International Journal of
Finance
(IJF)

Do Convolutional Neural Networks outperform Linear Models? Evidence from the Saudi Stock Market

International Journal of Finance
ISSN 2520-0852 (Online)
Vol. 11, Issue No. 1, pp. 14 - 35, 2026
www.carijournals.org

Do Convolutional Neural Networks outperform Linear Models? Evidence from the Saudi Stock Market

Eymen Errais[1]*, Farah Zouari[2], Montassar Ben Massoud[3]

[1]University of Tunis, Tunis Business School, Laboratoire de Recherche en Economie Quantitative du Développement (LAREQUAD).

https://orcid.org/0000-0003-4769-4137

[2,3]University of Tunis, Tunis Business School.

Accepted: 4th Dec, 2025, Received in Revised Form: 25th Dec, 2025, Published: 5th Jan, 2026

ABSTRACT

Purpose: The study investigates the effectiveness of artificial convolutional neural networks (CNNs) in predicting stock returns in the Saudi stock market and evaluates their predictive performance relative to traditional linear forecasting models commonly used in the financial literature.

Methodology: The research employs daily data from the Saudi Stock Exchange covering the period from January 2009 to August 2022 and includes a sample of 116 listed stocks. Both univariate and multivariate convolutional neural network models are developed and compared with benchmark linear models. Model performance is assessed using three standard forecasting accuracy measures: mean absolute error (MAE), mean squared error (MSE), and root mean squared error (RMSE).

Findings: The results indicate that convolutional neural network models significantly outperform linear models across all accuracy metrics. The superior performance of CNNs is observed consistently across the sampled firms, highlighting their robustness in capturing complex and nonlinear patterns in stock price dynamics.

Unique Contribution to Theory, Practice and Policy: The study contributes to the growing literature on machine learning applications in financial forecasting by providing empirical evidence on the superiority of deep learning models in an emerging market context. From a practical perspective, the findings support the adoption of advanced artificial intelligence techniques by investors and financial institutions. From a policy standpoint, the results underscore the potential of data-driven forecasting tools to enhance market efficiency in developing economies such as Saudi Arabia.

Keywords: *Stock Market Prediction, Capital Asset Pricing Model, 3-Factor Model, Convolutional Neural Network*

JEL Codes: *C3, C22, C32, C4*

International Journal of Finance
ISSN 2520-0852 (Online)
Vol. 11, Issue No. 1, pp. 14 - 35, 2026 www.carijournals.org

I. Introduction

A classic, yet difficult, challenge that has drawn the interest of both economists and computer scientists is stock market prediction. Investors and researchers have long been interested in learning how to understand the shifting regularity of the stock market and forecast the trajectory of stock prices. Politics, the economy, society, and the market all have an impact on how stock prices rise and fall. For publicly traded corporations, the stock price not only reflects the company's operating environment and aspirations for future growth, but it also serves as a crucial technical indicator for corporate analysis and research. Research on stock forecasting is crucial for understanding at what business cycle a nation is positioned. Therefore, there is significant theoretical value and a wide range of application possibilities for study on the intrinsic value and stock market forecasting. However, since the financial crisis of 2008 and the subsequent markets volatility, a lot of quantitative and traditional factor models have struggled to perform well and ceased to be profitable. Additionally, Finance is becoming more and more reliant on technological tools with each passing year. The rise of machine-readable data used to track, document, and convey financial system operations has substantial ramifications for how we approach modeling as a subject. Currently machine learning (ML) algorithms can offer a more rigorous methodology. Deep learning is particularly appealing as an alternative solution for current models and methodologies because of its ability to extract abstract features from data and find hidden nonlinear relationships without relying on econometric assumptions nor human experience. What makes these algorithms attractive to asset managers and financial specialists is their ability to self-adjust through a continuous process of trial and error.

The majority of these algorithms share two crucial characteristics:

1. Complex patterns and hidden correlations, such as non-linear and contextual relationships which are frequently challenging or impossible to find with linear analysis can be uncovered.

2. When multi-collinearity is present, ML algorithms are frequently more efficient than linear regression.

One of the relatively new techniques that has been extensively used for forecasting in a variety of fields is artificial neural networks (ANNs). The ANN application received a lot of attention in several academic fields, including finance. The first neural network model of a financial time series was presented by (White, 1988) for IBM, where he attempted to simulate a neural network model for understanding the nonlinear regularities in the dynamics of asset prices. Despite the work's limitations, it contributed to the body of evidence that disproves Efficient Market Hypothesis (EMH) (Malkiel, 1989). (Turban, 1996) examined the use of ANNs in various financial and investment fields.

