A Novel AutoCNN Model for Stock Market Index Prediction

Zakia Zouaghia
University of Tunis, Department of Computer Science SMART LAB, ISG, Tunisia

Zahra Kodia
University of Tunis, Department of Computer Science SMART LAB, ISG, Tunisia

Lamjed Ben Said
University of Tunis, Department of Computer Science SMART LAB, ISG, Tunisia

Abstract: Stock markets have a significant impact on the economic growth of countries. Predicting stock market indices has been a complex task in recent years. Indeed, many researchers and financial analysts are keenly interested in the research area of stock market prediction. In this paper, we propose a novel framework, titled AutoCNN, based on artificial intelligence techniques, to predict future stock market indices. AutoCNN is composed mainly of three stages: (1) A Convolutional Neural Network (CNN) for Automatic Feature Extraction; (2) The Halving Grid Search algorithm combined with a second CNN model for prediction of stock indices; and (3) Evaluation and recommendation. To validate our AutoCNN, we conduct experiments on two financial datasets that are extracted in the period between 2018 and 2023, which includes several events, such as economic, health and geopolitical international crises. The performance of the AutoCNN model is quantified using various metrics. It is benchmarked against different models and it proves to have strong prediction abilities. AutoCNN contributes to emerging technologies and innovation in the financial sector by automating decision-making, leveraging advanced pattern recognition, and enhancing the overall decision support system for investors in the digital economy.

Keywords: Stock index prediction, automated deep learning, crisis periods, financial decision-making, digitalised economy

Copyright © 2024
http://doi.org/10.18080/jtde.v12n1.843

Introduction

The financial sector plays an important role in the growth of economics of countries. A financial market is a physical or virtual place where market participants (buyers, sellers) meet to negotiate financial products (Grieger, 2003). Various economists and researchers have proven that there exists a link between financial development and economic growth (Agbloyor *et al.*, 2014). Stock markets, which are a particular type of financial market, are complicated financial businesses (Abraham *et al.*, 2001). There are various factors that affect the stock market indicators on a given day, such as industry performance, political changes, and economic variations (Goonatilake & Herath, 2007). Due to these factors, stock prices become highly unpredictable (Fathali *et al.*, 2022) and prediction is a challenging task.

In the literature, the stock price prediction approaches can be categorized into three types: (1) fundamental analysis; (2) technical analysis (Lawrence, 1997); and (3) machine learning models (Murkute & Sarode, 2015). The fundamental analysis considers economic factors as fundamentals and it is mostly employed for long-term predictions (Zouaghia *et al.*, 2023). The technical analysis is mainly based on charts that are able to identify trends and patterns in stock prices. However, the machine learning (ML) approach is based on artificial intelligence (AI) techniques for predicting stock market prices after building, training, and testing such a model. The process of stock index prediction can be seen as the prediction of future price variations by learning and analysing the historical stock data. Moreover, researchers (such as Kompella *et al.* 2019) demonstrated that traditional models did not generate an acceptable level of accuracy in the stock market prediction task and showed that models based on artificial intelligence techniques are more accurate.

In recent years, digital transformation has affected innovation in the majority of sectors of the economies of countries (Mgadmi *et al.*, 2021). Various researchers have demonstrated the impact of digital transformation on enhancing innovation in business (such as Li *et al.*, 2023) and implementing preventive tax risk management measures (Strauss *et al.*, 2020). Thus, the use of the Internet gives rise to digitization in the economy and it enhances the productivity of existing activities (Carlsson, 2004). Furthermore, the integration of artificial intelligence into economic activities represents a digitalization within the economic sector (Glushchenko *et al.*, 2020). In particular, the application of technologies in financial sectors, such as financial services, banks, companies and especially the financial markets, have recently given rise to the Financial Technology (Fintech) sphere (Abbasov *et al.*, 2020); it enhances economic growth (Chen *et al.*, 2022). As a result, the utilization of information with technologies has significantly facilitated the development of market economy strategies. Furthermore, digital technologies have also spurred business transformations across the value chains of all sectors.(Mukherjee *et al.*, 2023).

Recently, the use of advanced artificial intelligence techniques for business and, especially, to resolve financial problems have assisted us to have a global vision of future financial investments. Machine learning, which is a subfield of artificial intelligence, is one such tool that helps investors to make future stock market predictions (Leung *et al.*, 2014; Liu *et al.*, 2022; Mukherjee *et al.*, 2023). Deep learning is a subset of the machine learning technique, and most deep learning methods employ neural network architectures (Shinde & Shah, 2018). One of the most commonly practiced uses of neural networks are data prediction models, such as the forecasting of stock market prices based on stock historical data (Srivinay *et al.*, 2022). Neural networks are considered one of the common emerging generation of deep learning methods that demonstrate great abilities in solving complex problems (Aghapour *et al.*, 2023). Examples of neural-network models that are applied in stock market prediction are: Convolutional Neural Networks (CNNs); and Recurrent Neural Networks (RNNs), such as Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU) (Song & Choi, 2023), and Simple Recurrent Neural Network (SRNN) (Dey *et al.*, 2021). For this reason, we have chosen to apply a neural network-based model in our research work to enhance prediction in the stock market field. Investors need a more performant model to provide predictions to invest prudently during periods of uncertainty caused by many factors.

The originality of our contribution lies in:

1. CNN is applied in two different stages (the first step is in the task of automatic features extraction from financial stock data; the second step is in the task of stock index prediction). CNN is widely applied to extract features from charts (spatial dimension), but here we enhance its application to extract features from time series data (temporal dimension). It automates the process of features extraction. Additionally, CNN shows strong abilities in the task of prediction.
2. The Halving Grid Search (HGS) algorithm, which is rarely used in the literature, is hybridized with CNN to optimize its hyperparameters automatically. This combination of algorithms (CNN-HGS-CNN) was not applied previously in the task of stock index prediction.
3. The Automated Deep Learning model (AutoDL) is a recent artificial intelligence technique and has not been applied in the task of stock price prediction. A few papers in other fields (such as computer vision, natural language processing, speech recognition, and healthcare) use AutoDL and the results are prominent (Liu *et al.*, 2020; Faes *et al.*, 2019). Besides that, some authors automated a part of models and not all the process (Yi *et al.*, 2020).
4. AutoCNN can be considered as a valuable tool, because successful stock market prediction also requires an understanding of financial market behaviour. AutoCNN is trained on stock data that contains different crisis periods; thus it enhances the prediction during high volatility.

Copyright © 2024

5. The application of AutoCNN in predicting stock markets contributes to ongoing research in the intersection of finance and artificial intelligence. This research contributes to the development of novel combinations of hybrid models and insights, fostering innovation and knowledge advancement in both fields.

The rest of this paper is structured as follows. The following section describes a synopsis of related studies. The subsequent section details the proposed approach by analysing the data used and the steps applied to perform the task of stock index prediction. Section 4 details the conducted experiments and discusses the achieved results. Section 5 talks about limitations of the proposed approach. Section 6 presents the conclusion and proposes future directions. The final section has three appendices.

Related Literature

In the literature, various machine and deep learning models were suggested to predict future stock price indices of stock markets. Generally, two types of deep learning neural networks are used, namely CNNs and RNNs. These two networks have lately shown strong abilities in the task of stock index prediction (Zouaghia *et al.*, 2023).

The SRNN algorithm was proposed by Elman (1998); hence, it is called an *Elman Network*. It has shown strong abilities in identifying sequential data insights and predicting stock prices. The LSTM model is one of the most successful RNN architectures (Fathali *et al.*, 2022) and it is widely applied in the task of prediction of sequential data. The memory mechanism in LSTMs plays a crucial role in their ability to handle sequential data tasks such as time series prediction (Ozbayoglu *et al.*, 2020). The GRU model is a variant of RNN that was proposed in 2014 (Samarawickrama & Fernando, 2017). It excels in sequential data modelling, particularly in scenarios where fast training, computational efficiency, and effective handling of short-term dependencies are priorities.

Samarawickrama & Fernando (2017) proposed a RNN approach for predicting daily stock prices of some selected listed companies of the Colombo Stock Exchange (CSE). Experiments were conducted on three models: SRNN, GRU and LSTM networks. The authors chose as input three variables (Low, High, and Close prices).

Hoseinzade & Haratizadeh (2019) proposed a CNN-based framework that applied the CNN to gather data from a variety of sources from different stock markets, such as S&P500, NASDAQ, DJI, NYSE, and RUSSELL indices. After that, the model was applied to predict the next day's direction of movement for the considered indices. The authors tested two principal configurations of the CNN model: 2D-CNN and 3D-CNN. Their models were evaluated using the F-measure metric.

