ILKOM Jurnal Ilmiah Vol. 15, No. 2, August 2023, pp. 271-282
Accredited 2nd by RISTEKBRIN No. 200/M/KPT/2020; E-ISSN 2548-7779 | P-ISSN 2087-1716

Research Article Open Access (CC–BY-SA)

Rectified Linear Units and Adaptive Moment Estimation Optimizer on ANN with Saved Model Prediction to Improve The Stock Price Prediction Framework Performance

Sekhudin [a,1]; Yuli Purwati [a,2,*]; Fandy Setyo Utomo [a,3]; Mohd Sanusi Azmi [b,4]; Pungkas Subarkah [a,5]
[a] *Universitas AMIKOM Purwokerto, Jln. Letjend Pol. Soemarto No.127 Purwokerto, 53127, Indonesia*
[b] *Universiti Teknikal Malaysia Melaka, Melaka, 76109, Malaysia*
[1] *sekhudinpbg3@gmail.com;* [2] *yulipurwati@amikompurwokerto.ac.id;* [3] *fandy_setyo_utomo@amikompurwokerto.ac.id;* [4] *sanusi@utem.edu.my;* [5] *subarkah@amikompurwokerto.ac.id*
* *Corresponding author*

Article history: Received January 13, 2023; Revised April 06, 2023; Accepted June 27, 2023; Available online August 16, 2023

Abstract

A stock is a high-risk, high-return investment product. Prediction is one way to minimize risk by estimating future prices based on past data. There are limitations to solving the stock prediction problem from previous research: limited stock data, practical aspects of application, and less than optimal stock price prediction results. The main objective of this study is to improve the prediction performance by formulating and developing the stock price prediction framework. Furthermore, the research provides a stock price prediction framework that can produce superior prediction results than the previous study with fast computation time. The proposed framework deals with data generation, pre-processing and model prediction. In further, the proposed framework includes two prediction methods for predicting stock closing prices: stored model prediction and current model prediction. This study uses an artificial neural network with Rectified Linear Units as an activation function and Adam Optimizer to predict stock prices. The model we have built for each forecasting method shows a better MAPE value than the model in previous studies. Previous research showed that the lowest MAPE was 1.38% for TLKM shares and 0.81% for BRI. Our proposed framework based on the stored model prediction method shows a MAPE value of 0.67% for TLKM shares and 0.42% for BBRI. While the current model prediction method shows a MAPE value of 0.69% for TLKM shares and 0.89% for BRI. Furthermore, the stored model prediction method takes 1.0 second to process a single prediction request, while the current model prediction takes 220 seconds.

Keywords: Closing Price; Intelligent System; Investment; Machine Learning; Supervised Learning.

Introduction

Stocks are one of the most popular financial investment instruments[1]. Stocks offer quite lucrative returns compared to other investment instruments, but like a double-edged sword, they also carry a high level of risk due to fluctuating price movements. The law of supply and demand influences fluctuating price movements[2]. The trading process is organized by an official body called the capital market. A capital market is where long-term transaction processes take place[3]. The capital market is not a physical facility or a discrete entity, and the capital market brings together sellers and buyers of shares as a claim on business ownership [4]. The capital market or stock market consists of several stock exchanges around the world [5]. Indonesia Stock Exchange (IDX) is the official organizer of stock trading in Indonesia. At the time of writing, IDX has 820 stocks. With such a large number of stocks and constantly fluctuating stock prices, it is not easy to decide when to buy and sell stocks because it is related to an uncertain future. Therefore, it is necessary to make predictions about stock prices to help make the right investment decisions.

The movement of share prices on the IDX is influenced by many economic and non-economic factors. However, IDX has published information on the trading data of shares registered in the capital market. The public can access the data so that it is possible to make predictions to minimize risk. Efforts to predict stock prices have many variations, one of which predicts stock prices using computational calculations with certain algorithms, one of which is an artificial neural network. An artificial neural network is one of the algorithms that can be used in the case of prediction

http://dx.doi.org/10.33096/ilkom.v15i2.1586.271-282

[6]. The study [7] implemented an artificial neural network to predict the stock price of KAEF during the COVID-19 pandemic. This study uses three datasets, one of which is KAEF stock history data. The historical data comes from yahoo finance, but they do not explicitly explain how they got it. The best model they built has a configuration consisting of 1 input layer, two hidden layers of 20 nodes each, and an output layer. In addition, their activation function is a binary sigmoid with a maximum epoch of 1000. The best model they built has a value of 1.3% with MAPE (Mean Percentage Absolute Error) measurements.

With predictable stock limitations, the prediction results from the built model only apply to specific stocks. This method is not wrong, but it has the disadvantage that the dataset cannot be updated and other risks, for example, the loss of the dataset and the accumulation of duplicate datasets in local storage. [7] Based on previous studies, this paper proposes a framework that implements solutions related to research limitations with ANNs configurations that differ from previous studies [8].

