AutoKeras and particle swarm optimization to predict the price trend of stock exchange

Doaa A. Fattah, Amany A. Naim, Abeer S. Desuky, Mervat S. Zaki
Department of Mathematics, Faculty of Science, Al-Azhar University, Cairo, Egypt

Article Info

Article history:

Received Nov 16, 2021
Revised Feb 22, 2022
Accepted Mar 14, 2022

Keywords:

Auto machine learning
Deep neural network
Particle swarm optimization
Prediction
Stock predictions

ABSTRACT

The stock price varies depending on time, so stock market data is time-series data. The prediction of the trend of a stock price is a more interesting topic for investors to take an investment decision in a specific stock. Prediction of stock price always depends on machine learning algorithms. In this work, optimizing deep neural network (DNN) is used for predicting if the close price is reached to the profit which is determined by the investor or not and improve the prediction accuracy. Particle swarm optimization (PSO) and auto machine learning (AutoML) are used as optimizers with DNNs. The methods are applied to data of nine companies in Indonesia and National Stock Exchange (NSE) of India. The data is got from yahoo finance. Based on the experimental results, AutoML of deep learning proved to have the best accuracy rate, which is varying from 81 percent to 92 percent across all companies, and the accuracy after optimizing DNNs using PSO is varying from 73 percent to 82 percent across all companies.

Corresponding Author:

Doaa A. Fattah
Department of Mathematics, Faculty of Science, Al-Azhar University
Cairo, Egypt
Email: doaa.fattah1@gmail.com

1. INTRODUCTION

The stock market is an important component of the economy of any country. This represents one of the most significant investment opportunities. This is a great time for investors to buy shares and earn more profits. Investors need to predict the trend of stock price to decide to buy or sell stocks [1]. As the investment in the stock exchange has a risk, the researchers try to find a model to reduce the time and the cost of predicting the trend of the price. The traditional methods to make this prediction consume a long time and resources so automated ways using data mining algorithms are applied. To make a profit, investors must buy the shares that are expected to increase in price in the near future and sell the shares that are expected to decrease in price. Predicting the trend of a stock price is extremely difficult to challenge as the stock data contains incomplete, fuzzy, complex information, and a huge amount of data has to be processed in a relatively short time. If the approach used to predict the trend has high accuracy, the investors can realize significant profit [2], [3]. Before machine learning, researchers used various statistical and econometric models to build a model in researches. A conventional statistical model cannot be used to predict and analyze nonlinear data, so we need to change the nonlinear to linear models [4]. Nowadays, predicting the direction of a stock price is one of the machine learning applications. Many machine learning algorithms such as support vector machine (SVM), decision tree, deep neural network (DNN), and others have been already used to predict the trend of stock prices [5].

Neural networks (NNs) have a high capacity to analyze noisy data and have been extensively used to predict time series. Deep learning (DL) was applied to NNs to construct DNNs with multiple hidden layers to generate the nonlinear relationships from nonlinear data by using the nonlinear activation function. DNN was used to solve nonlinear problems more satisfactorily than other machine learning algorithms [6], [7]. The algorithms are trained on historical stock data to predict the direction of the stock price shortly.

In this paper, the main aim is to predict if the close price reached to profit determined by investors or not based on historical price (e.g., open and volume) using a DNN. Particle swarm optimization (PSO) and AutoKeras are used separately to optimize DNN performance. Figure 1 illustrates the objective of the paper. Figure 1 refers to information which determines the trend of close price for each day after the investor read this information (the output is 0 this mean that the close price does not increase than the open price of the day with specific profit which is determined by the investor, and 1 if the close price is increased. Based on this, the investors take the action to buy stocks, hold trading or sell stocks.

