



A New CNN-Based Model for Financial Time Series: TAIEX and FTSE Stocks Forecasting

Melih Kirisci¹ · Ozge Cagcag Yolcu² 

Accepted: 4 February 2022 / Published online: 10 March 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Financial time series forecasting has been becoming one of the most attractive topics in so many aspects owing to its broad implementation areas and substantial impact. Because of this reason in particular recent decades, various kinds of computational intelligence techniques like convolutional neural networks (CNNs) have been used for financial time series forecasting. However, in experiments reported so far, the number of applications of CNNs for the forecasting of financial time series seems quite a few and also in almost all-studies time sequence effect of time series is not preserved on forecasts because of image transformation. From this point of view, in this paper, by aiming to get better forecasting results and avoiding information loss which may occur the process of image transformation, we suggest a new CNN-based forecasting model that can be applied on some time series and, can successfully extract the features of them in the forecasting process. The proposed CNN forecasting model is composed of three convolutional layers and five full connected layers, also to be able to determine the nonlinear relation between input and output *Relu* and *Elu* activation functions have also been used. The suggested framework has been applied to some of the most evaluated financial time series, which are Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX) and Financial Time Stock Exchange for London stock market data (FTSE). The results have been evaluated on different aspects as an error criterion, a regression analyses and also a visual demonstration. It has been clearly observed that CNN structure has produced outstanding forecasts compared to some other state-of-the-art forecasting tools such as different kinds of ANN, LSTM, fuzzy-based approaches, and some traditional methods.

Keywords Convolutional neural network · Financial time series · Forecasting · Deep learning

✉ Ozge Cagcag Yolcu
ozge.cagcag@marmara.edu.tr

Melih Kirisci
melih.kirisci@dokap.gov.tr

¹ Ministry of Industry and Technology, Regional Development Administrations, Giresun, Turkey

² Department of Statistics, Faculty of Arts and Sciences, Marmara University, Istanbul, Turkey

1 Introduction

Time series forecasting methods have been widely used in many areas such as financial market prediction, stock exchange, weather and environmental state prediction, energy consumption etc. When the literature of time series forecasting is examined, it is seen that financial time series forecasting has a broad usage and significant impact of the future plans and behaviour of the markets. Although there are so many studies that used different kinds of forecasting tools in these areas, getting accurate and unbiased forecasting of time series from those methods cannot be achieved all the time. In these cases, usage necessity of the advanced forecasting model is inevitable.

In literature, there are different adapted methodologies for time series analysis and forecasting. Autoregressive moving average (ARMA), autoregressive integrated moving average (ARIMA), curve fitting, regression analysis, Bayesian analysis, Kalman Filter methods are the most known and used methods for the time series analyses process.

Besides these traditional approaches, in particular recent years, machine learning techniques have proved to be useful for making accurate forecasts. There are so many studies used different kinds of forecasting tools in these areas. Krollner et al. have evaluated the stock market forecasting data sets from different machine learning based models [1]. Cavalcante et al. have investigated so many forecasting model approaches such as ANN, SVM, hybrid mechanisms, optimization and ensemble methods [2]. Forecasting of stock market index has been investigated by Wang et al. [3] and Liao and Wang [4]. Kara et al. have predicted directional prediction movement of Istanbul Stock Exchange (ISE) National 100 Index through ANN and SVM [5]. Chen et al. have evaluated the TAIEX time series by taking advantage of a new ANN model to be able to obtain forecasting results [6]. Also difference between the prediction performance of multilayer perceptron (MLP) and dynamic ANN for NASDAQ Stock Index data has been investigated by Guresen et al. [7]. Kim and Han [8], Qiu and Song [9] and Qiu Song and Akagi [10] have preferred to use GA and SA to find initial weights of an ANN, and created hybrid method outperformed the standard ANN-based methods for the Nikkei 225 index return prediction. By proposing a MLP model, Vanstone et al. have managed to put create a system that can give an idea about the buying or selling points of the Australian market [11]. On the other hand there are other types of forecasting tools, used for financial time series forecasting, called fuzzy-based methods that compete with machine learning models [12–22]

Moreover, a deep MLP model trained via genetic algorithm has been put forward by Sezer et al. to be able to forecast Dow Jones stock index [23]. As well as GA, particle swarm optimization (PSO) also has a significant impact on the performance of forecasting tools introduced for financial time series forecasting. Pulido et al. have proposed a novel MLP trained by PSO for the prediction of Mexican Stock Exchange [24]. Moreover in order to predict the movements of stocks in the market, Ballings et al. have examined the difference between the performances of ensemble solutions (Random Forest, AdaBoost and Kernel Factory) and classifier models (ANN, Logistic Regression, SVM and KNN) [25]. In particular, recently with regard to the necessity of the analysis of a large amount of data, researchers have converged to use deep neural networks (DNNs) such as CNN, LSTM and other approaches. While the most of the CNN models are famous with image/video processing, classification, test detection and recognition and many more, Recurrent Neural network (RNN) which has internal memory in order to extract important historical features are the most attractive for time series. In the literature Krizhevsky et al. have taken the advantage of CNN for the classification of ImageNet [26], Karpathy et al. have used it for

the large-scale video sentence classification and sentence modelling [27], Kim et al. [28] and Kalchbrenner et al. [29] have put forward studies respectively on this issue. Krauss et al. have intended to compare the deep neural networks, gradient-boosted-trees, random forest [30]. Also Ding et al. have introduced deep learning method for the prediction of event driven stock market to be able to extract news texts, information from internet and newspapers [31]. Tino et al. have put forward comparative study in which using the Markov model and RNN in financial volatility trading for the German DAX and British FTSE indexes [32]. Deep direct reinforcement method and fuzzy deep direct reinforcement method with Recurrent DNN have been used in order to make some inference on the Chinese futures market and commodity futures market by Deng et al. [33]. Examining the literature in which use deep learning models for financial analysis, it seems quite limited except for several reports on the forecasting of stock price fluctuations [34–38]. On the other hand, Yeh et al. have proposed a method depend on deep learning method for forecasting stock price fluctuations [39]. As far as it is known, CNNs have been utilized in a few studies for the prediction of stock market. Di Persio and Honchar have used CNN, LSTM, and MLP to the historical data of close prices of the SandP 500 index [37]. Arévalo et al. have proposed a deep ANN consist of five hidden layers for the forecasting of Apple Inc.'s stock price [40]. Gunduz et al. have preferred to use CNN to be able to get prediction results of Istanbul exchange [41]. Nelson et al. have taken the advantage of LSTM in the prediction of the Brazilian stock prices direction [42].