Based on previous studies, this paper proposes a framework that implements solutions related to research limitations with ANNs configurations that differ from previous studies. This study uses an artificial neural network with Rectified Linear Units as an activation function and Adam Optimizer to predict stock prices. A programming framework includes code libraries, software models, Application Programming Interfaces (APIs), and various other elements that can simplify the programming process. The proposed framework is built on a particular server that implements Representational State Transfer (REST) architecture. REST is an architecture for providing standards between computer systems on the web, making it easier for systems to communicate with each other [9]. REST has an architecture that is lighter, flexible, and easy to maintain [10]. REST allows JSON message format [11]. The proposed framework handles at least three processes: data creation, pre-processing, and model prediction.

In the proposed framework, the dataset is obtained by retrieving it from the source with an HTTP request, pre-processing the data, then performing the prediction process and sending the prediction results to the client. To make the framework dynamic, we expand the range of predictable stocks so that clients can predict any stock listed on the IDX. On the other hand, the proposed framework also provides two prediction methods with different ANN model configurations from previous studies. Namely, the saved model prediction is a method where we have trained and tested the previous model and then saved it as a model file. The second method is the current model prediction, a prediction method in which the model training process is performed directly by the framework at the moment of analysis.

The main objective of this study is to improve the prediction performance by formulating and developing the stock price prediction framework. Furthermore, the research contribution is to provide a stock price prediction framework that is able to produce better prediction results than the previous study with fast computation time. In addition, the ANNs model configuration implemented in the proposed framework has two prediction methods: saved model prediction and current model prediction. The proposed framework handles the data generation, pre-processing and model prediction processes. The application of the ANNs model in the proposed framework is expected to generate reasonable predictions to help determine the right investment decision.

Method

Our proposed framework is based on a particular server. Research [12], [13], provides us with references on how servers should be made, and as a result, we use them as our default servers. Next, **Figure 1** is a flowchart that illustrates how the processes in the proposed framework are interconnected.

In **Figure 1**, that in this prediction process, the first is to make a request, then there are 2 possibilities, namely if you agree, you will enter the JSON data stage, but if you do not agree, 2 stages will be carried out, namely JSON and check history data, if the data is correct, it will enter the data pre-processing stage. Furthermore, if the model is stored, it will apply the prediction method and the results will come out, but if the model is still current, there will be a verification by means of the first and second training and finally the prediction results will come out.

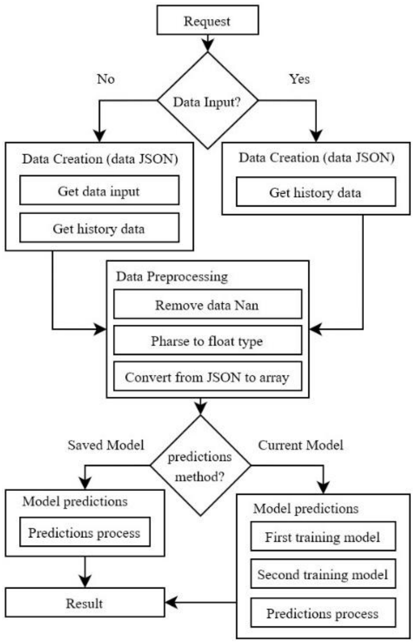

Figure 1. Proposed framework flow

Based on **Figure 1**, the proposed framework will handle the process of data generation, pre-processing and prediction according to the chosen method. The proposed framework uses ANNs as the prediction algorithm for the prediction process. Next, **Table 1** shows the tools and libraries used to build the proposed framework.

Table 1. Tools and libraries

Tools/Libraries	Version	Function
Nodejs	16.16.0	Runtime Javascript
Tensorflow/tfjs-node	3.20.0	Used to build the ANNs model
Express	4.18.1	Used to build servers
Yahoo-finance2	2.3.6	Used to get stock trading history from yahoo finance
Axios	0.27.2	Used to handle http requests
Htmlparser2	8.0.1	Used to parse html strings
Cheerio	1.0.0-rc-12	Handle the scrapping process

The proposed framework includes data creation, pre-processing and model prediction. Subsection 2.1.1 describes the data creation module, subsection 2.1.2 presents the pre-processing module, and subsection 2.1.3 presents the model prediction module.

A. Data Creation

The data required by the proposed framework are stock trade history data and input data. The proposed framework will generate both data. The data generation flow is shown in **Figure 2**.