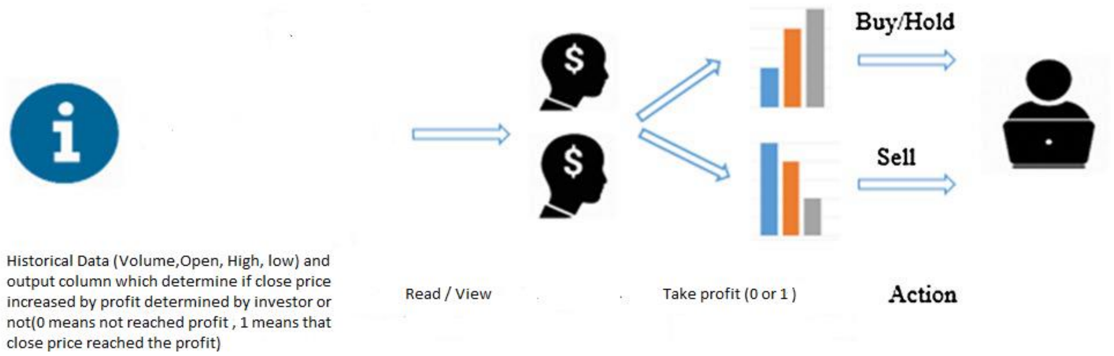

Figure 1. The target of the paper

In this field of the stock exchange, some research articles have been developed during the past years in terms of price prediction of stock as it is an important application for investors to utilize investing their money. Recently, Yu and Yan [8] proposed financial product price data were treated as a one-dimensional series generated by projecting a chaotic system consisting of multiple factors in the time dimension, and the price series were reconstructed using the time series phase-space reconstruction (PSR) method. A prediction model based on DNN was developed using PSR and long-and short-term memory networks (LSTMs) for DL.

Cagliero *et al.* [9] used methods to combine faceted classification models for supporting stock trading. In order to achieve this, separate classification models were created on each subgroup of features belonging to the same facet. The authors produced trading signals tailored to a specific facet. Then, signals were combined and filtered to generate a unified, multi-faceted recommendation. The experimental validation, carried out on different markets and in different conditions, shows that, in many cases, some of the faceted templates work as well or better than the templates formed on a mixture of different features. A set of faceted recommendations makes the generated trading signals more profitable but robust to drawdown periods.

The study by Khan *el al.* [10] was performed on social media and financial news data. algorithms were used to check the impact of this data on the accuracy of predictions of stock market for ten days. To improve the performance of predictions, feature selection and spam tweet reduction were performed on the data sets. Moreover, experiments were conducted to find such stock markets that are difficult to predict and those that were more impacted by social media and financial news by comparing the results of different algorithms to find a consistent classifier. DL was used, and some classifiers were grouped to achieve maximum prediction accuracy. The results showed that the highest predictive accuracy of 80.53% and 75.16% is obtained through social media and financial news, respectively. The New York and Red Hat stock markets were difficult to forecast and IBM stocks were more influenced by social media, while London and Microsoft stocks were more influenced by financial news. The random forest classifier was considered coherent and the highest precision of 83.22% was achieved by its whole.

The ensemble method is to build predictive models by combining the strengths of the classical classification method [11]. In this research, the purpose of ensemble based on Boosting for Regression appeared to enhance simple tree analysis and deals with some of the weaknesses found in uncomplicated techniques. The ensemble tree combines the prediction values of many simples trees into a single prediction value. Based on the experiments that have been carried out, the ensemble method proved to have a better accuracy rate, which amounted to 82%. The model has been assumed by the ensemble model that can get the relationship between variables to be more precise than the previous model.

2. METHOD

Two models based on optimization of a neural network are proposed to predict if the close price of the day is greater than the open price with specific profit determined by the investor (class with value 1) or not (class with value 0). The models are applied on historical data of stock prices for nine of Indonesia's companies for five years from 2015 to 2020. The steps are shown in Figure 2.

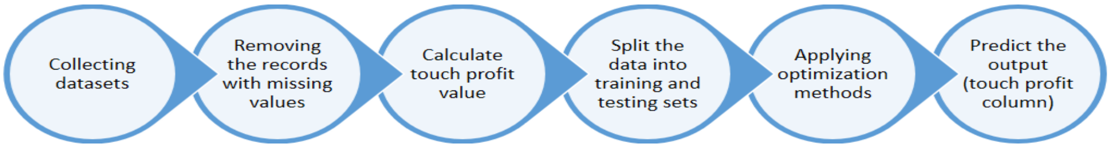

Figure 2. The steps are done in the paper

2.1. Dataset description

The dataset contains four features, which are open, volume, min, and max. Where open indicates the open price of the trading day, volume indicates the volume of trading in the day, min indicates the minimum price of the trading day, and max indicates the maximum price of the trading day. The input values are continuous data. Figure 3 represents a sample of historical price data in 2020 for JKSE company.