When the studies in the literature are examined, it is obviously seen that applications of CNNs to financial analyses have only been reported in a small number of studies. The reason behind this issue can be explained with the structure of CNNs. CNNs are more applicable for the detection of images and less applicable for the general numerical data including financial statements. In the limited CNN studies, which are used for the financial forecasts in the literature, the processes are performed by converting crisp data into images. However, it should be noted that this extra image transformation will cause a loss of information and the interruption of the time sequence property of the time series and thus deterioration of the predictive performance. Creating a proper CNN model with different structures can make CNN methods more applicable for the forecasting of time series and enhances the forecasting performance of the financial time series. In this sense, the basic motivation of this study is to create a CNN model that is not caused such information loss, the interruption, and the deterioration and to put forward a suitable working principle. From this point on, in this study, a new CNN structure without these kinds of issues and defects, and suitable for time series analysis is proposed to be able to obtain satisfactory forecasting results for financial time series.

Moreover, to prove the outstanding forecasting performance of the proposed CNN structure for financial time series analysis problem, TAIEX data sets which have been widely used in the related literature are investigated in detail. Also, to be able to support the outcomes of the proposed model, FTSE data sets are analysed. The obtained results have been comparatively evaluated with some of the art- mechanism including fuzzy-based models, traditional and deep neural network models. Moreover, detailed regression analyses and its outputs, have been also used to demonstrate the performance of the proposed model. And also, to support all of the findings the scatter plots have been presented as a visual exhibit of the performance.

The rest of the paper is fictionalised as follows: In the second section, a brief of CNN models is given and in the third section the methodology of the research is presented. The details and results of implementations are given in section four with comparatively comments and evaluations. Finally, the obtained findings and conclusions are discussed in the last section.

2 Convolutional Neural Networks

Deep learning as a machine learning approach allows us to train an artificial intelligence mechanism to forecast the outputs by given data-set. For this purpose, both supervised and unsupervised learning can be used. On the other hand, in its simplest definition, a deep neural network is an ANN that is made up of more than three layers–i.e., an input layer, an output layer and multiple hidden layers–is called a ‘deep neural network. The convolutional neural network is one of the most used of these in recent years and it is a class of deep, feed-forward neural networks that have been successfully applied to many application domains. While the traditional neural networks illustrated in Fig. 1 are created by mimicking nerve cells in the human brain CNN, given by Fig. 2, is inspired by the visual cortex of the human brain.

To better understanding of the working principle of CNN method, it would be better to mention the difference between traditional ANN methods and CNN. The performance of

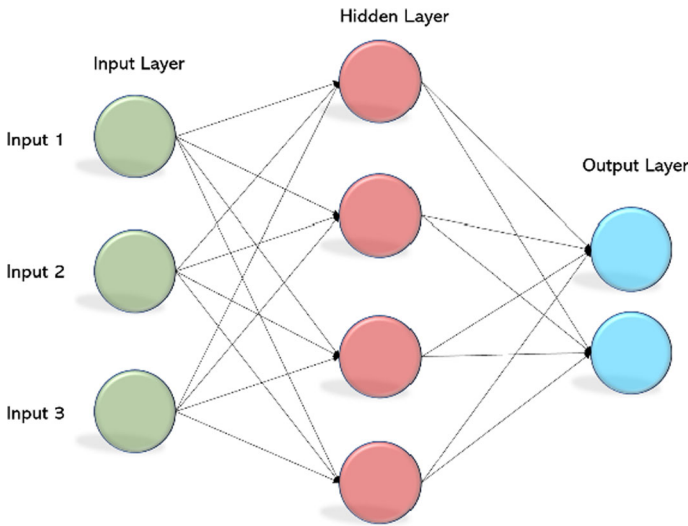


Fig. 1 The traditional ANN

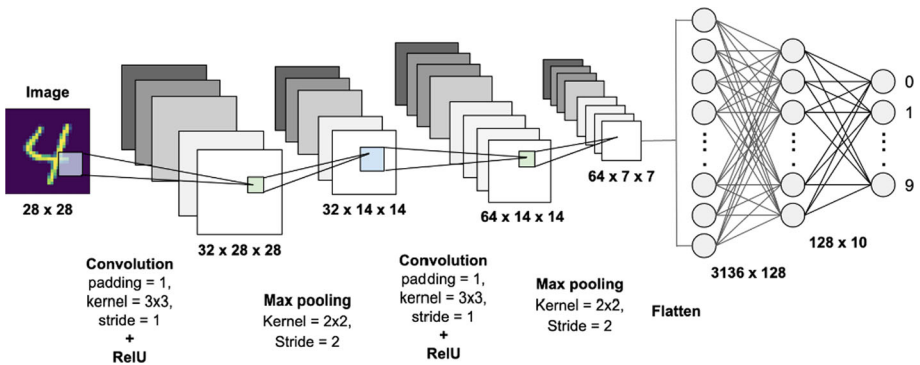


Fig. 2 The architecture of a CNN

traditional ANNs generally gives better results when the data size is not high. But the more data size has high dimensionality the more ANN has worse performance.

From this point of view, in order to get better results, researchers can use more layers in their ANN model. In this case, the parameters of the system will inevitably increase. An increasing number of parameters, as well as full connectivity, would quickly cause overfitting. Concisely when the data size is high preferring CNNs are much better than traditional ones.

CNNs are basically encouraged by biological processes which are based on the animal visual cortex. CNNs are great for capturing local/micro information as well as reducing the complexity of the model which give the opportunity for faster training, needs fewer samples, reduces the chance of overfitting. In literature, CNNs are used for computer vision, image recognition, object detection for self-driving cars, face recognition in social media, image analysis in health services and also especially in recent years in time series forecasting. While CNN is mostly used for image classification/analysis problems [43–45], it is generally not preferred for time series data analytics directly.

The first successful convolutional neural network is introduced by LeCun et al. [46]. CNN architectures consist of some layers as convolutional, max-pooling, dropout, and fully connected also output layer. Convolution is the first layer in which pixels are convolved with a filter or kernel to extract features from an input. And the output of each filter is passed through an activation function which serves to eliminate linearity in attribute matrices before entering the next layer. In the subsampling layer which is the second important transformation, most specifically the number of parameters is reduced. Also, this layer generally consists of pooling layer, which is responsible to decrease the data dimensionality and helpful for the overfitting problem. The pooling layer reduces the amount of information of the convolution layer created for each property and holds the most important information as well. There are many pooling types as max-pooling, min-pooling and average-pooling, among them, the most common type is max-pooling which the maximum value in a certain window is chosen. In a CNN structure, not only the convolution layer but also the pooling layer can repeat multiple times. The last part of the subsampling layer is connected to a fully connected layer which has similar working principles of the MLP. This layer is responsible for taking the output of the previous layers, flattening them and turning them into a single vector that can be an input for the next stage. Fully connected layer applies weights over the input generated by the feature analysis to predict an accurate label. Fully connected output layer gives the final probabilities for each label.

