIS)* (pp. 239–244).

[29] Tally, M.T. & Amintoosi, H. (2021). A Hybrid Method of Genetic Algorithm and Support Vector Machine for Intrusion Detection. *Int. J. Electr. Comput. Eng.*, 11(1), 900–908, doi: 10.11591/ijece.v11i1.pp900-908.

[30] Shahreza, H.O., Amini, A., & Behroozi, H. (2020). Predicting the empirical distribution of video quality scores using recurrent neural networks. *Int. J. Eng.*, 33(5), 984–991, doi: 10.5829/ije.2020.33.05b.32.

[31] Rikukawa, S., Mori, H., & Harada, T. (2020). Recurrent Neural Network Based Stock Price Prediction Using Multiple Stock Brands. *Int. J. Innov. Comput. Inf. Control*, 16(3), 1093–1099.

[32] Beulah, D. & Raj, P.V.K. (2022). The ensemble of unsupervised incremental learning algorithm for time series data. *Int. J. Eng.*, 35(2), 319–326, doi: 10.5829/IJE.2022.35.02B.07.

[33] Lee, C.S., Cheang, P.Y.S., & Moslehpour, M. (2022). Predictive analytics in business analytics: Decision tree. *Adv. Decis. Sci.*, 26(1), 1–30, doi: 10.47654/v26y2022i1p1-30.

Notes on contributors

Meilona Eurica Karmelia is a Software Developer with a demonstrated history of working in the human resources industry. Skilled in C#, Microsoft SQL Server, HTML, and JavaScript. Strong engineering professional with a Bachelor's degree focused in Informatics from Universitas Multimedia Nusantara and certified as Microsoft Technology Associate (MTA) on October 11, 2021.

Moeljono Widjaja is a lecturer at the Department of Informatics, Universitas Multimedia Nusantara (Indonesia) with a Doctoral Degree in Electrical Engineering from Monash University (Australia) in 2003. He obtained a Bachelor's Degree in Electrical Engineering from Wayne State University (USA) in 1992 and a Master's Degree in Electrical Engineering from the Ohio State University (USA) in 1994.
His research interests are artificial intelligence, big-data analytics, simulation/modeling, and optimization. He developed a fuzzy controller for an inverted pendulum system and a fuzzy-based bidding strategy for generators in an electricity market. He has been working on intelligent energy management systems. He is a professional member of ACM.

Seng Hansun received the Bc. degree in Mathematics (S.Si.) from Universitas Gadjah Mada, Yogyakarta, in 2008 and Master of Computer Science (M.Cs.) degree from the same university in 2011. Since then, he has been a Lecturer with the Computer Science Department, Universitas Multimedia Nusantara (UMN), Indonesia. He had been appointed as the Information and Communication Technology (ICT) Faculty Research Coordinator, Deputy Head of Computer Science Department, and Head of Informatics Department at UMN. He had published two books and more than 135 articles during his career as both academician and researcher at UMN. His research interests lately include computational science, soft computing methods, and internet and mobile technology in various fields, especially in the Medical Informatics area.