Date	Open	High	Low	Volume
2019-12-13	6181.055176	6197.317871	6167.640137	48546800.0
2019-12-16	6197.313965	6237.916016	6197.313965	55299700.0
2019-12-17	6223.442871	6244.352051	6205.627930	53264500.0
2019-12-18	6287.250000	6287.250000	6287.250000	0.0
2019-12-19	6274.388184	6281.416016	6235.844238	47624800.0
2019-12-20	6257.160156	6284.372070	6231.465820	52766200.0
2019-12-23	6309.670898	6315.721191	6270.539063	39077900.0
2019-12-26	6303.059082	6326.268066	6303.059082	33608900.0
2019-12-27	6321.568848	6337.335938	6312.380859	42100200.0
2019-12-30	6329.134766	6336.919922	6289.546875	47557400.0

Figure 3. Sample of JKSE data

The price of stocks is a time-series data, Figure 4 represents JKSE's close price from 2015 to 2019. The results are reported for nine companies related to the Indonesia stock exchange and the national stock exchange in India. The historical data has been collected from yahoo finance [12]. Indonesia companies were chosen to prove that the accuracy is optimized using AutoML for DL by comparing the accuracy with previous research that was applied to the same data. The dataset for Indonesia's companies includes five-year data from 1/2015 to 12/2019 of *PT Kimia Farma Tbk (PKF), PT Bank Negara Indonesia Tbk (PBNI), PT Perusahaan Perkebunan London Sumatra Indonesia Tbk (PPPLSI), PT Unilever Indonesia Tbk (PUI), PT Astra International Tbk (PAI), PT Indofarma Tbk (PI), PT Hanjaya Mandala Sampoerna Tbk (PHMS), and PT Telkom (PT)*.

Figure 4. The JKSE price for 1200 days since 2015

2.2. Data pre-processing

There are two steps for making pre-processing. These steps are cleaning the dataset from missing values and the second is calculating a new feature. The first step is to clean up the data, if any price is missing, the record of this trading day will remove. The next step calculates the touch profit feature by using open and close prices for each day with (1):

$$\text{Calculated price} = \text{open price} + (\text{open price} \times \text{profit}) \tag{1}$$

Where (1) calculates the price which the investor needs to reach it using the determined profit (it is determined by the investor).

The profit percentage is specified from the opening price before applying the model. If the calculated price is equal to or greater than the close price for the day, this means that the profit is touched, and the touch profit value will be one. If not, the touch profit value will be zero. Figure 5 represents a sample of historical price data in 2020 and the calculated column (touch_tp) which represents if this trading day reached the profit which the investor needs to be achieved or not reach. In other words, Figure 3 is converted to Figure 5. Figure 6 presents the flowchart for data pre-processing.

Date	Open	High	Low	Volume	touch_tp
2019-12-13	6181.055176	6197.317871	6167.640137	48546800.0	1
2019-12-16	6197.313965	6237.916016	6197.313965	55299700.0	1
2019-12-17	6223.442871	6244.352051	6205.627930	53264500.0	1
2019-12-18	6287.250000	6287.250000	6287.250000	0.0	1
2019-12-19	6274.388184	6281.416016	6235.844238	47624800.0	0
2019-12-20	6257.160156	6284.372070	6231.465820	52766200.0	1
2019-12-23	6309.670898	6315.721191	6270.539063	39077900.0	1
2019-12-26	6303.059082	6326.268066	6303.059082	33608900.0	1
2019-12-27	6321.568848	6337.335938	6312.380859	42100200.0	1
2019-12-30	6329.134766	6336.919922	6289.546875	47557400.0	0