Copyright © 2024

Gao *et al.* (2021) applied an optimized GRU and LSTM models based on the use of various technical indicators with financial data. These two models were merged with two other methods, LASSO and Principal Component Analysis (PCA), to optimize their hyperparameters.

Fathali, Kodia & Ben Said (2022) applied the RNN, LSTM, and CNN models to predict the future closing stock price trends of NIFTY 50. Then, they compared the achieved results using four regression metrics (MSE, RMSE, MAE and R2). The process of selecting features from financial data and tuning the hyperparameters of the models was conducted through manual tests.

Kumar *et al.* (2022) proposed a traditional time-series model, Auto Regressive Integrated Moving Average (ARIMA), combined with a neural network model (LSTM) to predict the stock market. For the task of hyperparameter selection, the Artificial Bee Colony (ABC) algorithm using differential evolution (DE) was applied. Their proposed model was tested on various financial datasets, such as the NASDAQ index, and benchmarked against other models like ARIMA.

Zouaghia *et al.* (2023) applied a hybrid deep learning (DL) model to predict the stock index of NASDAQ. They combined a CNN with four architectures of RNNs (GRU, Bidirectional GRU, LSTM, and Bidirectional LSTM). They used a 1D-CNN model to extract automatically the more accurate features from historical stock data. Then, the output of the 1D-CNN model was used as input to RNNs to predict the future stock index. For the hyperparameter optimization, they used a manual search strategy. To evaluate their models, they applied six metrics: ET, MSE, RMSE, MAPE, MAE and R^2.

Proposed Approach

In this section, we detail our proposed approach based on the AutoCNN model. This framework is used to predict future stock indices during periods of uncertainty and fear, especially during turbulent times due to external factors. The general architecture of AutoCNN is detailed in Figure 1. This framework is composed of a cycle, which is primarily based on three stages: (1) CNN for automatic feature extraction; (2) a HGS method combined with a second CNN for stock index prediction; and (3) the step of evaluation and comparison. Our framework performs many iterations to screen which is the best CNN model topology.

Copyright © 2024
http://doi.org/10.18080/jtde.v12n1.843

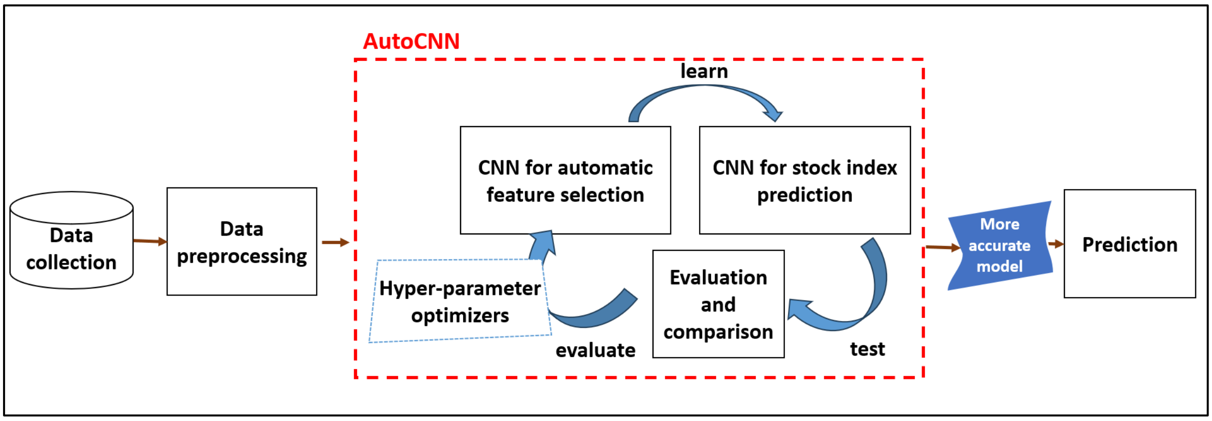

Figure 1. The general architecture of the AutoCNN framework

The proposed solution leads to assistance to financial decision makers to acquire a global vision about the actual and future fluctuations of stock indices during crisis periods and, then, to make the right decision with more confidence under uncertainty. AutoCNN is benchmarked against three other implemented neural networks (LSTM, GRU, and SRNN) in addition to existing models in the literature. The components of our framework are described below.

Stage 1: CNN for automatic feature extraction

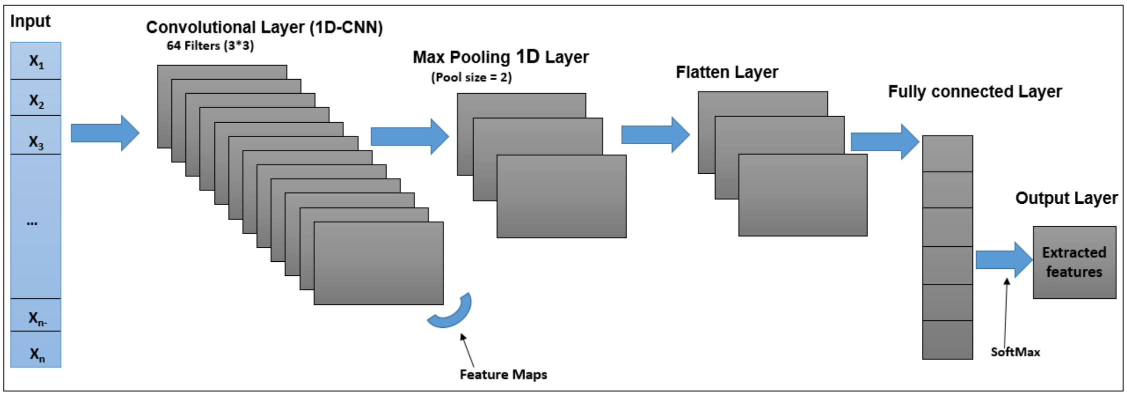

Figure 2. The architecture of 1D-CNN model for automatic feature extraction

The process of Feature Extraction is one of the most challenging problems in deep learning modelling, especially in time-series forecasting. This issue is tackled by several researchers, who proposed various approaches. The CNN model is proposed by Krizhevsky *et al.* (2017) to extract features from images but, in this work, we extract features from financial data. Time-series prediction is considered one of the most frequently applied uses of one dimensional (1D) CNN in practice (Markova, 2022). In this work, we choose the architecture of 1D-CNN for the step of automatic feature extraction. This configuration has strong capabilities to learn, understand and extract features from the input stack variables, owing to its hierarchical architecture and the nature of stock price data. Our choice is based on the work of Zouaghia *et al.* (2023). Figure 2 explains this architecture.

Copyright © 2024
http://doi.org/10.18080/jtde.v12n1.843

Stage 2: An optimized CNN for stock index prediction

In AutoCNN, the CNN model is not only used for the step of automatic feature extraction, but also we undertake stock index prediction using this model. According to Vargas *et al.* (2017) and Wu *et al.* (2021), the CNN model has shown good abilities in stock price prediction. Further, to perform this task, the output of the first 1D-CNN model, which is the extracted features, is used as input for a second optimized 1D-CNN model (HGS-CNN).

Hyperparameter optimization is a crucial step in the process of training machine-learning models. This step specifies many aspects of the learning algorithm and the model's architecture, such as the learning rate, the number of units in hidden layers, the optimizer and number of filters in the CNN model. In AutoCNN, we adopt the HGS algorithm for automatic optimization of its hyperparameters. The principle of this method is successive halving processes (Pedregosa *et al.*, 2011). In this work, we adopt this method to optimize hyperparameters of the considered neural network-based models for several reasons, namely:

1. It has been rarely applied in scientific research works (Jung *et al.*, 2023) and it has shown strong ability in the task of tuning optimized hyperparameters.
2. The evidence behind this choice is its computational efficiency, time saving, adaptability to high-dimensional search spaces, robustness to noisy metrics, and effective balance between exploration and exploitation.