According to (Qi, 1999) , neural networks enable predictions based on non-linear models to improve excess return prediction models by removing the linear structure from the model.

Many researchers used nonlinear methods, such as artificial neural networks (ANN), to carry out forecasting, like (Di Persio & Honchar, 2016) and (Kaastra & Boyd, 1996).

Deep Neural Networks (DNNs), which was first implemented in computer vision and natural language processing, is another powerful tool to predict complex financial time series. Multivariate time series analysis using deep learning models is demonstrated in (Batres-Estrada, 2015) and (Chong, Han, & Park, 2017) showed that time series in the stock market can be predicted by analyzing and processing sequential temporal data.

Convolutional Neural Networks (CNN) were initially developed by (Gardner & Dorling, 1998) and (LeCun, Bengio, & Hinton, 1995). Each of CNN's numerous layers serves a specific purpose, and they can be described as input, convolutional, pooling, fully connected, and output layers. (Dingli & Fourniner, 2017) used deep convolutional neural networks to make price predictions using historical data. Training data included past prices, technical indications, foreign exchange rates, global indexes, and data on a variety of commodities. Predictions can be made either weekly or monthly using one of two distinct datasets. The results did not provide evidence that CNNs are better than other classifiers like logistic regression and support vector machine. (Tsantekidis, et al., 2017) applied CNN to high-frequency data obtained from a limit order book (LOB) based on a predetermined set of restrictions to buy or sell a predetermined number of shares within a predetermined range of values. When compared to Support Vector Machines (SVM) and Multi-Layer Perceptron (MLP) models, the proposed model performed better. (Roman & Jameel, 1996) and (Heaton, Polson, & Witte, 2016) discussed multivariate financial time series modeling using deep learning architectures. (Qing, Leggio, & Schniederjans, 2005) compared the performance of univariate and multivariate linear models to the performance of neural networks in stock returns prediction for the Chinese stock market and found that neural networks outperform traditional financial models.

Artificial neural networks have been applied in numerous studies to make stock market forecasts. However, there has not been much research on the use of neural networks to forecast stock price movement in emerging markets, and especially the Middle East and North Africa (MENA) region. This could be because returns on developing markets are often more predictable than returns on developed markets and are more likely to be influenced by local information according to (Harvey, 1995). Still, (Alotaibi, et al., 2018) used real datasets from the Saudi Stock Exchange and historical oil prices to demonstrate the effectiveness of neural networks in predicting the Saudi stock exchange movements. However, there is still a very rare study on Saudi Arabian stock market forecasting.

To the best of our knowledge, we are the first to compare the prediction power of traditional linear models from the financial literature to non-linear convolutional neural networks in predicting stock returns of the Tadawul stock exchange. Will CNN be able to successfully take advantage of the anomalies and inefficiencies of emerging markets, and Saudi Arabia in particular, to offer a superior tool for investors to use in order to generate excess returns?

The paper is organized as follows. In section 2 we describe the dataset. In section 3 we present our research methodology for both linear and deep learning models. Section 4 presents the results and findings. Section 5 concludes.

II. Data

Tadawul Stock Market

Saudi Stock Exchange, also known as Tadawul, is currently the only stock exchange in the country and the main stock exchange among the Gulf Cooperation Council (GCC) countries, but that was not always the case.

Trading is the Saudi market started in 1954, and the market remained mostly informal up to the late 1970 with only 14 listed companies. However, in 1984 it started gaining some official legitimacy when a ministerial committee was charged by the government to regulate and monitor the market. The stock exchange continued evolving and becoming more significant, until reaching its next milestone which was the creation of Capital Market Authority (CMA) in 2004 that became its sole regulator. Finally, Tadwul was established in 2007 as a joint stock company and is the only entity in Saudi Arabia authorized to act as a securities exchange which was one of the primary reasons for the Saudi stock exchange to become the largest in the region. Another key factor in the success of this stock exchange was the agreement signed with Nasdaq at the end of 2017 to revamp the post-trade technological infrastructure at Tadawul. Before this accord, trading was restricted to. equities, Islamic bonds or sukuk, exchange-traded funds (ETFs), and mutual funds. However, the new partnership enabled the Saudi exchange to add new asset classes to the market, such as derivatives which was an important step in providing investors with a comprehensive and broad variety of investment products and services and further developing the Saudi capital market.

As of the start of 2022, the Saudi Stock Exchange has 210 listed companies on the main market with a market cap over 3 trillion U.S. dollars which makes it regarded as one of the largest exchanges not only regionally but also on the global scale. Another proof of the significance of Tadawul is its inclusion in the MSCI, FTSE Russell, and S&P Emerging Market indices.