Figure 5. Sample of JKSE data after pre-processing

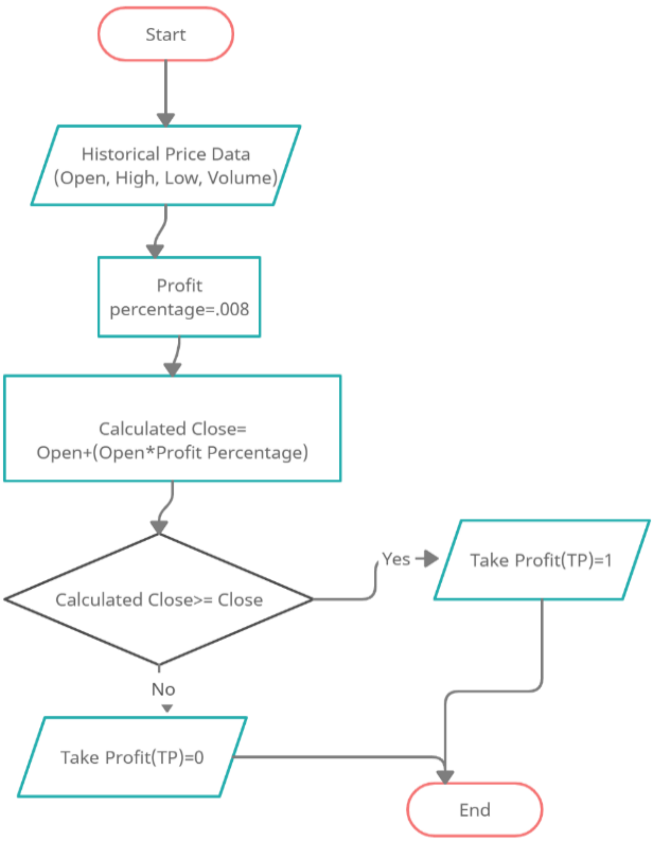

Figure 6. Flowchart for data pre-processing

2.3. Deep neural network

NNs are algorithms that compute the y-out from the x-input. they can be applied in images, videos and texts. In mathematics, such an algorithm defines a function f (i.e., y=f(x)). The computer used to calculate this function contains several stages, and each stage performs calculations like additions, multiplications, and a maximum. In comparison, the programs found in the operating system are more complex. The algorithm of the neural network depends on parameters, which are the weights of the neurons. These weights must be modified using mathematical and algorithmic methods so that the algorithm can best solve the requested task before using a neural network. This process is called “training” a neural network that requires a lot of time, calculations, and energy. It is therefore necessary to manipulate key concepts in computer science (iterative methods calculating temporal memory space effective implementation, and mathematics (linear algebra, optimization, and statistics) [13].

Assuming we have only two layers of neurons (the first layer between x and u, the second between u and y) as x refers to input, y refers to output, u refers to the hidden layer, but most efficient networks can have several hidden layers, we say that they are deeper. For example, assume we have three inputs x_1, x_2, x_3, but there can be more) and one hidden layer (neuron) u_1. The formula calculated by the neuron is $u_1=\max(w_1 \times x_1+w_2 \times x_2+w_3 \times x_3+b, 0)$. The neuron thus performs a weighted sum of the three inputs, with three weights w_1, w_2, w_3, and adding b, which is a bias. Then the neuron calculates the maximum between this sum and zero. The maximum function is the most popular but other functions can be applied. It is a thresholding operation. If the weighted sum $w_1x_1+w_2x_2+w_3x_3+b$ is smaller than 0, then the neuron returns the value $u_1=0$, otherwise it returns the value of this sum and places it in u_1 [14].

Neural network (NN) is most appropriate for stock price prediction [15], so in this section DNN, AutoKeras, and PSO are illustrated as a general before explaining the proposed method. NNs are an important area of the machine learning field and they are a type of data-driven algorithm. They are adaptive, have a relatively strong capacity to approximate nonlinear functions. NNs are the precursor of DNNs [8]. DNN is a neural network that has at least one hidden layer. It is more suitable for providing models for complicated nonlinear functions and has a high-level abstraction ability, which means that the power of the provided model is significantly improved. Meanwhile, it is a kind of discriminant model which could be trained through the backpropagation algorithm. Since the DNN is more appropriate for prediction problems with large data sets and complicated nonlinear mapping relations, so it can be used in stock trend prediction [16].

2.4. Proposed methods

In this paper two techniques related to the neural network are applied, the first technique is AutoKeras, and the second technique is optimizing the network’s biases and weights using PSO.

2.4.1. Auto machine learning

AutoML is a technique that automatically detects the most powerful model for a particular dataset. Applying this technique to NNs contains discovering the model architecture and the hyperparameters used to train the model. It is based on a neural architecture search algorithm (NAS). This technique provides the best combination of data preparation and hyperparameter models for a predictive modeling problem. This technique has more trials specified in the model to search and select the best architecture and parameters of neural network which achieved the best accuracy [17], [18]. Figure 7 summarizes the NAS algorithm.

Figure 7. Dimensions of neural architecture search (NAS)

AutoKeras is an open-source library to perform AutoML for DL models using Kera’s application programming interface (API) (released in 2019) in Python. The search is performed with so-called Keras models through the TensorFlow tf.keras API. It provides an effective and easy technique for finding top-performing models automatically for a wide range of predictive modeling tasks, comprising tabular or supposedly structured classification and regression datasets [19]. These are the steps of our model:

Step 1: collecting datasets.

Step 2: calculate the take profit value.

Step 3: pre-processing data by handling the missing values in the dataset.
Step 4: splitting dataset to input and output elements. Then splitting the data into training and testing data, and make sure that the training dataset is large than the test set as shown in the Table 1.
Step 5: applying AutoKeras technique using number of trials twenty to search for the NN model with architecture has the best accuracy for each data (noted that the architecture of neural network varies from data of the company to another)
Step 6: analyzing the results of the ensemble model performance in terms of its accuracy (running the program for ten times and calculate the mean of accuracies).