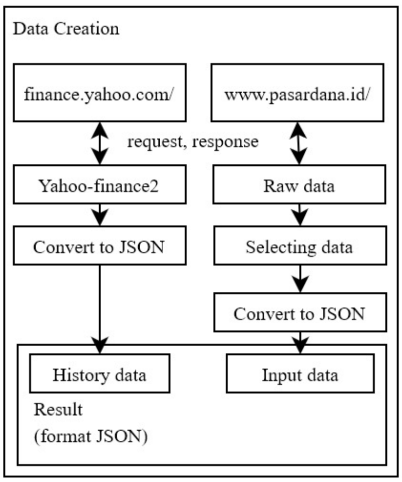

Figure 2. Data creation flow

Stock trading history data, which consists of the key date (date of the trading day), open (opening price), high (highest), close (closing price) and volume (number of shares sold). Hereafter referred to as historical data, this is past data that provides a summary of individual share price movements for shares traded on the stock market. This data includes the information needed to train the model, such as the opening price, the highest price, the lowest price and the closing price. This data is only used in the proposed framework to predict the current model prediction method. There are many ways to obtain this data. The most common is to get it in the form of a file, such as a CSV file downloaded from the yahoo finance web. In this proposed framework, the data is obtained from the yahoo finance web by using a third party library, yahoo-finance2. The library is available at www.npmjs.com/package/yahoo-finance2. The proposed framework will define values in stock code parameters on demand prediction and stock period. Based on these two parameters, yahoo-finance2 will provide historical data according to the parameters. The historical data will be in JSON format.

Input data is input to the prediction process. The framework provides the input data as default input if the client does not attach it to the prediction request. The input data is used in every prediction method, both the saved model prediction and the current model prediction. The input data contains data on the opening price, the highest price and the lowest price of a stock on that day. The input data was obtained from the website www.pasardana.id. The proposed framework was obtained using web scraping techniques. Web scraping is a technique for retrieving information from the web in the form of unstructured data and transforming it into structured/semi-structured data so that it is easy to understand [14], [15]. The proposed framework requests by defining the share code, the pasardana.id web will respond in the form of raw data in HTML form. The proposed framework will process the HTML data. The process is in the form of sorting information by taking certain HTML tags. The selected HTML tags contain information that will be used as input data. The required information will then be converted into JSON form. We aim to convert the required information into JSON format so that the scraped data can be easily used and understood.

B. Data Pre-processing

Data pre-processing is the process by which raw data is transformed, inspected, filtered and processed into a format that can be used for analytical applications such as machine learning [16], [17]. Data pre-processing is necessary to check the quality of the data by checking the completeness, accuracy and consistency of the data. The pre-processing process will prevent errors due to irregular, incomplete dataset conditions. Figure 3 shows the stages of the proposed framework during data pre-processing.

Historical and input data in the previous process will go through data pre-processing. According to **Figure 3**, the proposed framework has two stages, namely data cleaning and data transformation, and one additional stage, namely the data separation process. Data separation is the process of dividing historical data into training input data and training target data. This process only applies to historical data. In the proposed framework, historical data and input data are selected. The required data is then converted into float format. In addition, the input and historical data are converted to array form.

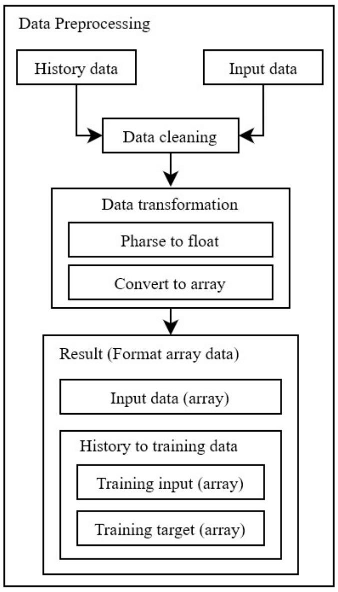

Figure 3. Data pre-processing flow

The data separation stage only applies to historical data. The purpose of data separation is to separate historical data into two parts, namely training input data and training target data. For example, for training data and test data in dynamic sets: historical training data comparing historical data from 1 January 2020 to 24 December 2022. However, the test data is historical from 1 October to 30 October 2022. The data is now divided into 576 rows of training data and 26 rows of test data. The details of the data used are 576 data consisting of 24 months of historical trading data, 26 are guilty of 1 month of historical trading. The training input data is an array of the opening price, the highest price and the lowest price. In contrast, the training target data is an array of closing prices. The results of the pre-processing data are input data, training input data and training target data. Each of these data is already in the form of an array.

C. Model Predictions

Model on the proposed framework using ANN algorithms. ANN henceforth referred to as ANNs, are inspired by biological brain networks[8][18][19]. ANN can recognise past activities or learn from experience. The past data is studied by the ANN so that it can predict future data that has never been studied. Like biological brain networks, ANN consist of many interconnected processing elements called neurons. The neurons are connected by synapses. Each synapse has a weight. These neurons work together to process and transmit information. Neurons can receive one or more inputs and produce outputs, and these outputs can also act as inputs from the neurons in front of them.