Stage 3: Evaluation and comparison

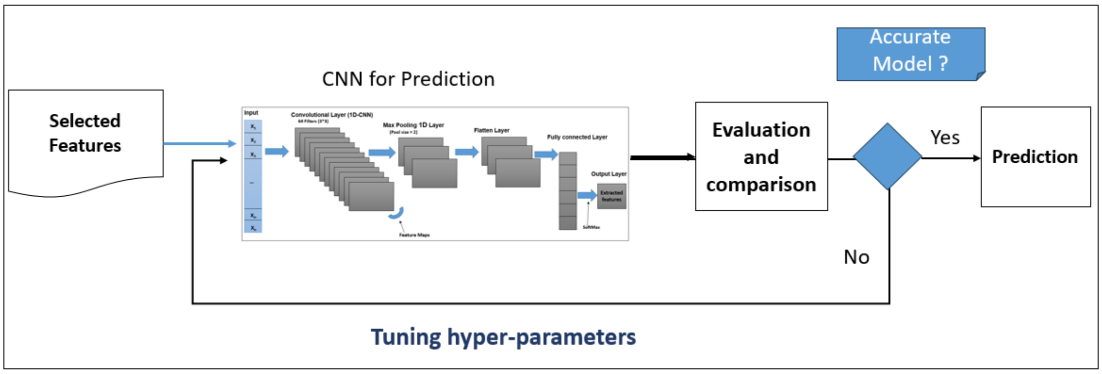

Figure 3. Hyperparameter optimization process in AutoCNN framework

For the evaluation and comparison task, we use various measures to obtain a more accurate model. Figure 3 details this process: after n iterations, if the considered measure does not satisfy a threshold of error acceptance, a new iteration of hyperparameter tuning is conducted using the HGS algorithm. The final output of the AutoCNN framework is a more accurate and stable model and the predicted stock index values. If the model generated accurate predictions, it is recommended to be applied by investors.

Experiments and Results

In this section, we present the datasets used to test AutoCNN. The experiments conducted and the achieved results are discussed, referring to various metrics.

Experimental environment

Our experiments are conducted on a computer with the following characteristics: Intel Core i7 7th Gen 2.8 GHz, 32 GBs of RAM, and the operating system of Microsoft Windows 10 Professional. The implementation has been made in Python programming language based on various libraries. The main used libraries are: (1) *Sklearn* to process data and select metrics; (2) *Keras* to select neural networks models and layers; (3) *Tensorflow* to build, train, and deploy deep learning models; (4) *Time* to compute the execution time of each model; (5) *Matplotlib* to create static and interactive visualizations; and (6) *statsmodels* to conduct a statistical test, like the ADF and the stepwise tests, etc.

Financial data description and analysis

Visualizing financial data is a powerful technique that allows us to foresee a clear perspective of stock price changes in response to certain events or external factors, and data analysis helps us to choose the more adequate model for the task of stock market index prediction.

Data gathering

In this paper, we consider two datasets of two different stock exchanges: (1) Shanghai Stock Exchange (SSE) index; and (2) National Association of Securities Dealers Automated Quotations (NASDAQ) index, covering the period between 2018 and 2023. Data are collected from Yahoo Finance Website and include seven variables as described in Table 1.

Table 1. Parameters of the used financial data

Variable	Description
Date	The trading date
Open	The opening price of the stock, at the start of a trading day
High	The highest price of the stock, during a trading day
Low	The lowest price of the stock, on a trading day
Close	The closing price of the stock, at the end of a trading day
Adj Close	The closing price after adjustments: the stock's value after distributing dividends
Volume	The amount of Stock traded in the market during a period

Data analysis

Researchers have proven that neural networks are suited to non-stationary data (Reid *et al.*, 2014; Kurle *et al.*, 2019 ; Zhou *et al.*, 2021). In this situation of non-stationarity, stock price

Copyright © 2024

data are hard to be predicted and CNNs are suitable for capturing insights for modelling short-term dependencies in data.

A statistical test is used to analyse the stationarity in time series of our two stock indices. To analyse the stationarity of the predicted variable (closing price), the Augmented Dickey-Fuller (ADF) test is applied. The ADF test is a common type of statistical test called a unit root test. It is used to identify the presence of a unit root and transform the series into a stationary state (Dadhich *et al.*, 2021). It is based on two hypotheses (H_0 and H_1):

- Null Hypothesis (H_0): The time series has a unit root, indicating that it is non-stationary.
- Alternative Hypothesis (H_1): The time series does not have a unit root, indicating that it is stationary.

The null hypothesis is evaluated through a calculated t-statistic, which is calculated by the formula as seen in the paper of Reddy (2019). If the t-statistic calculated is bigger than the critical value, we accept the hypothesis H_0 and data are non-stationary. On the other hand, if the t-statistic calculated is less than the critical value, we reject the H_0 and data are stationary. Additionally, "P-value" is also used to reject or accept the H_0: if the "P-value" is less than 0.05 ($P<0.05$), we reject the H_0; and vice-versa. Table 2 shows the generated values of the test ADF for both SSE and NASDAQ indices.

Table 2. ADF test

SSE index				NASDAQ index			
		t-statistic	Prob.*			t-statistic	Prob.*
ADF test statistic		-1.695	0.748	ADF test statistic		-1.487	0.540
Test critical values	1%	-3.435		Test critical values	1%	-3.435	
	5%	-2.863			5%	-2.863	
	10%	-2.568			10%	-2.568	

It is shown from Table 2 that the calculated "t-statistic" value is greater than critical values at 1%, 5% and 10% levels of significance. Therefore, we can conclude that data of SSE and NASDAQ indices are non-stationary. Further, the "P-value" is also greater than 0.05. So, in our case, we do not reject the H_0: this means that data has a unit root and is non-stationary.

Reaction of the stock markets to the last two crises

In this part, we analyse the impact of the last two crises (the international health crisis of the COVID-19 pandemic; and the international conflict between Russia and Ukraine) on the variation of prices in stock markets, because our AutoCNN model is trained on data covering these two major events, contributing to improved generalization, its adaptability, and robustness. Therefore, AutoCNN can be able to predict accurately future stock indices during uncertainty.

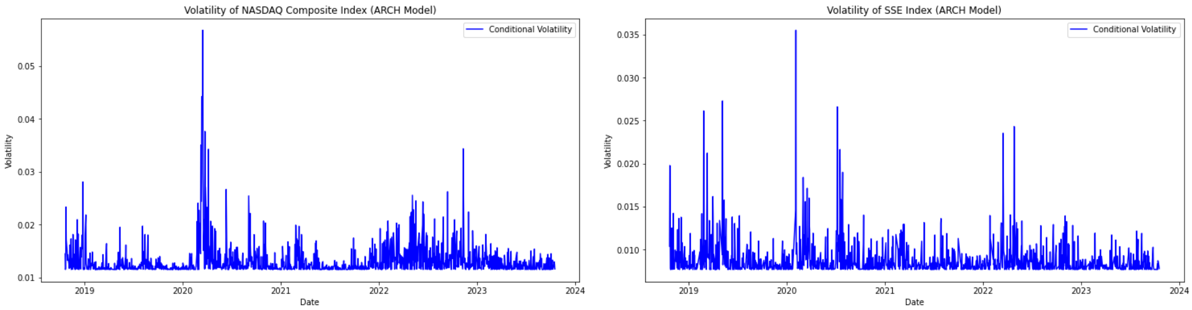

Figure 4. The calculated volatility of NASDAQ and SSE indices

First; an ARCH(1) model, which is a type of autoregressive conditional heteroskedasticity model, is designed to capture time-varying volatility in financial time-series data. Figure 4 visualizes how the volatility changes over different periods from October 2018 until October 2023.

The volatility provides insights into risk during crisis periods and into potential trading opportunities. Further, high volatility might represent increased risk but also, in some cases, it indicates potential profit opportunities for traders; low volatility reflects stability. The volatility of stock prices is considered as a fear gauge of investors. From Figure 4, it is clear that the stock prices of NASDAQ and SSE indices are more volatile during the periods of the global health crisis (COVID-19 pandemic) especially between 2020 and 2021. Additionally, since February and March 2022, it is seen that the variation is also more pronounced due to the recent geopolitical crises, such as the Russian, Ukrainian, European and American conflicts. Thus, due to these two major factors, investors become more disturbed and cannot easily take the right decisions (to buy or sell a given stock), which explains not only the rapid fluctuations and shocks in stock prices but also the suffered losses as seen in Figure 5 (represented by negative values). In this case, investors and financial analysts need help in making financial decisions in an uncertain environment.