Table 1. Statistics of the Indonesia's stock dataset

	Dataset	Training data	Testing data
Time interval	01/01/2015-31/12/2019	01/01/2015-31/12/2018	01/01/2019-31/12/2019

2.4.2. Optimizing network's biases and weights using PSO

PSO depends on the intelligence of the swarm. It is considered one of the evolutionary computational algorithms [20]. PSO is an optimization technique based on population, inspired by the motion of bird flocks and schooling fish. It is similar to evolutionary computation techniques. The system is initialized with a population of random solutions and then updates generations to find the optimal solution [21]. There are no evolution operators in PSO, such as crossover and mutation in a genetic algorithm. In PSO the possible solutions are called particles move into the problem area following the current optimal particles. PSO is considered more efficient based on speed and memory requirements [22]. In standard PSO which is presented in Figure 8, the new location of each particle is determined by a velocity term, which reflects the attraction of global best (g_b) and its own best (o_b) during the history of the particle and random coefficients [23].

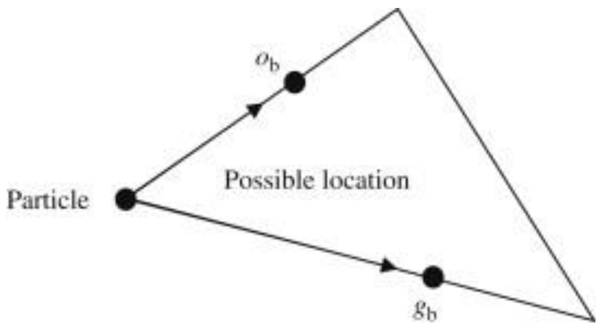

Figure 8. Standard PSO

In this technique, a neural network is optimized by optimizing the network's weights and biases. These are the steps of our model:
Step 1: load the data.
Step 2: build neural network as: i) input layer contains four inputs (open price, high price, low price, volume; ii) hidden layer with size 20 using activation function tanh; iii) output layer with size 2 (zero, one).
Step 3: create a method to do forward propagation for one particle by create a swarm with several dimensions equal to the weights and biases and scroll these parameters into an n-dimensional array, and have each particle take on different values, so each particle represents a candidate neural network with its weights and bias. When feeding back to the network, the learned weights and biases are reconstructed. Then, recall the shape and bias matrices: i) shape of input-to-hidden weight matrix: (4, 20) as a number of input layers is 4 and number of hidden layers is 20; ii) shape of input-to-hidden bias array: (20,); iii) shape of hidden-to-output weight matrix: (20, 2) as a number of output layers is 2 and number of hidden layers is 20; iv) shape of hidden-to-output bias array: (2,). So, we have (4x20)+(20x2)+20+2=142 parameters, or 142 dimensions for each particle in the swarm. We did not perform backpropagation because PSO does not rely on the gradients.
Step 4: create an objective function to compute forward propagation for the whole swarm.
Step 5: performing PSO on the custom function: i) initialize swarm; ii) call instance of PSO by sending the number of particles (assumed: 100), dimensions (number of parameters calculated: 142), options (initialized weights (four weights as we have four inputs)) to GlobalBestPso function in python; iii) perform optimization using function optimize in python and send the method to do forward propagation as a parameter to it.
Step 6: checking the accuracy.

3. EXPERIMENTAL RESULTS

This section validates the efficiency of using AutoML and PSO methods in predicting stock price trends. Two experiments are designed and reported associated results. The first one uses AutoML. The other experiment was done using PSO which optimize the neural network. The results were compared among all experiments. Also, the results were compared with the results of another research [11]. This research used Boosted regression tree (BoostRT) models which is a combination of two techniques: decision tree algorithms and boosting. All results were calculated based on a set of standard evaluation metrics such as overall accuracy, recall, and precision.

3.1. Performance evaluation metrics

Accuracy, Recall, Precision and are used to indicate the performance of the model. When the model correctly predicts the positive category, in this case, the result is a true positive (TP), and likewise, when the model correctly predicts the negative category the positive category, in this case, the result is a true positive (TP), and likewise, when the model correctly predicts the negative category, the negative result is true (TN). When the model incorrectly predicts the positive category, in this case, the result is a false positive (FP), and likewise, when the model incorrectly predicts the negative category, the negative result is false (FN). From that we can calculate accuracy the precision and recall to indicate the performance of our model [24].

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{2}$$

In (2) represents accuracy which is a measure of the rating model's performance. In other words, it is part of the predictions that our model got correctly

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{3}$$

In (3) represents precision which is a proportion of positive identifications that are correct.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{4}$$

In (4) represents recall which is the proportion of actual positives that are identified correctly.