Equation 1 shows an artificial neural network when written in a mathematical formula, where is the output of the neurons, is the input value, represents the weight value, is the bias value, and is the number of neurons[20]. (1):

$$Z = \left(\sum_{i=1}^{n} X_i \,.\, W_i\right) + b \qquad (1)$$

The central aspect that determines the performance of ANNs is the configuration of the model parameters. In this study, we use Rectified Linear Units (ReLU) as the activation function and Adam optimizer on the ANN model parameters. The Rectified Linear Units are formulated as in Equation 2 below[20] : (2):

$$f(y) = \begin{cases} y, if\ y > 0 \\ 0, if\ y\ \leq 0 \end{cases} \qquad (2)$$

D. Data Collections

It has already been explained that the framework generates historical data and input data. The proposed framework has dynamically set the historical data period to 2 years from today. For example, if today is 1 October 2022, the historical data period is 10 January 2019 to 24 December 2021. Meanwhile, the input data will be today's opening price, the highest price and the lowest price. For the stock code, the proposed framework provides flexibility for the client when attaching the stock code to the request.

In the saved model forecasting method, the model training uses TLKM (Telkom Indonesia) stock data. For training data, use historical stock data from 10 January 2019 to 24 December 20221. Meanwhile, for testing data, use data from 1 January 2022 to 1 June 2022. In the current model prediction method, the prediction model has not been trained. The training process takes place when there is a prediction request. Under these conditions, the current model requires historical and input data. The historical and input data come from the proposed framework, so the period is also the same.

To test the proposed framework, we randomly selected two stocks from different sectors, and these stocks include TLKM and, BRI (Bank Rakyat Indonesia). In contrast, the data range used in the testing process is from 1 October 2022 to 30 October 2022. **Table 2** shows the stocks used to test the framework and their trading period.

Table 2. Stock list for framework testing

Code	Name	Sector Name	Period
TLKM	Telkom Indonesia	Telecommunications and network services	02/10/2020 to 30/10/2022
BRI	Bank Rakyat Indonesia	Finance	

E. Tes Scenario

In the previous section, it was mentioned that the saved model prediction uses a previously trained model and the training of the model is outside the proposed framework. In the current model prediction, the model training process is carried out within the proposed framework. Although the training process is different, the training and framework testing are performed using the same devices.

We tested the proposed framework on two stocks from two different sectors. In our tests, we also include the average time needed to make predictions for each prediction method. The application we used during the test was the Postman application. To test the prediction results, we use the RMSE (Root Mean Square Error), MSE (Mean Square Error) and MAPE metrics.

F. Metrics for evaluation

We use several evaluation metrics to measure the prediction results generated by our framework. The metrics include RMSE, MSE and MAPE. Next, **Table 3** shows a list of the evaluation metrics used and their calculation formulas.

Table 3. Metrics evaluation

Metrics	Formula	Evaluation Focus
RMSE	$RMSE = \sqrt{\sum \frac{(Y'-Y)^2}{n}}$	The sum of the squared errors or the difference between the predicted and actual value[21]
MSE	$MSE = \sum \frac{(Y'-Y)^2}{n}$	Calculates the square of the error between the predicted result and the actual value [21]-[23]. Unlike the RMSE, the MSE does not go through a root process.
MAPE	$MAPE =$ $= \frac{1}{n}\sum_{t=1}^{n}\frac{\lvert y-yt\rvert}{yt}x100$	Calculate the average of the absolute percentage error value between the target and the prediction [23].

The lower the MAPE, the more accurate the forecast model. The MAPE forecast accuracy rating scale[24] is shown in **Table 4**.

Table 4. MAPE accuracy scale

MAPE	MAPE accuracy scale
≤ 10%	Highly accurate
11% to 20%	Good forecast
21% to 50%	Reasonable forecast
≥ 51%	Inaccurate forecast

Results and Discussion

We conducted several experiments to determine the performance of the proposed framework. The experiment covers the entire process, namely data creation, data pre-processing and model prediction, and the time taken by the framework to process prediction requests until the framework completes and provides prediction results. We run a trial period every day during trading hours. Specifically, the trial period is from 1 October to 30 October 2022.

A. Data creation and pre-processing phase

Table 5 and **Table 6** show the prediction date, input data (opening, high and low prices) and targets (closing prices). We obtained the input and target data using observation. This means that the framework produces the input data, while we know the prediction targets from stock market information after the prediction process has been carried out.