3 The Proposed CNN Forecasting Structure

Forecasting methods have become more attractive every day, especially for forecasting financial time series since, for instance, accurately forecasting the future performance of a business is crucial for the investors to get profit from trading securities. There are many forecasting tools in the time series analysis. While some of them are still based on traditional forecasting models such as ARIMA, Time Series Regression and VAR Models, fuzzy and computational based forecasting tools have been widely preferred due to their better forecasting performance as well. Although especially in recent years, deep neural networks like LSTM have been widely used for time series forecasting, it should be noted that the RNN-based these models that have been quite famous in time series analyses still have some problems need to be worked on; such as time-consuming iterations, complex gate mechanisms, and slow

response to dynamic changes Moreover, while the LSTM helps to solve some problems, it only saves the long-term sequence information and not that much good at about getting global modelling.

On the contrary of the RNNs such as LSTM, CNNs have the strength of fast training [47–49] simple structures, and no dependency constraints on previous steps. It has the ability to focus on some key features of the series from a global perspective.

Thanks to these advantages, although they are used in time series prediction problems, CNNs need to convert crisp time-series data into an image using some transformation process. It is clear that this extra image transformation will cause a loss of information and the interruption of the time sequence property of the time series and thus deterioration of the predictive performance. So, from this point of view in this study, our basic aim is to propose a new CNN-based forecasting tool for various financial time series, which is capable of not only extracting the significant features by taking the advantage of its own specific learning process but also providing outstanding forecasting results.

Compared to some current forecasting tools, the basic contributions of the proposed CNN structure can be given as;

- There is no overfitting problem unlike classical ANN models.
- It is better than LSTM in global modelling.
- Since extra image transformation is not required in forecasting process, loss of information is prevented, so superior forecasts can be obtained.
- As the pooling process is not used, again the loss of information is prevented and also time sequence effect of time series is preserved on forecasts.

The proposed CNN forecasting structure includes three convolutional layers and five fully connected layers also in the convolutional layers, Relu and Elu activation functions have been preferred to get nonlinear relations.

The input layer of created CNN is 1D and the size of input layer depends on lagged variables of time series. The created model can be given with the Fig. 3.

The inputs of the CNN given in Fig. 2 is a matrix which consists of vectors formed by lagged variables of the time series. The proposed CNN structure does not require extra image transformation of the crisp data of the time series and processes them directly as an image. In the case of the number of lagged variables (model order) are p and the size of learning sample is n_{trn} , the size of the matrix would be $(n_{trn} - p) \times p$.

In the analyses of the proposed CNN process consists of the input layer $((n_{trn} - p) \times p)$, three convolutional layers (8, 16, 32 with 2×2 kernel), one dropout (0.20), fully connected

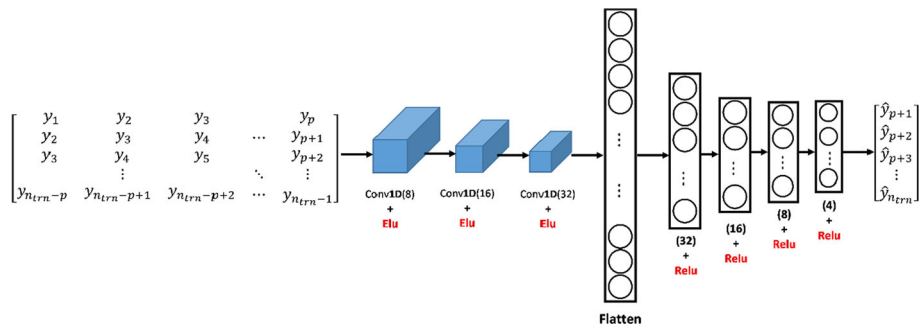


Fig. 3 The structure of the proposed CNN

layers (32, 16, 8, 4), and an output layer. Also, to be able to prevent overfitting, dropout layers are utilized. In other words, a certain percentage of neurons are ignored for full connection layers after flatten operation. The neurons are selected randomly to be turned off. therefore, by being “turned off”, they do not have any role in education in the forward run and backpropagation and hence, the model uses the other neurons to carry on with the learning. Dropout provides using a smaller neural network and so it has a regularizing effect. The regulation means making the network less sensitive to the individual weights of the neurons, allowing other neurons to take on the tasks of the dropped neurons to learn more. The dropout used in this study is realized with a 20% probability of being ignored for each neuron. 2×2 filter size the smallest and most commonly used kernel size is preferred in our model as filter. In literature, although there is different size of CNN filters like 2×2 , 3×3 , 5×5 and 7×7 , usage of small size filters generally provides better results due to ability of catching more details of the relationships that the time series contains. In the process, pooling layer is eliminated in order to keep as much as necessary information corresponding to the forecasted time point.

In this study, in the analysis process of new CNN forecasting structure Keras, Tensorflow infrastructure are used and each test run lasts for 1000 epochs. Algorithm 1 represents the basic working principles of proposed CNN forecasting structure.

Generalized Proposed Algorithm

Procedure AllPhases().

Data-Set train/test Phase:

```
dataset = read(d)
x_train = d[0 : (length(d) - p)]
x_test = d[(length(d) - p + 1) : length(d)]
```

Input Creation Phase for CNN:

A single variable time series model must be prepared before training. As such, the sequence of observations must be transformed into multiple examples from which the model can learn.

Consider a given univariate time series

$$x_{train} = [5, 15, 25, 35, 45, 55, 65, 75, 85]$$

Let’s divide the time series into multiple instances where each instance has a certain number of time steps and the output is a single time step.

For time step $(p) = 4$

$$X = \begin{bmatrix} 5 & 15 & 25 & 35 \\ 15 & 25 & 35 & 45 \\ 25 & 35 & 45 & 55 \\ 35 & 45 & 55 & 65 \\ 45 & 55 & 65 & 75 \end{bmatrix} \quad y = \begin{bmatrix} 45 \\ 55 \\ 65 \\ 75 \\ 85 \end{bmatrix}$$

The univariate time series divides into five samples, in which each sample has four input time steps and one output time step.