3.2. Indonesia's data prediction using AutoML and PSO

In this experiment, both AutoML, PSO, and BoostRT [11] models are compared for the stock prices dataset of nine companies. Tables 2-4 represent the compared performances of AutoML, PSO, and BoostRT upon nine companies of Indonesia's exchange. This comparison shows that AutoML has the highest performance for all companies for example: for PHMS Company the accuracy, recall, and precision are 87%, 87%, and 74% for AutoML method while it is 82%, 81%, and 70% for PSO and 82%, 82%, and 71% for BoostRT. The illustration of observations for companies which is used in Tables 2-4. Figure 9 represents a comparison between accuracies of AutoML, PSO, and BoostRT upon Indonesia's exchange

Table 2. Accuracy of models upon Indonesia's exchange

Method	PKF	PBNI	PPPLSI	PUI	PAI	PI	PHMS	PT
AutoML	81%	87%	84%	85%	85%	83%	87%	84%
PSO	74%	73%	76%	78%	76%	73%	82%	75%
BoosRT [11] in 2020	76%	81%	79%	82%	82%	80%	82%	78%

Table 3. Recall of models upon Indonesia's exchange

Method	PKF	PBNI	PPPLSI	PUI	PAI	PI	PHMS	PT
AutoML	79%	87%	84%	85%	85%	83%	87%	84%
PSO	72%	70%	72%	77%	76%	73%	81%	76%
BoosRT [11] in 2020	75%	79%	79%	82%	82%	80%	82%	78%

Table 4. Precision of models upon Indonesia's exchange

Method	PKF	PBNI	PPPLSI	PUI	PAI	PI	PHMS	PT
AutoML	76%	81%	74%	81%	92%	84%	74%	79%
PSO	70%	69%	69%	72%	75%	70%	70%	71%
BoosRT [11] in 2020	73%	73%	70%	73%	88%	82%	71%	76%

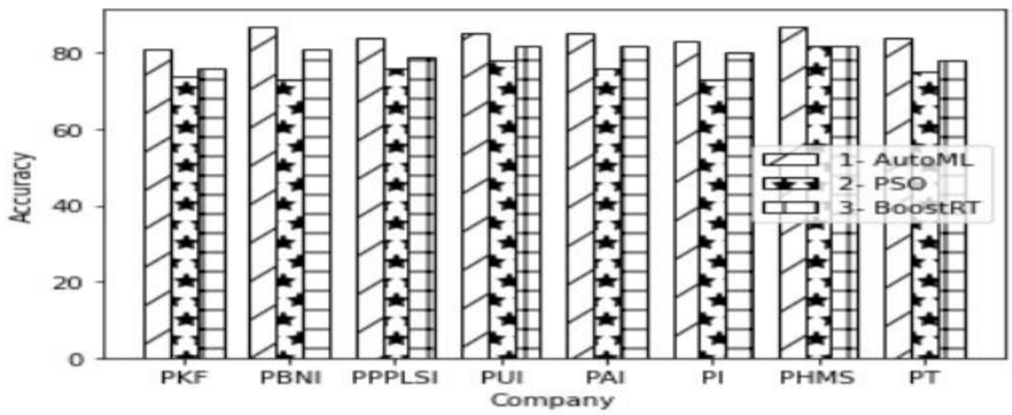

Figure 9. Accuracy of models upon Indonesia's exchange

3.2. Comparison between the results of the proposed method and other techniques

By applying linear discriminant analysis, logistic regression, quadratic discriminant analysis, KNN, Naïve bayes, BoosRT, PSO and AutoML on JKSE dataset. The results are as shown in Table 5 which represents that AutoML has the highest performance using accuracy, Recall and precision indicators as the performance is 92%, 94 %, and 89% respectively. Also, the accuracy of AutoML is higher than the highest accuracy of the rest applied methods. Also, when applying Logistic regression, K-nearest neighbor, Decision tree, bagging, boosting, random forest, Artificial neural network, Support vector machines, PSO and AutoML on National stock exchange of India dataset. The results represent that AutoML has the highest accuracy than the highest accuracy of the rest models which is reached 94%.

Table 5. Accuracy of AutoML and other machine learning algorithms upon JKSE and national stock exchange (NSE) of India

DataSets	Method	Accuracy	Recall	Precision
JKSE	Logistic regression [11] in 2020	54%	14%	61%
	Linear discriminant analysis (LDA) [11] in 2020	54%	16%	52%
	Quadratic discriminant analysis (QDA) [11] in 2020	55%	52%	51%
	KNN [11] in 2020	65%	63%	62%
	Naïve bayes [11] in 2020	77%	76%	77%
	BoosRT [11] in 2020	82%	83%	71%
	Adaptive neuro-fuzzy inference system (ANFIS) [25]	77.3%	-	-
	Fuzzy Kernal C-means (FKCM) [25]	83.7%	-	-
	PSO	82%	77%	81%
	AutoML	92%	94%	89%
National stock exchange (NSE) of India	Logistic regression [26]	89.91%	-	
	K-nearest neighbor [26]	68.73%	-	-
	Decision tree [26]	90.83%	-	-
	Bagging [26]	90.37%	-	-
	Boosting [26]	92.10%	-	-
	Random forest [26]	91.32%	-	-
	Artificial neural network [26]	71.62%	-	-
	Support vector machines [26]	90.57%	-	-
	PSO	70.95%	-	-
	AutoML	94.01%	-	-