Table 5. TLKM stock data

Date	Input Data (IDR)			Target (IDR)
	Open	*High*	*Low*	
2022/10/03	4460	4470	4380	4460
2022/10/04	4460	4510	4430	4440
2022/10/05	4490	4540	4460	4460
2022/10/06	4460	4480	4420	4430
2022/10/07	4400	4430	4350	4350
2022/10/10	4340	4420	4330	4420
2022/10/11	4360	4380	4310	4310
2022/10/12	4300	4350	4280	4340
2022/10/13	4330	4350	4290	4300
2022/10/14	4360	4360	4290	4290
2022/10/17	4200	4330	4200	4330
2022/10/18	4310	4340	4250	4250
2022/10/19	4220	4300	4200	4200

Date	Input Data (IDR)			Target (IDR)
	Open	***High***	***Low***	
2022/10/20	4200	4350	4200	4350
2022/10/21	4300	4390	4300	4360
2022/10/24	4380	4450	4380	4410
2022/10/25	4430	4470	4390	4390
2022/10/26	4420	4430	4330	4370
2022/10/27	4390	4410	4340	4370
2022/10/28	4420	4450	4380	4450

Based on **Tables 5** and **6**, the proposed framework consistently produces the data needed in the prediction process, namely the input data. **Tables 5** and **6** show the prediction dates of the two stocks and the input data obtained from the framework and the prediction targets known from stock market information at the end of trading. For example, in Table 8, 03 October 2022 is the prediction date. The input data for the forecast consists of an opening price of IDR 4460, a high price of IDR 4470 and a low price of IDR 4380. In addition, the prediction target at the close of trading is IDR 4460.

Table 6. BRI stock data

Date	Input data (IDR)			Target (IDR)
	Open	***High***	***Low***	
2022/10/03	4500	4550	4480	4530
2022/10/04	4560	4650	4560	4640
2022/10/05	4660	4660	4540	4540
2022/10/06	4540	4570	4510	4510
2022/10/07	4480	4490	4430	4440
2022/10/10	4420	4430	4340	4370
2022/10/11	4390	4430	4370	4430
2022/10/12	4470	4470	4400	4400
2022/10/13	4390	4420	4350	4350
2022/10/14	4380	4390	4270	4270
2022/10/17	4270	4320	4260	4300
2022/10/18	4280	4340	4280	4290
2022/10/19	4280	4340	4280	4300
2022/10/20	4280	4400	4280	4390
2022/10/21	4360	4460	4360	4430
2022/10/24	4430	4540	4430	4500
2022/10/25	4560	4610	4550	4590
2022/10/26	4600	4610	4500	4530
2022/10/27	4550	4600	4530	4600
2022/10/28	4600	4630	4560	4630

Next, **Table 7** shows the amount of historical data generated by the proposed framework during the data creation stage and after the data pre-processing stage.

Table 7. Amount of historical data during data creation and after data pre-processing

Code	Historical data	
	Data creation (data)	***After data pre-processing (data)***
TLKM	734	730
BRI	734	733

B. *Stock Prediction Results*

We have presented them in **Table 8** and **Table 9** shows the prediction results of TLKM and BRI stocks using the two prediction methods.

Table 8. TLKM stock prediction

Oct 2020	Saved Model Prediction (IDR)			Current Model Prediction (IDR)		
	7 nodes	*10 nodes*	*13 nodes*	*7 nodes*	*10 nodes*	*13 nodes*
3	4404,88	4407,06	4416,13	4404,88	4407,06	4416,13
4	4463,44	4478,31	4484,55	4463,44	4478,31	4484,55
5	4493,4	4508,36	4514,65	4493,4	4508,36	4514,65
6	4439,59	4446,52	4453,09	4439,59	4446,52	4453,09
7	4377,88	4386,53	4393,75	4377,88	4386,53	4393,75
10	4374,74	4397,78	4402,82	4374,74	4397,78	4402,82
11	4332,42	4338,49	4345,54	4332,42	4338,49	4345,54
12	4310,94	4325,93	4331,23	4310,94	4325,93	4331,23
13	4309,75	4316,31	4322,65	4309,75	4316,31	4322,65
14	4306,79	4306,8	4314,94	4306,79	4306,8	4314,94
17	4269,8	4305,3	4309,96	4269,8	4305,3	4309,96
18	4280,7	4288,52	4296,23	4280,7	4288,52	4296,23
19	4247,59	4269,72	4275,2	4247,59	4269,72	4275,2
20	4280,83	4321,25	4326,12	4280,83	4321,25	4326,12
21	4347,6	4373,56	4377,99	4347,6	4373,56	4377,99
24	4416,47	4437,74	4442,1	4416,47	4437,74	4442,1
25	4420,66	4432,42	4439,15	4420,66	4432,42	4439,15
26	4357,63	4359,13	4368,78	4357,63	4359,13	4368,78
27	4362,38	4368,54	4375,64	4362,38	4368,54	4375,64
28	4405,16	4414,43	4421,04	4405,16	4414,43	4421,04