CNN Phase:

With the proposed CNN structure, the time series training process cycle is started. The filters used in the CNN are actually randomly generated at the beginning (*wt*) and they correspond to weights of the classical ANN. In the training process, the number of iterations is taken as 1000 epochs.

Table 1 The properties of data organization

Series no.	Time series	Year	Training	Test
1	TAIEX	2000	224	47
2		2001	201	43
3		2002	205	43
4		2003	206	43
5		2004	205	45
6	FTSE	2010	252	10
7		2011	251	10
8		2012	252	10
9		2013	253	10
10		2014	253	10

for($i = 0; i < epoch; epoch ++$)

model = *CNN*(i)

model.train()

model = *CNN*(i)

Evolution Phase:

In the CNN model whose training has been completed, the best model is obtained by calculating the test data and real values statistical error metric (RMSE) for each different starting time.

d : time series in vector format

ts : data size to be tested

wt : First weighting time of convolution filters

4 Experimental Results and Discussion

4.1 Data Preparation

The forecasting performance of the proposed CNN structure has been evaluated with the analysis of two different financial time series; TAIEX (available from https://www.twse.com.tw/en/page/trading/indices/MI_5MINS_HIST.html) and FTSE (available from <https://www.wsj.com/market-data/quotes/index/UK/UKX/historical-prices>), and the forecasting results-based findings have been presented as evidence of the superior performance of the proposed model. In many studies proposed in forecasting problems, a convenient ratio to separate in-samples from out-of-samples ranging from 70:30% to 90:10%. From this point of view, the observations of the period of the last two months are taken as the out-of-samples, for TAIEX data observed in the years of 2000, 2001, 2002, 2003, and 2004. Moreover, for FTSE data observed in the years of 2010, 2011, 2012, 2013, and 2014, the observations of the period of the last two weeks are taken as the out-of-samples. The information including the organization of the data for both time series is summarized in Table 1.

4.2 Performance Measure

In the literature, various evaluation metrics have been used to compare performance of the models. In this study previously, Root Mean Square Error (RMSE) given in Eq. (1) is utilized to evaluate the outcome of the models.

$$RMSE = \sqrt{\frac{1}{n_{trn}} \sum_{t=1}^{n_{trn}} (Target_t - Output_t)^2} \quad (1)$$

The second evaluation one is the analysis of the regression model to be created for the predictions and target values. This model can be represented as in Eq. (2). As can be seen, this model is a model that does not contain constant terms. For a superior estimation tool, the regression and specification coefficients of this model are expected to be quite close to 1.

$$Y_t = \beta \hat{Y}_t + \varepsilon_t \quad (2)$$

Moreover, the scatter diagram which illustrates the harmonisation between the predictions and real values can also be utilized.

4.3 Alternative Comparison Models

In the study, to be able to compare and make comment on the forecasting ability of the proposed CNN structure, various methods which are known their outstanding forecasting performance are chosen. The forecasting tools used in comparison for TAIEX can be given below;

SC93	Song and Chissom's Method - Fuzzy Based [13]
C96	Chen's Method - Fuzzy Based [14]
H01	Huarng's Method - Fuzzy Based [15]
HY06	Huarng and Yu's Method - Fuzzy Based [16]
E10	Egrioglu et al.'s Method - Fuzzy Based [18]
CC15	Chen and Chen's Method - Fuzzy Based [20]
B15	Bas et al.'s Method - Fuzzy Based [19]
S18	Sarica et al.'s Method - Fuzzy Based [22]
L-NL	Linear and Non-Linear ANN - ANN Based [50]
MLP	Multi-Layer Perceptron ANN - ANN Based [51]
SP	Sigma-Pi Neuron Model ANN - ANN Based [52]
SMN	Single Multiplicative Neuron Model ANN - ANN Based [52]
ANFIS-G	Adaptive Neuro-Fuzzy Inference System - Grid Partition - Fuzzy Based [12]
LSTM	Long-Short Term Memory [34] - Deep ANN Based [53]

Moreover, the prediction tools used in comparison for FTSE can be given below;

ANFIS-G	Adaptive Neuro-Fuzzy Inference System - Grid Partition - Fuzzy Based [12]
ANFIS-S	Adaptive Neuro-Fuzzy Inference System - Subtractive Clustering - Fuzzy Based [12]
MANFIS	Modified Adaptive Neuro-Fuzzy Inference System - Fuzzy Based [21]
T1FRF	T1 Fuzzy Regression Function - Fuzzy Based [17]
MLP	Multi-Layer Perceptron ANN - ANN Based [17]
RBF	Radial Basis Function Neural Network - ANN Based [54]
SP	Sigma-Pi Neuron Model ANN - ANN Based [52]

- SMNM Single Multiplicative Neuron Model ANN - ANN Based [55]
- L-NL Linear and Non-Linear ANN - ANN Based [50]

4.4 Evaluation of the Methods' Performance

4.4.1 TAIEX Experiment

Firstly, daily TAIEX data belongs to five years are analysed through proposed CNN forecasting model and obtained results are evaluated with various state-of-the-art models' results, together. In the analysis process, while the observations of time series in the first ten months are taken as the training set, the last two months are taken as the test set for each of year. The inputs of the proposed CNN forecasting model are composed of the lagged crisp data matrix. Also, the model order (p) has been changed from 3 to 7. So, let # observations of the training set is n_{trn} , in this case, the dimension of the input matrix of proposed CNN model will change from $(n_{trn} - 3) \times 3$ to $(n_{trn} - 7) \times 7$. The number of elements of the output vector will be $n_{trn} - p$. In the comparison process, the architecture that produces the best forecasting performance of all the architectures is used.

At the end of the analysis of the TAIEX data sets, the best forecasting results of the proposed CNN forecasting model, in terms of RMSE criterion, are presented with the results of some well-known models in Tables 2.