Copyright © 2024
http://doi.org/10.18080/jtde.v12n1.843

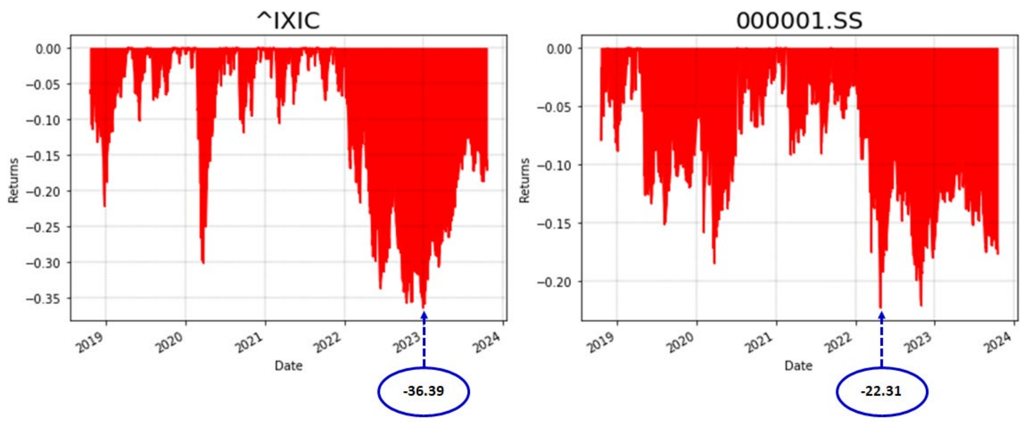

Figure 5. The calculated drawdown falls for NASDAQ and SSE indices

The drawdown for a stock market index represents the peak-to-trough decline in the index's value during a specific period. It measures the maximum percentage decrease from the highest point to the lowest point before a new peak is achieved. NASDAQ (^IXIC) and SSE (000001.SS) indices absorbed several losses, and the maximum losses are 36.39% and 22.31%, respectively, for NASDAQ and SSE indices, as shown in Figure 5.

Evaluation criteria

AutoCNN is evaluated using the Execution Time (ET) metric, four regression evaluation metrics — Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) — and the Coefficient of Determination (R^2). The formulas for calculation are shown in Table 2.

Table 3. The evaluation metrics used

Metric	Formula
ET	Execution Time =Current time - Start time
MSE	$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2$
RMSE	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2}$
MAE	$MAE = \frac{1}{n}\sum_{i=1}^{n}\lvert x_i - y_i\rvert$
MAPE	$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\frac{\lvert x_i - y_i\rvert}{x_i}$
R^2	$R^2 = \frac{\sum_{i=1}^{n}(y_i - x_i)^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$

Note: n is the number of data points in the test set, x_i is the actual value, $\overline{x}$ is the mean value of the actual values, and y_i is the predicted value.

MSE, RMSE, MAE, MAPE and R^2 metrics are used for model evaluation after the model has been trained. Once the implemented models are trained, these metrics are used to assess how well the models generalize to new, unseen data (the principle of supervised machine learning models). These metrics provide a quantitative measure (Zouaghia *et al.*, 2023) of the AutoCNN's performance on subset test data. Moreover, metrics like MAPE provide insights into the percentage error, which can be more interpretable from a business perspective. This is particularly important in financial applications where understanding the relative size of errors is valuable.

Smaller values of ET, MSE, RMSE, MAE and MAPE designate a better result. However, greater values of R^2 indicate a better result. Further, concerning the ET metric, it measures the cost of each implemented model in terms of time (seconds) that depends on different components, such as the size of the used dataset, the complexity of each model, and the power of the hardware used for our experiments.

Additionally, a stepwise multiple testing method (a statistical technique) is used to conduct multiple statistical tests of AutoCNN to assess various aspects of model's performance. Figure 6 shows all the steps used to evaluate our model using this statistical method. In step 8, the Holm method is selected, which is a step-down procedure that adjusts the significance level based on the order of hypotheses, for adjusting p-values to control the family-wise error rate.

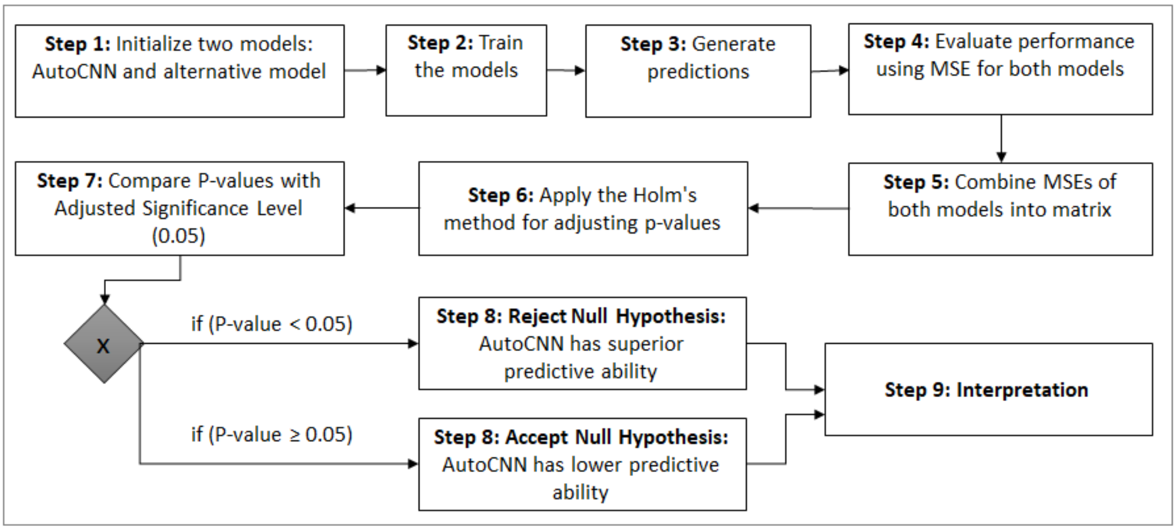

Figure 6. Flowchart for the stepwise multiple testing method

Auto-generated model parameters

The chosen parameters for AutoCNN, used in common for all comparative models, are detailed in Table 3.

Table 4. Parameters common to all models

Parameter	**Value**
Prediction frequency	Daily
Dataset	SSE index and NASDAQ Composite index

Parameter	Value
Data specification	From 2018 to 2023
Normalisation	[0-1]
Input features	Automatic selection using CNN
Target variable	Close price
Training: Testing	80%: 20%

The structure of the AutoCNN model is comprised of ten layers (5 layers in the task of automatic feature extraction; and 5 layers in the task of stock index prediction). Additionally, changes in data have an impact on the robustness and stability of the model, because AutoCNN is trained on data containing sensitivity periods, like the international COVID 19 pandemic health crisis especially during the year 2020, or the last geopolitical crisis of the Russia-Ukraine conflict during 2022. Our models are trained through these changes (correlation) in stock data in order to be able to predict future stock indices in similar cases using AutoCNN.

Results and discussion

We provide comparative results between AutoCNN and other benchmarked models: LSTM, GRU, SRNN, and ARIMA. Results are summarized in Tables A1 and A2 (Appendix A), where the best performance results are marked in bold for each stock market index per metric. In Tables A1 and A2, it is clearly seen that AutoCNN outperforms the three benchmarked neural network models (CNN-LSTM, CNN-GRU, and CNN-SRNN). For the SSE index, AutoCNN generated 0.00001, 0.003, 0.002, 0.4%, and 0.998, respectively, for the MSE, RMSE, MAE, MAPE, and R^2 indicators. For the NASDAQ Composite index, AutoCNN generated 0.00001, 0.004, 0.003, 0.53%, and 0.999, respectively, for the MSE, RMSE, MAE, MAPE, and R^2 indicators. Moreover, concerning the execution time metric, it is seen that AutoCNN is trained with the minimum time, less than two seconds for both the stock indices. AutoCNN did not only outperform the implemented neural network models, but also it achieved superior performance compared to the traditional ARIMA model (Lv *et al.*, 2022; Kumar *et al.*, 2022).

Additionally, Figures B1, B2, B3 and B4 (Appendix B) plot the real and predicted values of SSE and NASDAQ indices using neural network models based on the test set of data. The blue line indicates the actual stock prices, while the red and orange lines represent the predicted stock prices. In these four figures, it is shown that the AutoCNN model performs better than the other models considered for comparison; especially, Figure B1 shows visually that the predicted values and the real values of stock prices are very close. AutoCNN was successful in capturing the pattern in all the periods of the data.

To compare the implemented models in more detail, Figures C1, C2, C3, C4, C5 and C6 (Appendix C) provide a graphical visualization of the results obtained for each metric to evaluate and compare the performance of each model. From these Figures, it is clear that

AutoCNN generates the lower error rates in terms of MSE, RMSE, MAE, R^2, and MAPE; and is trained in a very short time compared to other networks considered for comparison.

It is observed through our experiments that AutoCNN always generates the superior prediction results in the Chinese context (SSE index), as well as in the American context (NASDAQ Composite index), compared to the other benchmarked models. The theoretical foundation behind achieving different results with CNN and in a Chinese context or an American context, in comparison to other research works in the literature, is that we did not use the same configuration or the same hyperparameter values, because this process is handled automatically. AutoCNN is able to detect correlations between stock price variations across temporal dependencies in CNN.