Table 9. BRI stock prediction

Oct 2020	Saved Model Prediction (IDR)			Current Model Prediction (IDR)		
	7 nodes	*10 nodes*	*13 nodes*	*7 nodes*	*10 nodes*	*13 nodes*
3	4510,69	4526,24	4531,9	4525,6	4454,41	4570,38
4	4607,27	4633,97	4638,86	4612,91	4534,27	4664,18
5	4569,91	4567,92	4579,85	4619,24	4579	4652,31
6	4532,3	4542,49	4548,66	4552,11	4483,43	4594,29
7	4446,75	4450,71	4457,86	4474,01	4413	4511,9
10	4364,93	4367	4376	4400,75	4351,07	4435,79
11	4395,31	4408,09	4413,45	4410,73	4341,33	4453,4
12	4416,65	4416,97	4425,31	4450,64	4396,55	4485,8
13	4375,19	4384,38	4390,93	4397,38	4334,73	4437,41
14	4303,08	4303,33	4314,2	4347,54	4308,84	4379,91
17	4288,28	4303,75	4308,3	4299,4	4228,79	4342,85
18	4311,08	4329,6	4333,6	4318,4	4244,3	4363,82
19	4311,08	4329,6	4333,6	4318,4	4244,3	4363,82
20	4344,18	4377,45	4382,1	4346	4272,6	4398,06
21	4413,04	4441,63	4446,2	4417	4341,97	4467,57
24	4488,47	4519,71	4524,5	4491,8	4415,64	4544,09

25	4577,91	4594,2	4599,3	4590,1	4514,45	4636,21
26	4530,1	4531,54	4542,1	4572,4	4526,5	4607,6
27	4560,62	4576,32	4582	4575,7	4503,66	4620,96
28	4584,93	4594,71	4601,64	4607,88	4541,59	4649,84

Based on Tables 8 and 9, Table 8 shows the prediction results for TLKM shares, while Table 8 shows the prediction results for BRI shares. The two prediction results use two different prediction methods and ANN model configurations. The column Oct 2022 shows the date the prediction was made, while columns 7, 10 and 13 nodes represent the number of nodes or neurons in the hidden layer, which are ANN configurations. The prediction target for TLKM stocks is shown in Table 9, column Target (IDR). For example, the first row in Table 7 means that a prediction was made on 3 October 2022. The results show that using the saved model prediction method with seven nodes generates IDR 4404.88, using ten nodes generates IDR 4407.06, using 13 nodes generates IDR 4416.13.

C. The proposed framework evaluation results

The best configuration can be identified by calculating the error value of the resulting predictions against the target value, which is the actual closing price of the stock. The error calculations used are MSE, RMSE and MAPE. The lower the MAPE, the better the model. Tables 10 and 11 also show the error values resulting from each test.

Table 10. Evaluation metrics on TLKM predictions

Prediction Method	Configuration	MSE	RMSE	MAPE (%)
Saved model prediction	3 hidden layer, 7 node	1215,91	34,87	0,68
	3 hidden layer, 10 node	1109,38	33,31	0,67
	3 hidden layer, 13 node	1275,96	35,72	0,71
Current model prediction	3 hidden layer, 7 node	1873,78	43,29	0,89
	3 hidden layer, 10 node	4086,80	63,93	1,08
	3 hidden layer, 13 node	1052,31	32,44	0,62

Based on Table 10, for the smallest MAPE in predicting TLKM shares, the saved model prediction method using 3 hidden layers with 10 nodes has a MAPE of 0.67%. Meanwhile, the current model prediction method for the lowest MAPE obtains the lowest MAPE using 3 hidden layers with 13 nodes, which is 0.62%. Table 11 also shows the metric evaluation when predicting BRI stocks.

Table 11. Evaluation merics on BRI predicitons

Prediction Method	Configuration	MSE	RMSE	MAPE (%)
Saved model prediction	3 hidden layer, 7 node	653,48	25,56	0,49
	3 hidden layer, 10 node	501,80	22,40	0,42
	3 hidden layer 13 node	668,55	25,86	0,49
Current model prediction	3 hidden layer, 7 node	1414,80	37,61	0,69
	3 hidden layer, 10 node	4577,56	67,66	1,30
	3 hidden layer 13 node	4141,52	64,35	1,29

Based on Table 11, the saved model prediction method for the smallest MAPE in predicting BRI shares using 3 hidden layers with 10 nodes has a MAPE of 0.42%. Meanwhile, the current model prediction method for the smallest MAPE obtains the smallest MAPE using 3 hidden layers with 7 nodes, which is 0.69%.

Based on Tables 10 and 11 with the similar ANNs model configuration and tested on different stocks (TLKM and BRI), the saved model prediction method has the best results using the ANNs model configuration consisting of 3 hidden layers with ten nodes. The MAPE value of 0.42% is evident in the prediction of BRI shares and the MAPE of 0.67% in the prediction of TLKM shares. Both are the lowest MAPE values. The current model prediction method has the best results using the ANNs model configuration consisting of 3 hidden layers with seven nodes. The MAPE value of 0.89% for the TLKM stock prediction and 0.69% for the BRI stock prediction prove this. To test the performance of the ANN model, we use the MAPE accuracy scale as shown in Table 4. From the table, we know that a MAPE of 10% means that the ANNs model has high accuracy.