Considering Table 2, with respect to the RMSE values obtained for out-of-sample, it is obviously seen that forecasting performance of the proposed CNN forecasting model is

Table 2 The forecasting performances of the models for the TAIEX

Forecasting models	RMSE					Average
	TAIEX					
	2000	2001	2002	2003	2004	
SC93	293	116	76	77	82	129
C96	225	116	76	77	82	115
H01	152	130	84	56	116	108
HY06	133	124	82	62	85	97
E10	255	130	84	56	116	128
CC15	125	115	65	53	53	82
B16	140	120	77	60	59	91
S18	123	111	66	52	54	81
L-NL	342	113	66	51	53	125
MLP	304	130	80	68	64	129
SP	318	119	80	85	63	133
SMN	588	128	82	61	65	185
ANFIS-G	137	115	66	57	61	87
LSTM	136	101	89	92	70	108
The proposed CNN	105	110	60	51	50	75

Underlined and bold values represent the best forecasting performances

better than other state-of-the-art models for all years except the 2004. Even for the 2004-year, proposed CNN forecasting method has the second-best prediction performance and its results are quite close to the best one. The difference between these two models is less than 10%. So, it means that from this aspects, proposed CNN forecasting model has the competitive prediction ability which is as expected. The average of RMSE criterion can be seen as further proof of this outcome i.e., that the proposed CNN forecasting model produces superior performance.

Besides the different ANN, fuzzy-based, and FTS model, when the obtained results are in particular examined for LSTM which is a very important model of deep learning-based, it is seen that the proposed CNN forecasting model could generate on average about 30% better forecasts than LSTM.

Another evaluation approach is the analysis of the regression model to be created for the predictions and target values. To be able to demonstrate the outstanding forecasting performance of the proposed CNN forecasting model, some characteristics of a linear regression model established as $Y_t = \beta \hat{Y}_t + \varepsilon_t$ can be taken advantage of. From this aspect, the regression coefficient (β) and also the coefficient of determination (R^2) of the model are evaluated. If the forecasting model produces satisfactory results, then both $\hat{\beta}$ and R^2 values are expected to be equal 1, or quite close to 1 as possible as. Table 3 summarises all findings obtained from the regression model ($Y_t = \beta \hat{Y}_t + \varepsilon_t$) which takes into account the observed values as the dependent variable and forecasted values as the independent variable. When the findings given by Table 2 are examined, it is obviously seen that, for all cases, both $\hat{\beta}$ and R^2 values are quite close to 1. Also considering both standard errors of the regression coefficients, which are very close to zero, and the %95 confidence intervals, which are quite narrow and cover 1, it can be said that proposed CNN forecasting model can be used as a powerful and preferable forecasting tool.

Furthermore, to be able to evaluate the proposed CNN forecasting model from the different aspects, we obtained the scatter diagrams which are the mathematical diagrams using Cartesian coordinates showing the observed and forecasted values for the out of sample. The satisfactory performance of the proposed CNN structure can be also observed from the scatter plots given in Fig. 4. So, to talk about a successful prediction tool, most of the points on these scatter plots are expected to be close to the line segment. When the Fig. 4 are investigated, it is clear that the scattering the points satisfy the expectations for an inference tool having an outstanding prediction performance. These distribution graphs also show that they all have a

Table 3 Characteristics of a linear regression model $Y_t = \beta \hat{Y}_t + \varepsilon_t$

Year	Model estimation	Sig. of <i>F</i>	St. error of $\hat{\beta}$	Sig. of β	%95 Confidence Interval of β		R^2
					Lower bound	Upper bound	
2000	$Y_t = 0.999627\hat{Y}_t$	2.74E-66	3.22E-03	8.93E-68	0.993109	1.006145	0.999797
2001	$Y_t = 1.000029\hat{Y}_t$	3.08E-64	3.65E-03	1.14E-65	0.992646	1.007413	0.999481
2002	$Y_t = 0.999927\hat{Y}_t$	5.35E-66	2.19E-03	1.12E-67	0.995486	1.004369	0.999916
2003	$Y_t = 0.999039\hat{Y}_t$	3.32E-78	1.40E-03	4.63E-80	0.996210	1.001869	0.999926
2004	$Y_t = 1.000370\hat{Y}_t$	1.42E-80	1.36E-03	1.95E-82	0.997626	1.003115	0.999928

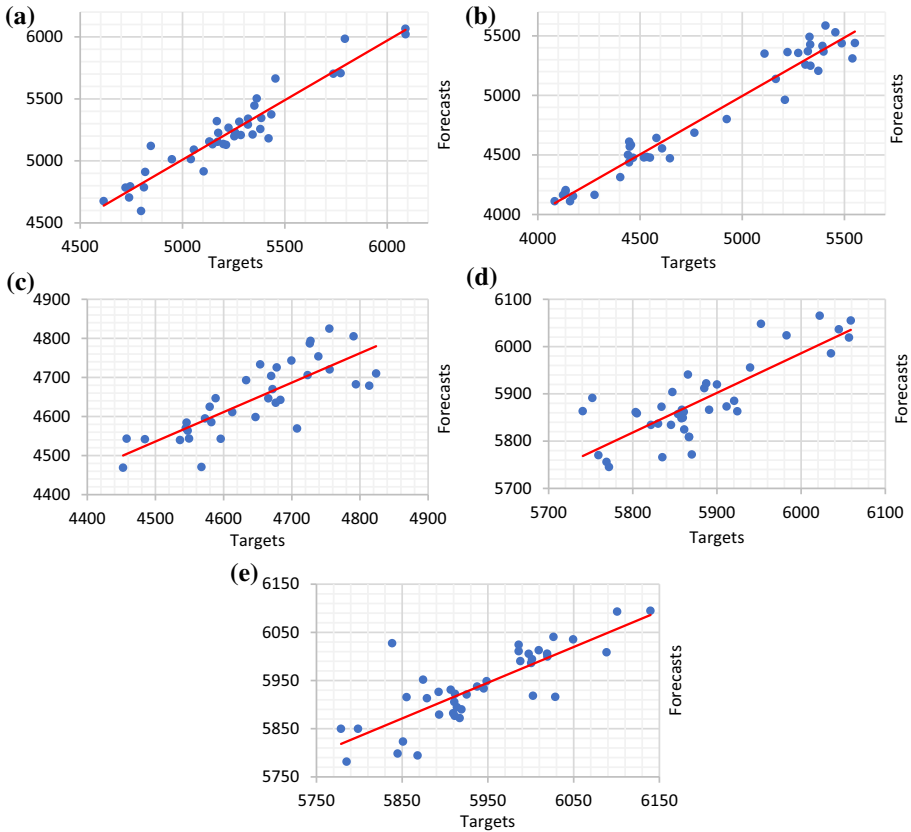


Fig. 4 **a** The Scatter Plot for the TAIEX2000 **b** The Scatter Plot for the TAIEX2001 **c** The Scatter Plot for the TAIEX2002 **d** The Scatter Plot for the TAIEX2003

strong positive relationship. This means that the predicted values are very close to the actual test values.