Additionally, referring to the stepwise multiple testing procedure of Romano & Wolf (2005) and in order to prove the superior predictive ability of AutoCNN using a statistical method, we benchmarked our AutoCNN model against a very popular and applied machine learning model (Support Vector Regression, SVR) in prediction. In this paper, SVR shows strong abilities in the task of stock index prediction (Zheng *et al.*, 2005). In our experiments, the SVR model is trained on the same data (NASDAQ index and SSE index), then it is evaluated alongside our AutoCNN model for predicting the stock prices. Through our experiments, the adjusted p-values for the model comparisons are provided in Table A3 (Appendix A).

In Table A3, for the NASDAQ Composite index, the Adjusted p-value= $3.16e\text{-}05<0.05$: this provides strong statistical evidence to reject the null hypothesis, indicating a significant difference in predictive ability between the AutoCNN model and the alternative model (SVR). There is the same interpretation for the results using the SSE index. For the SSE index data, the adjusted p-value=$2.03e\text{-}05<0.05$: this indicates strong evidence against the null hypothesis — the AutoCNN model is deemed to have a significantly different predictive ability compared to the SVR model.

To summarize, AutoCNN is experimented on with two different stock index datasets, from different countries, and it is proven that this model is robust and stable, because, when we change the input, it always generated superior results. We argue that this hybrid auto-generated model can be reproduced on other indices or financial time series.

Overall, AutoCNN is able to better understand the dynamical variations in daily SSE index and NASDAQ index that are caused by external factors, especially in crisis periods. AutoCNN generated results in the task of stock index prediction that are superior to those from other implemented neural networks (SRNN, GRU and LSTM), due to its ability to effectively capture local patterns and temporal dependencies within the financial data. It excels at feature

Copyright © 2024
http://doi.org/10.18080/jtde.v12n1.843

extraction, automatically learning hierarchical representations of input data, which is beneficial for identifying short-term trends and specific features influencing stock price movements. Additionally, AutoCNN requires less preprocessing and can be computationally more efficient, contributing to its superior performance in the stock prediction scenario. Thus, the hybridization of the halving grid search technique with a CNN has demonstrated robust capabilities in optimizing model hyperparameters effectively, with reduced computational cost compared to exhaustive methods such as the grid search technique, which is greedy in terms of resource use.

Limitations of the Proposed Approach

AutoCNN can be considered as a financial decision support system that can help investors or financial institutions. It aims to enhance the decision-making processes in the context of stock index prediction and it involves several key aspects, all of which contribute to more effective and successful investment strategies, such as:

1. Informed decision-making;
2. Risk mitigation;
3. Optimized Timing;
4. Adaptation to Market Dynamics.

Nevertheless, AutoCNN is not without limitations. One notable constraint is the inherent difficulty in capturing the dynamic and intricate nature of financial markets solely through the analysis of historical price charts (combining fundamental analysis with artificial intelligence techniques). AutoCNN may struggle to discern the complex temporal dependencies and subtle market trends that influence stock prices. The explainability of the AutoCNN model represents a prominent issue for future work. Moreover, stock prices are influenced by a multitude of factors, such as economic indicators and market sentiment, which may not be adequately represented in the input data for the AutoCNN model. Therefore, while this approach offers a promising avenue for stock price prediction, it is crucial to acknowledge these limitations and consider complementary approaches, such as incorporating fundamental analysis or sentiment analysis, to enhance the model's robustness and reduce the risk of overestimating its predictive capabilities.

Conclusions and Orientation for Future Research

Recent technological advances, such as the deep learning approach, contribute to digital transformation of the economy sectors, including finance. Deep learning has become widely applied and it is considered a motive power in the current digital economy. The purpose of this paper is to use the automated deep learning (AutoDL) technique to propose an intelligent financial decision-support system to investors, to help them to make accurate decisions on financial investment during periods of uncertainty.

Deep learning models, and especially CNN, are applied in this research work to propose a robust and stable solution, named AutoCNN, which is used in three steps, namely: (1) CNN for Automatic Feature Extraction; (2) HGS algorithm hybridized with CNN for stock price prediction; and (3) Evaluation and comparison. AutoCNN is firstly benchmarked against five models (LSTM, GRU, SRNN, ARIMA, and SVR), using five regression metrics (MSE, RMSE, MAE, R^2, and MAPE) , an execution time (ET) metric, and a statistical method (stepwise multiple testing procedure). After conducting several experiments, we have realized that the proposed AutoCNN can stably predict the stock index prices under uncertainty. Therefore, we recommend it to be used by investors, without demand for expert knowledge because all the process is automated. The benefits from this work are to prove the effectiveness of applying neural network architectures to offer reliable predictions for financial decision makers in the digitalised economy during crisis periods.

For future work, we propose an extension of this paper by including other emergent neural network architectures, such as the Spiking Neural Networks (SNNs), and involving other stock exchange indexes.

References

Abbasov, A., Mamedov, Z., & Kovalenko, V. (2020). Financial innovation in the conditions of digitalization of the economy. Economic and Social Development: Book of Proceedings, 515–524. Retrieved from https://www.esd-conference.com/upload/book_of_proceedings/Book_of_Proceedings_esdBaku2020_Vol1_Online.pdf

Abraham, A., Nath, B., & Mahanti, P. K. (2001). Hybrid intelligent systems for stock market analysis. In Computational Science-ICCS 2001: International Conference San Francisco, CA, USA, May 28—30, 2001 Proceedings, Part II 1 (pp. 337–345). Springer: Berlin, Heidelberg. https://doi.org/10.1007/3-540-45718-6_38

Agbloyor, E. K., Abor, J. Y., Adjasi, C. K. D., & Yawson, A. (2014). Private capital flows and economic growth in Africa: The role of domestic financial markets. *Journal of International Financial Markets, Institutions and Money*, *30*, 137–152. https://doi.org/10.1016/j.intfin.2014.02.003

Aghapour, A., Arian, H. R., & Seco, L. A. (2023). Deep-Time Neural Networks: An Efficient Approach for Solving High-Dimensional PDEs. Available at SSRN. http://dx.doi.org/10.2139/ssrn.4554574

Carlsson, B. (2004). The Digital Economy: what is new and what is not? *Structural change and economic dynamics*, *15*(3), 245–264. https://doi.org/10.1016/j.strueco.2004.02.001

Chen, X., Yan, D., & Chen, W. (2022). Can the digital economy promote FinTech development? *Growth and Change*, *53*(1), 221–247. https://doi.org/10.1111/grow.12582

Dadhich, M., Pahwa, M. S., Jain, V., & Doshi, R. (2021). Predictive models for stock market index using stochastic time series ARIMA modeling in emerging economy. In Advances in Mechanical Engineering: Select Proceedings of CAMSE 2020 (pp. 281-290). Springer Singapore. https://doi.org/10.1007/978-981-16-0942-8_26

Dey, P., Hossain, E., Hossain, M. I., Chowdhury, M. A., Alam, M. S., Hossain, M. S., & Andersson, K. (2021). Comparative analysis of recurrent neural networks in stock price prediction for different frequency domains. Algorithms, *14*(8), 251. https://doi.org/10.3390/a14080251

Elman, J. (1998). Generalization, simple recurrent networks, and the emergence of structure. In Proceedings of the twentieth annual conference of the Cognitive Science Society (p. 6). Mahwah, NJ: Lawrence Erlbaum Associates. Retrieved from https://langev.com/pdf/elman98generalizationSimple.pdf

Faes, L., Wagner, S. K., Fu, D. J., Liu, X., Korot, E., Ledsam, J. R., ... & Keane, P. A. (2019). Automated deep learning design for medical image classification by health-care professionals with no coding experience: a feasibility study. *The Lancet Digital Health*, *1*(5), e232–e242. https://doi.org/10.1016/S2589-7500(19)30108-6

Fathali, Z., Kodia, Z., & Ben Said, L. (2022). Stock market prediction of Nifty 50 index applying machine learning techniques. *Applied Artificial Intelligence*, *36*(1), 2111134. https://doi.org/10.1080/08839514.2022.2111134

Gao, Y., Wang, R., & Zhou, E. (2021). Stock prediction based on optimized LSTM and GRU models. *Scientific Programming*, *2021*, 1–8. https://doi.org/10.1155/2021/4055281

Glushchenko, T. E., Khodarinova, N. V., Ishchenko, O. V., Shaposhnikov, V. L., & Aksenova, Z. A. (2020). The development of cooperation in the digital economy based on scientific research by av chayanov. In Frontier Information Technology and Systems Research in Cooperative Economics (pp. 43–52). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-57831-2_5

Goonatilake, R., & Herath, S. (2007). The volatility of the stock market and news. *International Research Journal of Finance and Economics*, *3*(11), 53–65. Retrieved from https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=567df765560dfb426c184bd34b47510eaaabb500

Grieger, M. (2003). Electronic marketplaces: A literature review and a call for supply chain management research. *European Journal of Operational Research*, *144*(2), 280–294. https://doi.org/10.1016/S0377-2217(02)00394-6