4. CONCLUSION

In this work, AutoML and PSO were proposed to optimize the DNN to predict if the close price of the day reached profit determined by the investor before investing or not. They were applied on datasets of historical prices and the results compared to state-of-the-art methods upon JKSE and NSE of India datasets.AutoML achieved the highest accuracy, precision, and recall which reached 92%, 92%, and 89% respectively. This highest performance encourages the investors to use this model to predict if they will achieve profits or not as the risk is reduced to 8% which is a small percentage by comparing it with the state-of-the-art method. With Regards PSO, it does not achieve a good accuracy as it is reached 82%, so the risk is 18%. So, we recommend using AutoML for prediction when dealing with stock prices data but PSO is not recommended for this type of data.

REFERENCES

[1] A. K. Nassirtoussi, S. Aghabozorgi, T. Y. Wah, and D. C. L. Ngo, "Text mining of news-headlines for FOREX market prediction: A multi- semantics and sentiment," *Expert Syst. Appl.*, vol. 42, no. 1, pp. 306 –324, 2015, doi: 10.1016/j.eswa.2014.08.004.

[2] Chih-FongTsai and Yu-Chieh Hsiao," Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches," *Decis. Support Syst.*, vol. 50, no. 1, pp. 258–269, 2010, doi: 10.1016/j.dss.2010.08.028.

[3] V. Govindasamy and P. Thambidurai, "Probabilistic fuzzy logic-based stock price prediction," *Int. J. Comput. Appl.*, vol. 71, no. 5, pp. 28-32, 2013, doi: 10.5120/12356-8669.

[4] L. Zhang, F. Wang, B. Xu, W. Chi, Q. Wang and T. Sun, "Prediction of stock prices based on LM-BP neural network and the estimation of overfitting point by RDCI," *Neural Comput. Appl.*, vol. 30, pp. 1425-1444, 2018, doi: 10.1007/s00521-017-3296-x.

[5] S. T. Ahmed, "A study on multi objective optimal clustering techniques for medical datasets," *2017 Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, 2017, pp. 174-177, doi: 10.1109/ICCONS.2017.8250704.

[6] Y. L. Cun, Y. Bengio, and G. Hinton, "Deep learning Nature ", *nature*, vol. 521, pp. 436-449, 2015, doi: 10.1038/nature14539.

[7] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85-117, 2015, doi: 10.1016/j.neunet.2014.09.003.

[8] P. Yu and X. Yan, "Stock price prediction based on deep neural networks," *Neural Comput. & Applic.*, vol. 32, pp. 1609–1628, 2020, doi: 10.1007/s00521-019-04212-x.

[9] L. Cagliero, P. Garza, G. Attanasio, and E. Baralis, "ensembles of faceted classification models for quantitative stock trading. Computing," *Springer-Verlag GmbH Austria*, vol. 102, pp. 1213–1225, 2020, doi: 10.1007/s00607-019-00776-7.

[10] W. Khan, M. Ghazanfar, M. A. Azam, A. Karami, K. H. Alyoubi, and A. S. Alfakeeh, "Stock market prediction using machine learning classifiers and social media, news," *J. Ambient Intell. Humaniz. Comput.* pp. 1-24, 2020, doi: 10.1007/s12652-020-01839-w.

[11] M. N. Tentua and D. Rosadi, "Ensemble Classification Method for Daily Return Stock Market," *Songklanakarin J. Sci. Technol. (SJST)*, vol. 43, no. 5, pp. 1-26, 2020.

[12] Serwer, *Yahoo finance*, [Online]. Available: https://finance.yahoo.com/quote/KAEF.JK/history?period1=1420070400&period2=1577750400&interval=1d&filter=history&frequency=1d&includeAdjustedClose=true

[13] G. Peyré, "Mathematics of Neural Networks," PSL, École Normale Supérieure, [Online]. Available: https://mathematical-tours.github.io/book-basics-sources/neural-networks-en/NeuralNetworksEN.pdf.

[14] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of mathematical biophysics*, vol. 5, pp. 115–133, 1943, doi: 10.1007/BF02478259.