4.4.2 FTSE Experiment

Secondly, five different daily FTSE data belong to the years of 2010, 2011, 2012, 2013 and 2014 are analysed via the proposed CNN forecasting model and obtained results are evaluated with various state-of-the-art models' results, together. In the analysis process, the observations of the last ten days are taken as the test set for each year. So, it is aimed to obtain the two weeks forecast. Just like the TAIEX experiment, in the FTSE experiment, the inputs of the forecasting CNN model are composed of the lagged crisp data matrix. Also, the model order (p) has been changed from 3 to 7. Forecasting error for the optimal results obtained from the proposed method as well as forecasting error of other alternative methods including both fuzzy-based and ANN-based models are presented in Table 4.

In the light of Tables 4, we can obviously notice that the proposed CNN forecasting model is the model that has the minimum RMSE values compared to the other forecasting models. So, it can be said that proposed CNN forecasting model has the best forecasting performance

Table 4 The forecasting performances of the models for the FTSE

Models	RMSE						
	FTSE					Average	Median
	2010	2011	2012	2013	2014		
ANFIS-G (<i>Fuzzy-Based</i>)	38.49	50.09	21.91	49.34	82.75	48.52	49.34
ANFIS-S (<i>Fuzzy-Based</i>)	40.81	40.63	19.24	47.45	68.66	43.36	40.81
MANFIS (<i>Fuzzy-Based</i>)	31.36	37.77	18.50	45.60	63.36	39.32	37.77
T1FF (<i>Fuzzy-Based</i>)	31.35	42.64	18.90	49.88	62.46	41.05	42.64
MLP(<i>ANN-Based</i>)	32.43	46.46	18.69	48.68	76.59	44.57	46.46
RBF(<i>ANN-Based</i>)	31.67	38.99	18.51	47.16	62.68	39.80	38.99
SP(<i>ANN-Based</i>)	32.11	44.89	18.12	46.94	76.32	43.68	44.89
SMN(<i>ANN-Based</i>)	32.60	44.78	17.88	49.29	76.61	44.23	44.78
L-NL(<i>ANN-Based</i>)	31.88	44.33	19.11	48.70	75.95	43.99	44.33
The Proposed CNN	30.88	30.82	18.69	30.18	52.55	32.62	30.82

Underlined and bold values represent the best forecasting performances

for whole data sets except the year of 2012. Moreover, when the average and median statistics of RMSE relevant to all methods are also considered, it is obviously noticed that the proposed CNN forecasting model produces superior forecasting performance.

On the other hand, with Fig. 5, we sum up the rate of improvement in forecasting accuracy according to fuzzy-based models not only for each year but also for the average and median statistics. When Fig. 5 is examined, in particular, for the year of 2013, the proposed model has a remarkable improvement rate with 33.82%, compared to the best fuzzy-based model. Furthermore, considering the mean and median statistics of all years, 17.03 and 18.41% improvement rates are achieved, respectively.

Similarly, from Fig. 6, compared to the best ANN-based model, it is seen that the improvement rates are changed between 2.48 and 35.70%. Moreover, for the average and the median statistics, it can be said that the proposed CNN forecasting model generate considerable improvement rate with 18% and 21%, respectively. In the light of this information, it is demonstrated that the proposed CNN forecasting model displays remarkable forecasting

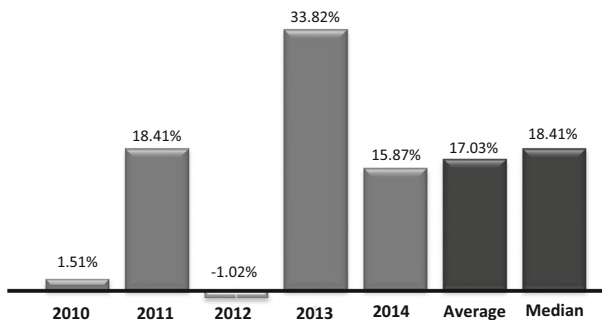


Fig. 5 The improvement rates compared to the best Fuzzy-based model

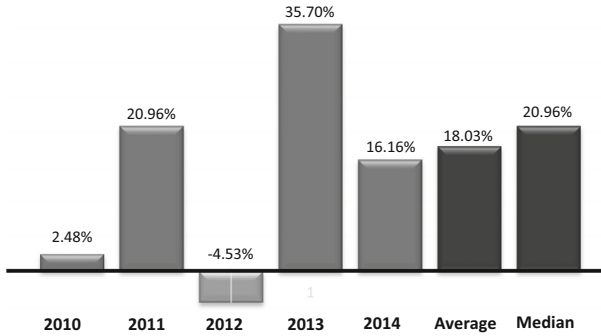


Fig. 6 The improvement rates compared to the best ANN-based model

performance and can be used as an effective and competitive forecasting tool for the financial time series to better accuracy.

4.4.3 Evaluation of the Computational Load

In the general sense, deep learning methods are known as time-consuming methods. In this respect, in the analysis of TAIEX and FTSE data sets, the computational time of the proposed CNN has been evaluated from a simple perspective. For each model order (p) which are changed from 3 to 7, and the average of them, the computational time of the proposed CNN has been visualized in Figs. 7 and 8, for TAIEX and FTSE data sets, respectively. From Figs. 7 and 8, it is seen that the average of the computational times is about just 8 and 9 s. This duration is seen as really reasonable for a time series forecasting problem.