In addition to calculating the performance of the built ANN model, we observe the performance of the proposed framework by observing the processing time required for the proposed framework to process a prediction request. Furthermore, Table 12 shows the time required by the proposed framework for a prediction process.

Table 12. Proposed Framework processing time

Code	Processing time (seconds)	
	Saved model	*Current model*
TLKM	0,52 – 0,83 seconds	120 – 200 seconds
BRI	0,53 – 1,0 seconds	135 – 220 seconds

D. Benchmarking Analysis

The best ANN model and parameters we built were then compared with models in previous studies developed by [7]. The evaluation aims to determine the error value of each configuration. We used the dataset for performance testing as shown in **Tables 5** and **6**. Furthermore, **Table 13** shows the parameters we used for the evaluation, which are the best according to the evaluation results in Section 3.3.

Table 13. Proposed ANN parameter configuratiions

Prediction Method	Configuration
Saved model prediction	(a) 1 input layer; (b) 3 hidden layer with 10 node; (c) 1 output layer; (d) Activation function: ReLu; (e) Optimizer: Adam; (f) Epoch: 1000.
Current model prediction	(a) 1 input layer; (b) 3 hidden layer with 7 node; (c) 1 output layer; (d) Activation: ReLu; (e) Optimizer: Adam; (f) Epoch: 100

Based on **Figure 4**, our proposed model is able to produce better MAPE values than Research [7]. The model built by Research [7] produces a MAPE of 1.38% for TLKM shares and 0.83% for BRI shares. In contrast, the current model prediction method has a MAPE of 0.67% for TLKM shares and 0.42% for BRI shares. The current model prediction method has a MAPE value of 0.69% for TLKM shares and 0.89% for BRI shares. In general, the ANN model built by research [7] and our model have a MAPE value of < 10%, are classified as good and fall into the high accuracy category, as shown in **Table 4**.

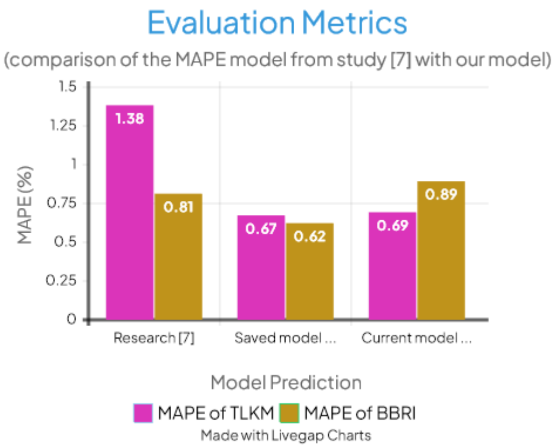

Figure 4. Comparison of MAPE value

Conclusion

Tables 9 and **10** show the best ANN model configuration suitable for each prediction method in the proposed framework, based on the smallest MAPE value produced. The saved model prediction method has the best configuration consisting of 1 input layer, 3 hidden layers with 10 nodes and 1 output layer. In contrast, the current prediction model has the best ANN model configuration consisting of 1 input layer, 3 hidden layers with 7 nodes and 1 output layer. With the best parameter configuration, the two prediction methods have a MAPE value of <10%, which means that the ANN model has high accuracy. The saved model prediction method has a MAPE value of 0.42% and 0.67%. Meanwhile, the current model prediction has a MAPE value of 0.69% and 0.89%. Next, based on Figure 4, the ANN models we built have better performance than the model developed by the research [7]. Furthermore, the runtime required for the proposed framework to process a prediction request to completion is impressive, as shown in **Table 11**. In the saved model prediction, the longest processing time is up to 1 second, while in the current model prediction it is up to 220 seconds.

Acknowledgement

The authors would like to thank Universitas AMIKOM Purwokerto, Universiti Teknikal Malaysia Melaka, and the Amikom Purwokerto Intelligent System (AI-SYS) research group for their assistance this research.

References

[1] I. Sudarman and N. Diana, "The Effect of Financial Ratios on Sharia Stock Prices in Companies in the LQ45 Index 2020-2021," *J. Ilm. Ekon. Islam*, vol. 8, no. 1, p. 117, 2022.

[2] J. Jefry and A. Djazuli, "The Effect of Inflation, Interest Rates and Exchange Rates on Stock Prices of Manufacturing Companies in Basic and Chemical Industrial Sectors on the Indonesia Stock Exchange (IDX)," *Int. J. Business, Manag. Econ. Res.*, vol. 1, no. 1, pp. 34–49, 2020.

[3] S. M. Prabin and M. S. Thanabal, "A repairing artificial neural network model-based stock price prediction," *Int. J. Comput. Intell. Syst.*, vol. 14, no. 1, pp. 1337–1355, 2021.

[4] S. S. Abubaker and S. R. Farid, "Stock Market Prediction Using LSTM," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 8.5.2017, pp. 2003–2005, 2022.