Hoseinzade, E., & Haratizadeh, S. (2019). CNNpred: CNN-based stock market prediction using a diverse set of variables. *Expert Systems with Applications*, *129*, 273–285. https://doi.org/10.1016/j.eswa.2019.03.029

Jung, J. O., Crnovrsanin, N., Wirsik, N. M., Nienhüser, H., Peters, L., Popp, F., ... & Schmidt, T. (2023). Machine learning for optimized individual survival prediction in resectable upper gastrointestinal cancer. *Journal of Cancer Research and Clinical Oncology*, *149*(5), 1691–1702. https://doi.org/10.1007/s00432-022-04063-5

Kompella, S., & Chakravarthy Chilukuri, K. C. C. (2020). Stock market prediction using machine learning methods. *International Journal of Computer Engineering and Technology*, *10*(3), 2019. Available at SSRN: https://ssrn.com/abstract=3554945

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 1097–1105. https://doi.org/10.1145/3065386 doi: 10.1145/3065386

Kumar, R., Kumar, P., & Kumar, Y. (2022). Three stage fusion for effective time series forecasting using Bi-LSTM-ARIMA and improved DE-ABC algorithm. *Neural Computing and Applications*, *34*(21), 18421–18437. https://doi.org/10.1007/s00521-022-07431-x

Kurle, R., Cseke, B., Klushyn, A., Van Der Smagt, P., & Günnemann, S. (2019). Continual learning with Bayesian neural networks for non-stationary data. In International Conference on Learning Representations (pp. 1–28). Retrieved from https://openreview.net/pdf?id=SJlsFpVtDB

Lawrence, R. (1997). Using neural networks to forecast stock market prices. University of Manitoba, 333(2006), 2013. Available from https://cmps-people.ok.ubc.ca /rlawrenc/research/Papers/nn.pdf

Leung, C. K. S., MacKinnon, R. K., & Wang, Y. (2014, July). A machine learning approach for stock price prediction. In Proceedings of the 18th International Database Engineering & Applications Symposium (pp. 274-277). https://doi.org/10.1145/2628194.2628211

Li, S., Gao, L., Han, C., Gupta, B., Alhalabi, W., & Almakdi, S. (2023). Exploring the effect of digital transformation on Firms' innovation performance. *Journal of Innovation & Knowledge*, *8*(1), 100317. https://doi.org/10.1016/j.jik.2023.100317

Liu, Q., Tao, Z., Tse, Y., & Wang, C. (2022). Stock market prediction with deep learning: The case of China. *Finance Research Letters*, *46*, 102209. https://doi.org/10.1016/j.frl.2021.102209

Liu, Z., Xu, Z., Rajaa, S., Madadi, M., Junior, J. C. J., Escalera, S., ... & Guyon, I. (2020, 08–14 Dec). Towards automated deep learning: Analysis of the autodl challenge series 2019. In H. J. Escalante & R. Hadsell (Eds.), Proceedings of the neurips 2019 competition and demonstration track (Vol. 123, pp. 242–252). PMLR. Retrieved from https://proceedings.mlr.press/v123/liu20a.html

Lv, P., Wu, Q., Xu, J., & Shu, Y. (2022). Stock index prediction based on time series decomposition and hybrid model. *Entropy*, *24*(2), 146. https://doi.org /10.3390/e24020146

Markova, M. (2022, April). Convolutional neural networks for forex time series forecasting. In AIP Conference Proceedings (Volume 2459, No. 1). AIP Publishing. https://doi.org/10.1063/5.0083533

Mgadmi, N., Moussa, W., Béjaoui, A., Sadraoui, T., & Afef, G. (2021). Revisiting the nexus between digital economy and economic prosperity: evidence from a comparative analysis. *Journal of Telecommunications and the Digital Economy*, *9*(2), 69–91. https://doi.org/10.18080/jtde.v9n2.384

Mukherjee, S., Sadhukhan, B., Sarkar, N., Roy, D., & De, S. (2023). Stock market prediction using deep learning algorithms. *CAAI Transactions on Intelligence Technology*, *8*(1), 82–94. https://doi.org/10.1049/cit2.12059

Murkute, A., & Sarode, T. (2015). Forecasting market price of stock using artificial neural network. *International Journal of Computer Applications*, *124*(12), 11–15. https://doi.org/10.5120/ijca2015905681

Ozbayoglu, A. M., Gudelek, M. U., & Sezer, O. B. (2020). Deep learning for financial applications: A survey. *Applied Soft Computing*, *93*, 106384. https://doi.org/10.1016/j.asoc.2020.106384

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, *12*, 2825–2830.

Reddy, C. V. (2019). Predicting the stock market index using stochastic time series ARIMA modelling: The sample of BSE and NSE. Indian Journal of Finance, *13*(8), 7–25. http://dx.doi.org/10.2139/ssrn.3451677

Reid, D., Hussain, A. J., & Tawfik, H. (2014). Financial time series prediction using spiking neural networks. *PloS one*, *9*(8), e103656. https://doi.org/10.1371/journal.pone.0103656

Romano, J. P., & Wolf, M. (2005). Stepwise multiple testing as formalized data snooping. *Econometrica*, *73*(4), 1237–1282. https://doi.org/10.1111/j.1468-0262.2005.00615.x

Samarawickrama, A. J. P., & Fernando, T. G. I. (2017, December). A recurrent neural network approach in predicting daily stock prices an application to the Sri Lankan stock market. In 2017 IEEE International Conference on Industrial and Information Systems (ICIIS) (pp. 1–6). IEEE. https://doi.org/10.1109/ICIINFS.2017.8300345

Shinde, P. P., & Shah, S. (2018, August). A review of machine learning and deep learning applications. In 2018 Fourth international conference on computing communication control and automation (ICCUBEA) (pp. 1–6). IEEE. https://doi.org/10.1109/ICCUBEA.2018.8697857

Srivinay, Manujakshi, B. C., Kabadi, M. G., & Naik, N. (2022). A hybrid stock price prediction model based on pre and deep neural network. *Data*, *7*(5), 51. https://doi.org/10.3390/data7050051

Strauss, H., Fawcett, T., & Schutte, D. (2020). Tax risk assessment and assurance reform in response to the digitalised economy. *Journal of Telecommunications and the Digital Economy*, *8*(4), 96–126. https://doi.org/10.18080/jtde.v8n4.306

Vargas, M. R., De Lima, B. S., & Evsukoff, A. G. (2017, June). Deep learning for stock market prediction from financial news articles. In 2017 IEEE international conference on computational intelligence and virtual environments for measurement systems and applications (CIVEMSA) (pp. 60–65). IEEE. https://doi.org/10.1109/CIVEMSA.2017.7995302

Vijh, M., Chandola, D., Tikkiwal, V. A., & Kumar, A. (2020). Stock closing price prediction using machine learning techniques. *Procedia computer science*, *167*, 599–606. https://doi.org/10.1016/j.procs.2020.03.326

Wu, J. M. T., Li, Z., Srivastava, G., Tasi, M. H., & Lin, J. C. W. (2021). A graph-based convolutional neural network stock price prediction with leading indicators. *Software: Practice and Experience*, *51*(3), 628–644. https://doi.org/10.1002/spe.2915

Yi, H., & Bui, K. H. N. (2020). An automated hyperparameter search-based deep learning model for highway traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, *22*(9), 5486–5495. https://doi.org/10.1109/TITS.2020.2987614

Zheng, J., Wang, Y., Li, S., & Chen, H. (2021). The stock index prediction based on SVR model with bat optimization algorithm. *Algorithms*, *14*(10), 299. https://doi.org/10.3390/a14100299

Zhou, F., Zhou, H., Yang, Z., & Gu, L. (2021). IF2CNN: Towards non-stationary time series Feature Selection by integrating iterative filtering and convolutional neural networks. *Expert Systems with Applications*, *170*, 114527. https://doi.org/10.1016/j.eswa.2020.114527

Zouaghia, Z., Aouina, Z. K., & Said, L. B. (2023, October). Hybrid Machine Learning Model for Predicting NASDAQ Composite Index. In 2023 International Symposium on Networks, Computers and Communications (ISNCC) (pp. 1–6). IEEE. https://doi.org/10.1109/ISNCC58260.2023.10323903

Appendices

In this section, we include Tables and Figures concerning the comparison of results.