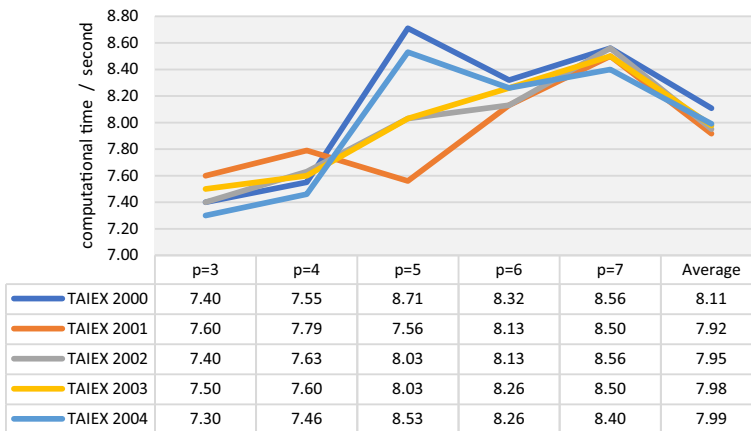


Fig. 7 The computational time of the proposed CNN for TAIEX data sets

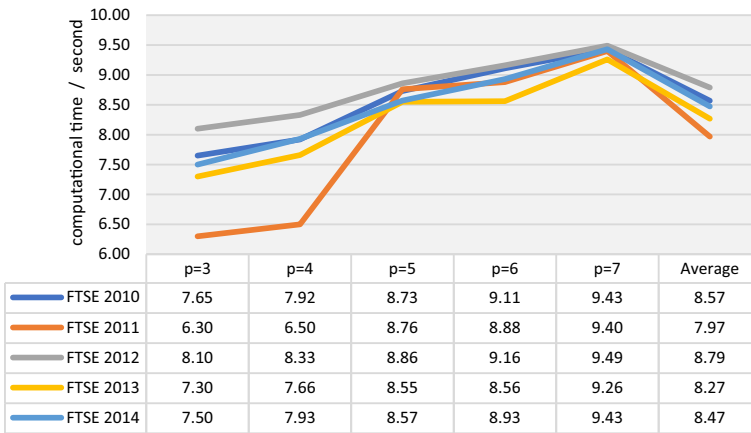


Fig. 8 The computational time of the proposed CNN for FTSE data sets

5 Conclusions

Financial time series forecasting has been quite popular among machine learning researchers for many years. However, especially in recent years, deep neural networks have been used effectively for time series prediction. Although LSTM has been widely preferred for time series analyses, different CNN models with different structures have been used as a prediction tool to improve forecasting performance, as well. Here, the basic and crucial trouble is that the information loss and the interruption of the time sequence caused by extra image transformation and pooling transaction. In this study, we have aimed to enhance the performance of CNN, which has an outstanding forecasting ability, by utilizing a special structure for financial time series analysis. The proposed CNN forecasting structure is designed to be able to get superior forecasting results for the financial time series by effectively extracting the relevant information from the data set. The created CNN forecasting structure does not need extra image transformation for its inputs and also pooling transaction. The results obtained at the end of ten different implementations including the forecasting process of TAIEX and FTSE data sets demonstrate that the proposed CNN forecasting structure can be used as an effective and competitive prediction tool compared to some fuzzy, ANN, and deep ANN-based models. It is also shown that the proposed model is able to improve the performance of forecasting compared to the baseline models by about in average 18% in terms of RMSE. On the other hand, one of the important contributions of the proposed model to the financial time series literature is that, again, proposed CNN forecasting model has achieved 30% better accuracy than LSTM on average. Moreover, both scatter plots and regression analysis outputs demonstrate visually and analytically that the obtained predictions and observations are highly compatible, as an indicator of the superior prediction performance of the proposed model. Consequently, the proposed CNN forecasting model has put forward an extraordinary forecasting performance and can be taken advantage of as a reliable and competitive forecasting tool for the financial time series.

For future work, we are planning to create a multivariate forecasting CNN model in order to investigate the forecasting performance and combine it with other neural network frameworks. Also, it is clear that some other potential studies can be followed by applying

a fuzzy-based set up by getting the advantage of fuzzy time series methods on CNN models to be able to come up with more accurate forecasting model for time series.

References

1. Krollner B, Vanstone B, Finnie G (2010) Financial time series forecasting with machine learning techniques: a survey. In: Proceedings of the 18th European symposium on artificial neural networks - computational intelligence and machine learning, ESANN 2010
2. Cavalcante RC, Brasileiro RC, Souza VLF et al (2016) Computational intelligence and financial markets: a survey and future directions. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2016.02.006>
3. Wang NY, Chen SM (2009) Temperature prediction and TAIFEX forecasting based on automatic clustering techniques and two-factors high-order fuzzy time series. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2007.12.013>
4. Liao Z, Wang J (2010) Forecasting model of global stock index by stochastic time effective neural network. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2009.05.086>
5. Kara Y, Acar Boyacioglu M, Baykan ÖK (2011) Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the Istanbul Stock Exchange. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2010.10.027>
6. Chen AS, Leung MT, Dauok H (2003) Application of neural networks to an emerging financial market: forecasting and trading the Taiwan Stock Index. *Comput Oper Res*. [https://doi.org/10.1016/S0305-0548\(02\)00037-0](https://doi.org/10.1016/S0305-0548(02)00037-0)
7. Guresen E, Kayakutlu G, Daim TU (2011) Using artificial neural network models in stock market index prediction. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2011.02.068>
8. Kim KJ, Han I (2000) Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Syst Appl*. [https://doi.org/10.1016/S0957-4174\(00\)00027-0](https://doi.org/10.1016/S0957-4174(00)00027-0)
9. Qiu M, Song Y (2016) Predicting the direction of stock market index movement using an optimized artificial neural network model. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0155133>
10. Qiu M, Song Y, Akagi F (2016) Application of artificial neural network for the prediction of stock market returns: the case of the Japanese stock market. *Chaos, Solitons Fractals*. <https://doi.org/10.1016/j.chaos.2016.01.004>
11. Vanstone B, Finnie G, Hahn T (2012) Creating trading systems with fundamental variables and neural networks: the Aby case study. *Math Comput Simul*. <https://doi.org/10.1016/j.matcom.2011.01.002>
12. Jang JSR (1993) ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Trans Syst Man Cybern*. <https://doi.org/10.1109/21.256541>
13. Song Q, Chissom BS (1993) Fuzzy time series and its models. *Fuzzy Sets Syst* 54:269–277. [https://doi.org/10.1016/0165-0114\(93\)90372-O](https://doi.org/10.1016/0165-0114(93)90372-O)
14. Chen SM (1996) Forecasting enrollments based on fuzzy time series. *Fuzzy Sets Syst*. [https://doi.org/10.1016/0165-0114\(95\)00220-0](https://doi.org/10.1016/0165-0114(95)00220-0)
15. Huarng K (2001) Effective lengths of intervals to improve forecasting in fuzzy time series. *Fuzzy Sets Syst*. [https://doi.org/10.1016/S0165-0114\(00\)00057-9](https://doi.org/10.1016/S0165-0114(00)00057-9)
16. Huarng K, Yu THK (2006) Ratio-based lengths of intervals to improve fuzzy time series forecasting. *IEEE Trans Syst Man, Cybern Part B Cybern*. <https://doi.org/10.1109/TSMCB.2005.857093>
17. Türkşen IB (2008) Fuzzy functions with LSE. *Appl Soft Comput J*. <https://doi.org/10.1016/j.asoc.2007.12.004>
18. Egrioglu E, Aladag CH, Yolcu U et al (2010) Finding an optimal interval length in high order fuzzy time series. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2009.12.006>
19. Bas E, Egrioglu E, Aladag CH, Yolcu U (2015) Fuzzy-time-series network used to forecast linear and nonlinear time series. *Appl Intell*. <https://doi.org/10.1007/s10489-015-0647-0>
20. Chen SM, Chen SW (2015) Fuzzy forecasting based on two-factors second-order fuzzy-trend logical relationship groups and the probabilities of trends of fuzzy logical relationships. *IEEE Trans Cybern*. <https://doi.org/10.1109/TCYB.2014.2326888>
21. Egrioglu E, Aladag CH, Yolcu U, Bas E (2015) A new adaptive network based fuzzy inference system for time series forecasting. *Aloy J Soft Comput Appl* 2:25–32
22. Sarıca B, Eğrioğlu E, Aşıkçıl B (2018) A new hybrid method for time series forecasting: AR–ANFIS. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-016-2475-5>
23. Sezer OB, Ozbayoglu M, Dogdu E (2017) A deep neural-network based stock trading system based on evolutionary optimized technical analysis parameters. *Procedia Computer Science* 114:473–80