[5] P. Chhajer, M. Shah, and A. Kshirsagar, "The applications of artificial neural networks, support vector machines, and long–short term memory for stock market prediction," *Decis. Anal. J.*, vol. 2, no. November 2021, p. 100015, 2022.

[6] M. Jufri, "Implementation of Artificial Neural Network in Predicting Birth Rate in Batam City Using Backpropagation Method," *JURTEKSI (Jurnal Teknol. dan Sist. Informasi)*, vol. 8, no. 1, pp. 85–94, 2021.

[7] Melina, Sukono, H. Napitupulu, A. Sambas, A. Murniati, and V. A. Kusumaningtyas, "Artificial Neural Network-Based Machine Learning Approach to Stock Market Prediction Model on the Indonesia Stock Exchange During the COVID-19," *Eng. Lett.*, vol. 30, no. 3, pp. 988–1000, 2022.

[8] S. S. Al Al-Hasnawi and L. H. Al Al-Hchemi, "Stock Closing Price Prediction of ISX-listed Industrial Companies Using Artificial Neural Networks," *Am. J. Bus. Oper. Res.*, vol. 6, no. 2, pp. 47–55, 2022.

[9] O. M. A. AL-atraqchi, "A Proposed Model for Build a Secure Restful API to Connect between Server Side and Mobile Application Using Laravel Framework with Flutter Toolkits," *Cihan Univ. Sci. J.*, vol. 6, no. 2, pp. 28–35, 2022.

[10] S. Mowla and S. V Kolekar, "Development and integration of E-learning services using rest APIs," *Int. J. Emerg. Technol. Learn.*, vol. 15, no. 4, pp. 53–72, 2020.

[11] G. Susrama, M. Diyasa, G. S. Budiwitjaksono, H. Amarul, and I. Ade, "Comparative Analysis of Rest and GraphQL Technology on Nodejs-Based Api Development," vol. 2021, pp. 1–9, 2021.

[12] U. Singh, "REST API Framework : Designing and Developing Web Services," *Int. Res. J. Eng. Technol.*, vol. 8, no. June, pp. 815–817, 2021.

[13] S. Y. M, S. H. R, and R. Nagapadma, "Survey Paper: Framework of REST APIs," *Int. Res. J. Eng. Technol.*, no. June, pp. 1115–1119, 2020.

[14] M. A. Khder, "Web scraping or web crawling: State of art, techniques, approaches and application," *Int. J. Adv. Soft Comput. its Appl.*, vol. 13, no. 3, pp. 144–168, 2021.

[15] K. BABU, "Survey on Web scraping technology," *Waffen-Und Kostumkd. J.*, vol. 16, no. 06, pp. 1–8, 2020.

[16] C. Fan, M. Chen, X. Wang, J. Wang, and B. Huang, "A Review on Data Preprocessing Techniques Toward Efficient and Reliable Knowledge Discovery From Building Operational Data," *Front. Energy Res.*, vol. 9, no. March, pp. 1–17, 2021.

[17] A. P. Joshi and B. V Patel, "Data Preprocessing: The Techniques for Preparing Clean and Quality Data for Data Analytics Process," *Orient. J. Comput. Sci. Technol.*, vol. 13, no. 0203, pp. 78–81, 2021.

[18] A. G. Farizawani, M. Puteh, Y. Marina, and A. Rivaie, "A review of artificial neural network learning rule based on multiple variant of conjugate gradient approaches," *J. Phys. Conf. Ser.*, vol. 1529, no. 2, 2020.

[19] A. Duykuluoğlu, "The Significance of Artificial Neural Networks in Educational Research : A Summary of Research and Literature," vol. 2, no. 2, pp. 107–116, 2021.

[20] A. C. Cinar and N. Natarajan, "An artificial neural network optimized by grey wolf optimizer for prediction of hourly wind speed in Tamil Nadu, India," *Intell. Syst. with Appl.*, vol. 16, no. March 2021, p. 200138, 2022.

[21] R. R. Waliyansyah and N. D. Saputro, "Forecasting New Student Candidates Using the Random Forest Method," *Lontar Komput. J. Ilm. Teknol. Inf.*, vol. 11, no. 1, p. 44, 2020.

[22] J. Qi, J. Du, S. M. Siniscalchi, X. Ma, and C. H. Lee, “On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression,” *IEEE Signal Process. Lett.*, vol. 27, pp. 1485–1489, 2020.

[23] M. del R. C. Estrada, M. E. G. Camarillo, M. E. S. Parraguirre, M. E. G. Castillo, E. M. Juárez, and M. J. C. Gómez, “Evaluation of Several Error Measures Applied to the Sales Forecast System of Chemicals Supply Enterprises,” *Int. J. Bus. Adm.*, vol. 11, no. 4, p. 39, 2020.

[24] C. C. Aggarwal, *Neural Networks and Deep Learning*. 2018.