Appendix A

Table A1. Comparison of validation metrics of AutoCNN against benchmarked models for SSE index

Model		Metrics				
	MSE	RMSE	MAE	R^2	MAPE (%)	ET (second)
AutoCNN	**0.00001**	**0.003**	**0.002**	**0.998**	**0.4%**	**1.63**
LSTM	0.00008	0.009	0.006	0.986	1.1%	21.4
GRU	0.00008	0.009	0.007	0.985	1.3%	20.5
SRNN	0.00007	0.008	0.007	0.987	1.2%	8.09
ARIMA (Lv *et al.*, 2022)	-	36.981	25.102	0.9690	0.8%	-

Table A2. Comparison of validation metrics of AutoCNN against benchmarked models for NASDAQ index

Model		Metrics				
	MSE	RMSE	MAE	R^2	MAPE (%)	ET (second)
AutoCNN	**0.00001**	**0.004**	**0.003**	**0.999**	**0.53%**	**1.8**
CNN-LSTM	0.00003	0.006	0.003	0.997	0.54%	22.8
CNN-GRU	0.00004	0.006	0.004	0.997	0.67%	21.5
CNN-SRNN	0.00007	0.008	0.006	0.995	0.99%	8.4
ARIMA (Kumar *et al.*, 2022)	-	20.438	-	-	1.43%	-

Table A3. The stepwise multiple testing method applied to AutoCNN

Results generated by the stepwise multiple testing method		
	SSE index	**NASDAQ index**
AutoCNN vs SVR	2.03e-05	3.16e-05

Appendix B

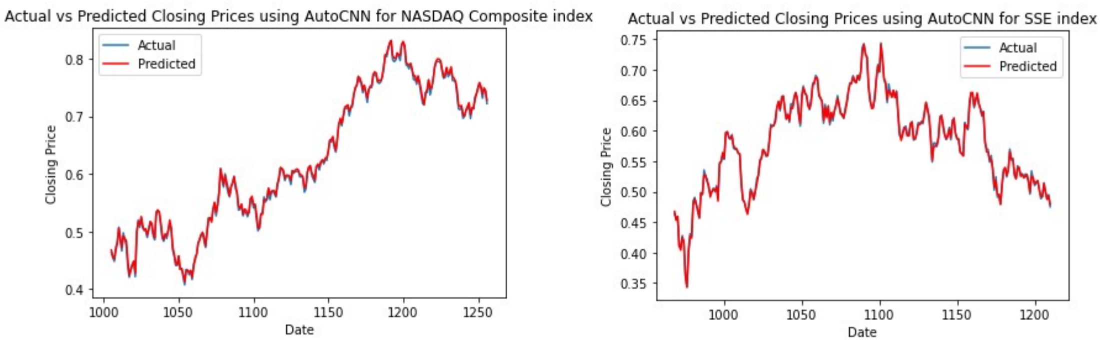

Figure B1. Real value vs Predicted value using the AutoCNN model

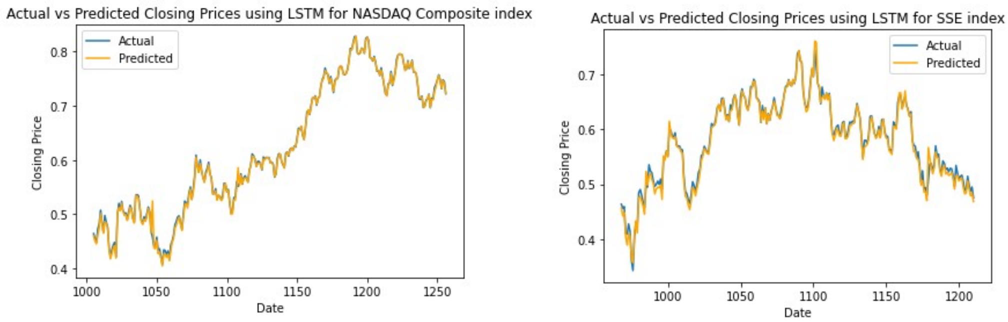

Figure B2. Real value vs Predicted value using the LSTM model

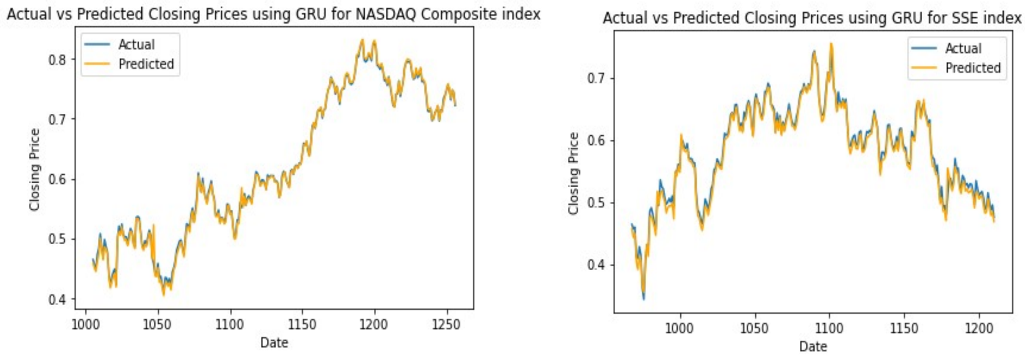

Figure B3. Real value vs Predicted value using the GRU model

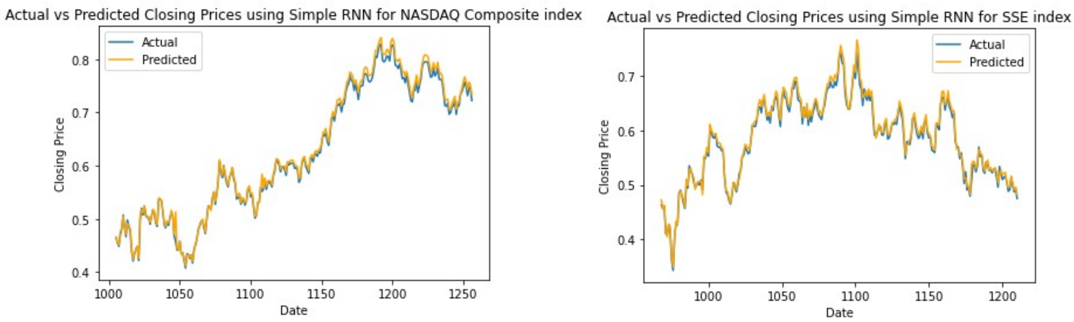

Figure B4. Real value vs Predicted value using the SRNN model

Appendix C

In Figures C1, C2, C3, C4 and C5, the y-axis represents the error rates generated by the models considered, while the x-axis denotes the name of each implemented model per dataset. However, for Figure C6, the y-axis represents the time consumed by each model during the training phase and the x-axis denotes the name of the model.