[15] P. Gao, R. Zhang, and X. Yang," The Application of Stock Index Price Prediction with Neural Network," *Math. Comput. Appl.*, vol. 25, no. 3, p. 53, pp. 1-16, 2020, doi: 10.3390/mca25030053.

[16] W. Souma, I. Vodenska and H. Aoyama, "Enhanced news sentiment analysis using deep learning methods," *J. Comput. Soc. Sci.*, vol. 2, pp. 33–46, 2019, doi: 10.1007/s42001-019-00035-x.

[17] H. Jin, Q. Song, and X. Hu," Auto-Keras: An Efficient Neural Architecture Search System," *arXiv:1806.10282v3*, pp. 1-11, 2019.

[18] J. G. M. Perez, "Autotext: AutoML for Text Classification: Theoretical framework in Autotext: AutoML for Text Classification, Tonantzintla," Ph.D dissertation, Instituto Nacional de Astrof´ısica, Optica y Electr´onica, 2019.s

[19] AutoKeras, [Online]. Available: https://autokeras.com/.

[20] Andreas, M. H. Purnomo and M. Hariadi, "Controlling the hidden layers' output to optimizing the training process in the Deep Neural Network algorithm," *2015 Annu. IEEE Int. Conf. Cyber Technol. Autom. Control Intell. Syst. CYBER (CYBER)*, 2015, pp. 1028-1032, doi: 10.1109/CYBER.2015.7288086.

[21] M. El-Shorbagy and A. E. Hassanien, "Particle swarm optimization from theory of applications," *International journal of rough sets and data analysis*, vol. 5, no. 2, pp. 1-24, 2018, doi: 10.4018/IJRSDA.2018040101.

[22] "Swarm Intelligence: From Natural to Artificial Systems", 2020 .[Online]. Available: https://oxford.universitypressscholarship.com/view/10.1093/oso/9780195131581.001.0001/isbn-9780195131581

[23] "Training a Neural Network," 2017. [Online]. Available: https://pyswarms.readthedocs.io/en/development/examples/custom_objective_function.html

[24] Q. Song, H. Ge, J. Caverlee, and X. Hu, "Tensor completion algorithms in big data analytics," *arXiv*, vol. 13, no. 1, pp. 1-48, 2017.

[25] F. Fanita and Z. Rustam, "Predicting the Jakarta composite index price using ANFIS and classifying prediction result based on relative error by fuzzy Kernel CMeans," in *AIP Conference Proceedings*, vol. 2023, no. 1, pp. 1-7, 2018, doi: 10.1063/1.5064203.

[26] S. Mehtab and J. P. Sen, "A Time Series Analysis-Based Stock Price Prediction Using Machine Learning and Deep Learning Models," *arXiv*, pp. 1-53, 2020.

BIOGRAPHIES OF AUTHORS

Doaa A. Fattah received the B.Sc. degree in science, in 2016. She is currently a researcher in the Mathematics Department, Faculty of Science, Al-Azhar University, Cairo, Egypt. she can be contacted at email: doaa.fattah1@gmail.com.

Amany A. Naim received the B.Sc. degree in science, in 2009, and the M.Sc. and Ph.D. degrees in computer science, in 2015 and 2019, respectively. She is currently a lecrature in computer science with the Mathematics Department, Faculty of Science, Al-Azhar University, Cairo, Egypt. She has published several research papers in the field of AI, machine learning, meta-heuristic optimization, and data mining and analysis. She is also a supervisor of some master's. She can be contacted at email: amany.naim@azhar.edu.eg.

Abeer S. Desuky received the B.Sc. degree in science, in 2003, and the M.Sc. and Ph.D. degrees in computer science, in 2008 and 2012, respectively. She is currently an Associate Professor in computer science with the Mathematics Department, Faculty of Science, Al-Azhar University, Cairo, Egypt. She has published several research papers in the field of AI, machine learning, meta-heuristic optimization, and data mining and analysis. She is also a supervisor of some master's and Ph.D. theses. She is a reviewer in many Scopus-indexed journals, such as IEEE Access and Peerj. She can be contacted at email: abeerdesuky@azhar.edu.eg.

Mervat S. Zaki is a professor in pure mathematics and operations research at department of mathematics, Faculty of Science, Al-Azhar University, Cairo, Egypt. She is a reviewer for a lot of masters and Ph.D. theses at Cairo, Banha, Swiz Cannel and Helwan university. She also works as a programmer in Cairo University. She has published several research papers in Pure mathematics and OR. She is also a supervisor of some master's and Ph.D. theses. She can be contacted at email: mervatzaki.1959@azhar.edu.eg.