24. Pulido M, Melin P, Castillo O (2014) Particle swarm optimization of ensemble neural networks with fuzzy aggregation for time series prediction of the Mexican stock exchange. *Inf Sci (Ny)*. <https://doi.org/10.1016/j.ins.2014.05.006>
25. Ballings M, Van Den Poel D, Hespeels N, Gryp R (2015) Evaluating multiple classifiers for stock price direction prediction. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2015.05.013>
26. Krizhevsky A, Sutskever I, Hinton GE (2017) ImageNet classification with deep convolutional neural networks. *Commun ACM*. <https://doi.org/10.1145/3065386>
27. Karpathy A, Toderici G, Shetty S, et al (2014) Large-scale video classification with convolutional neural networks. In: *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*
28. Kim Y (2014) Convolutional neural networks for sentence classification. In: *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*
29. Kalchbrenner N, Grefenstette E, Blunsom P (2014) A convolutional neural network for modelling sentences. In: *52nd Annual meeting of the association for computational linguistics, ACL 2014 - proceedings of the conference*
30. Krauss C, Do XA, Huck N (2017) Deep neural networks, gradient-boosted trees, random forests: statistical arbitrage on the SandP 500. *Eur J Oper Res*. <https://doi.org/10.1016/j.ejor.2016.10.031>
31. Ding X, Zhang Y, Liu T, Duan J (2015) Deep learning for event-driven stock prediction. In: *IJCAI international joint conference on artificial intelligence*
32. Tiño P, Schittenkopf C, Dorffner G (2001) Financial volatility trading using recurrent neural networks. *IEEE Trans Neural Networks*. <https://doi.org/10.1109/72.935096>
33. Deng Y, Bao F, Kong Y et al (2017) Deep direct reinforcement learning for financial signal representation and trading. *IEEE Trans Neural Networks Learn Syst*. <https://doi.org/10.1109/TNNLS.2016.2522401>
34. Aggarwal S, Aggarwal S (2017) Deep investment in financial markets using deep learning models. *Int J Comput Appl*. <https://doi.org/10.5120/ijca2017913283>
35. Bao W, Yue J, Rao Y (2017) A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0180944>
36. Chong E, Han C, Park FC (2017) Deep learning networks for stock market analysis and prediction: methodology, data representations, and case studies. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2017.04.030>
37. Di Persio L, Honchar O (2016) Artificial neural networks architectures for stock price prediction: comparisons and applications. *Int J Circuits, Syst Signal Process* 10:403–13
38. Yoshihara A, Fujikawa K, Seki K, Uehara K (2014) Predicting stock market trends by recurrent deep neural networks. *Lecture Notes in computer science (including Subser Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 8862: <https://doi.org/10.1007/978-3-319-13560-1>
39. Yeh SH, Wang CJ, Tsai MF (2015) Deep belief networks for predicting corporate defaults. In: *2015 24th wireless and optical communication conference, WOCC 2015*
40. Arévalo A, Niño J, Hernández G, Sandoval J (2016) High-frequency trading strategy based on deep neural networks. In: *Lecture Notes in computer science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*
41. Gunduz H, Yaslan Y, Cataltepe Z (2017) Intraday prediction of Borsa Istanbul using convolutional neural networks and feature correlations. *Knowledge-Based Syst*. <https://doi.org/10.1016/j.knosys.2017.09.023>
42. Nelson DMQ, Pereira ACM, De Oliveira RA (2017) Stock market's price movement prediction with LSTM neural networks. In: *Proceedings of the international joint conference on neural networks*
43. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*
44. Lin M, Chen Q, Yan S (2014) Network in network. In: *2nd International conference on learning representations, ICLR 2014 - Conference track proceedings*
45. Szegedy C, Liu W, Jia Y, et al (2015) Going deeper with convolutions. In: *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*
46. Lecun Y, Bengio Y (2010) Convolutional networks for images, Speech, and Time Series Variable-Size Convolutional Networks : SDNNs. *Processing*
47. Zhang L, Li K, Xu Y et al (2015) Maximizing reliability with energy conservation for parallel task scheduling in a heterogeneous cluster. *Inf Sci (Ny)*. <https://doi.org/10.1016/j.ins.2015.02.023>
48. Chen C, Li K, Teo SG, et al (2018) Exploiting Spatio-temporal correlations with multiple 3D convolutional neural networks for citywide vehicle flow prediction. In: *Proceedings - IEEE international conference on data mining, ICDM*
49. Chen C, Li K, Ouyang A, et al (2016) GFlink: An In-memory computing architecture on heterogeneous CPU-GPU clusters for big data. In: *Proceedings of the international conference on parallel processing*

50. Yolcu U, Egrioglu E, Aladag CH (2013) A new linear and nonlinear artificial neural network model for time series forecasting. *Decis Support Syst*. <https://doi.org/10.1016/j.dss.2012.12.006>
51. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. *Nature*. <https://doi.org/10.1038/323533a0>
52. Shin Y, Ghosh J (1991) The pi-sigma network: an efficient higher-order neural network for pattern classification and function approximation. In: *Proceedings. IJCNN-91-Seattle: international joint conference on neural networks*
53. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9:1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
54. Broomhead D (1988) Multivariable functional interpolation and adaptive networks. *Complex Syst* 2
55. Yadav RN, Kalra PK, John J (2007) Time series prediction with single multiplicative neuron model. *Appl Soft Comput J*. <https://doi.org/10.1016/j.asoc.2006.01.003>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.