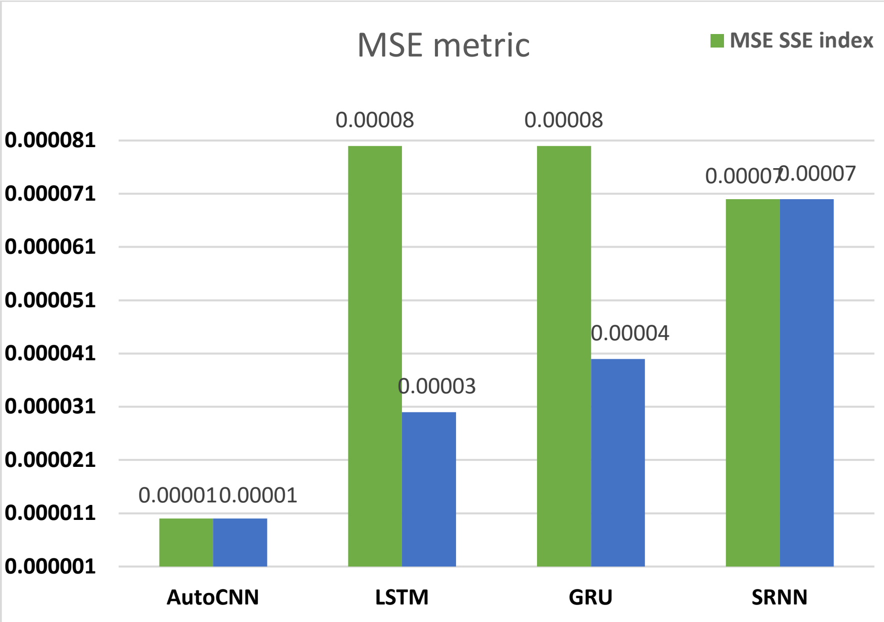

Figure C1. MSE comparison of the implemented models

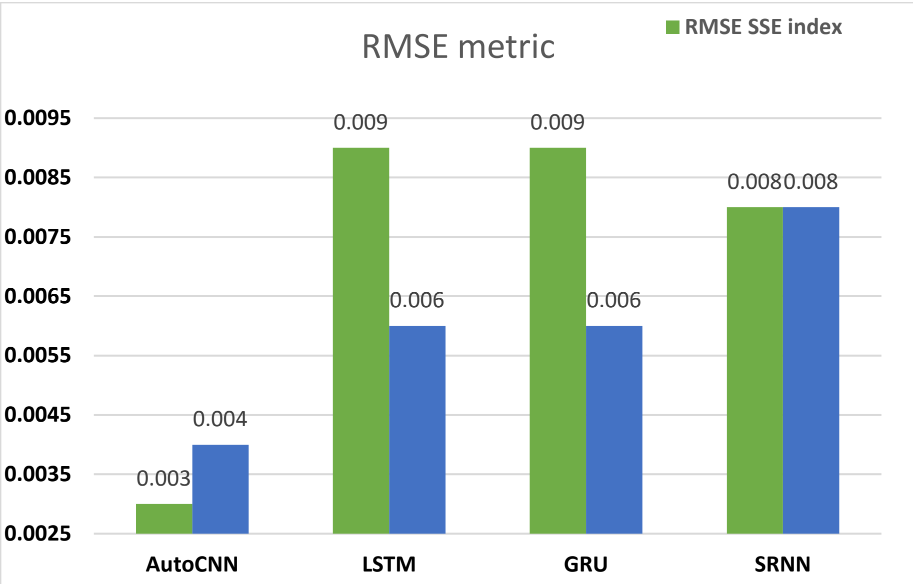

Figure C2. RMSE comparison of the implemented models

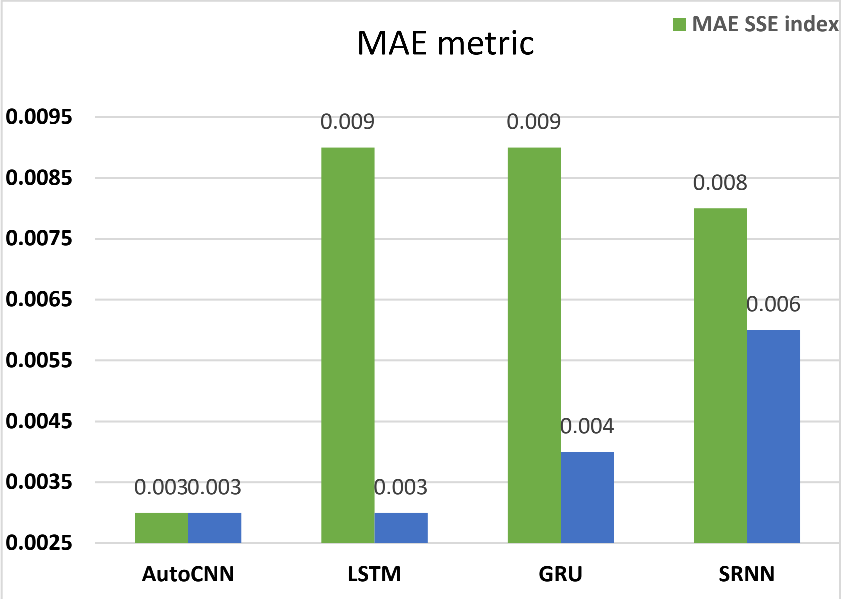

Figure C3. MAE comparison of the implemented models

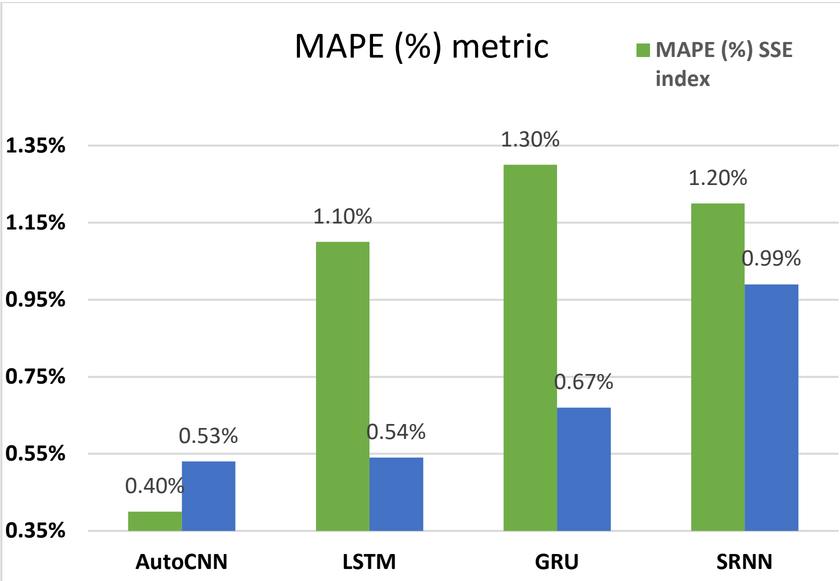

Figure C4. MAPE comparison of the implemented models

Copyright © 2024

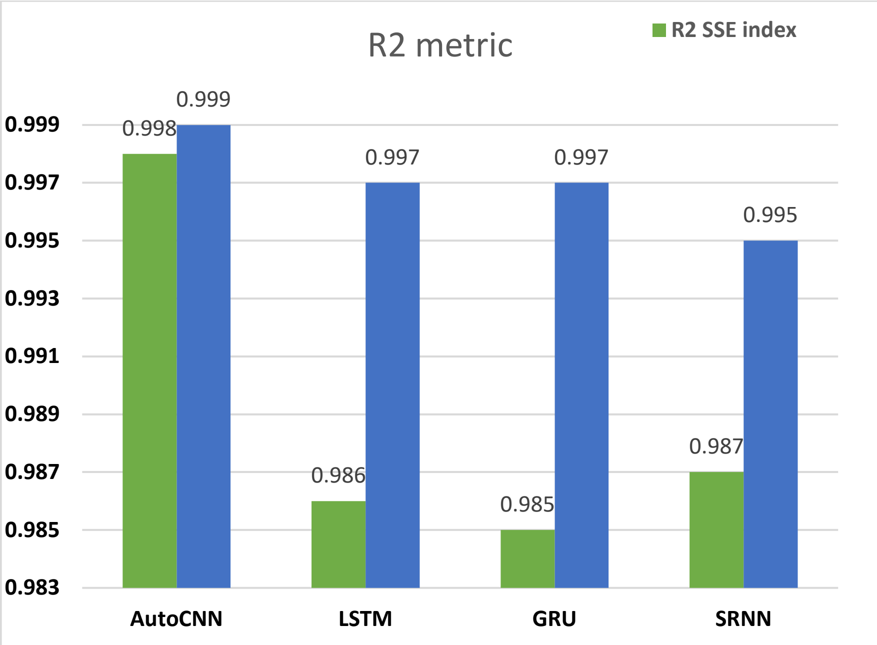

Figure C5. R^2 comparison of the implemented models

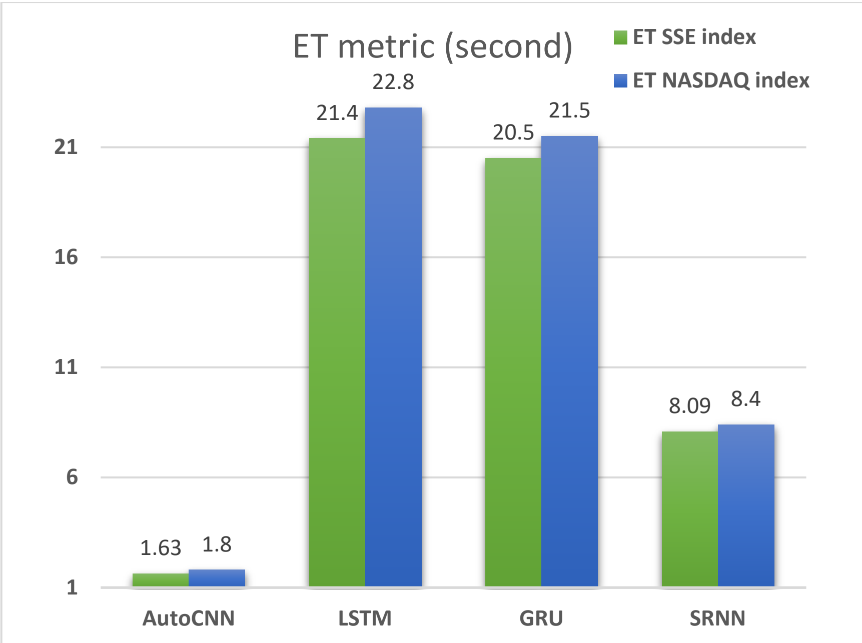

Figure C6. Execution Time comparison of the implemented models