Data Description

With the assumption that past information can be used to forecast the future returns of securities, we use historical market and fundamental data of 116 public corporations traded on the Saudi stock exchange, including the TADAWUL All Shares Index (TASI) for a 12-year period starting January 3rd, 2009, through August 8th, 2022. Table 1 below describes the dataset.

International Journal of Finance
ISSN 2520-0852 (Online)
Vol. 11, Issue No. 1, pp. 14 - 35, 2026
www.carijournals.org

We use Bloomberg to retrieve the stock market data and the official website of the Saudi Central Bank to manually retrieve the 3-month SAMA (Saudi Arabia Monetary Authority) treasury bills rates, that we will use later as the risk-free rate.

Table 1: Description of the market data.

Market Data	
Independent Variables	***Definition***
High and Low Price (HP & LP)	The maximum and minimum prices in a specific time period (in this case, daily) are referred to as the "high" and "low", respectively, in stock trading.
Open and Close Price (OP & CP)	The opening and closing prices of a stock are the same period (in this case, daily) at which trading started and ended.
Average Daily Traded Volume (ADTV)	The average daily volume of shares traded for a given stock is a technical indicator that investors utilize. By representing an expression of the general degree of interest in a stock, it is a crucial metric for determining investors' liquidity and price support or resistance levels over the course of a single trading day.
Market Capitalization (MCAP)	Size Factor: Market capitalization value is frequently used by the investment community. Investors utilize it to rate firms and assess their sizes relative to a certain sector or industry.
Risk Free Rate (Rf)	The Risk-Free Rate is the theoretical rate of return on assets with no risk. This rate is used to determine the minimum return needed on investments with more risk.
Market Return (Rm)	The Tadawul All Share Index (TASI) is a major stock market index which tracks the performance of all companies listed on the Saudi Stock Exchange.
Market Risk Premium	The additional return an investor will receive (or expects to receive) from holding a risky market portfolio instead of risk-free assets

Table 2: Description of the fundamental data.

Fundamental Data	
Features (Input)	***Definition***
Price/Earnings Ratio	Value Factor: Investors can use the P/E ratio to compare a stock's market value to its earnings. The P/E ratio, in essence, demonstrates the price the market is prepared to pay for a company at the present time based on expected future earnings. With a low P/E ratio, the price of a stock may be excessively low in relation to its earnings, i.e., it's undervalued. A high P/E ratio, on the other hand, would suggest that the current stock price is rather expensive in comparison to earnings, i.e., it's overvalued.
Price/Book Value Ratio	Value Factor: A company's market value and book value are compared using the Price/Book value where the book value is represented by the net assets of the firm.
Dividend Yield	The dividend yield is a financial indicator of a company's annual dividend payout in relation to its stock price. It measures the cash flow amount received for every dollar invested in a stock.
Net Assets	The value of a company's resources can be roughly estimated using the net asset indicator. The worth of a firm is typically inversely correlated with its net asset value.
Return on Equity (ROE)	A company's profitability and the effectiveness of its revenue generation are measure by its Return on Equity. A high ROE is an indicator of the effectiveness of the firm in generating profits from equity financing.
Profit Margin	One of the frequently employed profitability indicators is the profit margin. It determines how profitable a firm or a particular business activity is by showing the portion of sales that resulted into profits.

Before working on the models, we need to check the data for stationarity. Stationarity is a characteristic of time series data which states that the distributional properties (mean and standard deviation) of the data series have not altered through time. The assumption of stationarity is widespread across a wide variety of procedures and techniques used in time series research: estimating trends, making predictions, and drawing inferences about the causes of events are some examples of these.

For the purposes of forecasting, it is essential that the data is stationary since, in the absence of this condition, one is expecting the model to predict data that is completely unlike anything it has ever encountered in the past. The question now is, how can we verify its existence?

In autoregressive modeling, we can apply several different tests, such as the Kwiatkowski–Phillips–Schmidt–Shin (KPSS), the Phillips–Perron, and the Dickey-Fuller test. In this paper, we choose to follow the Dickey-Fuller test to test the stationarity of our data.

The Dickey-Fuller test is a straightforward AR model presented as follows:

$$y_t = \rho y_t + u_t$$

Where:

- y_t is our variable of interest at the time t
- ρ is a coefficient that defines the unit root
- u_t is noise or can be considered as an error term.

It is a statistical significance test, which indicates the results in the form of hypothesis testing using both null and alternative hypotheses. As a consequence of this, we will be in possession of a p-value, which will provide us with the basis upon which to draw conclusions regarding the time series and determine whether or not it is stationary.

We define:

- <u>The null hypothesis, H_0:</u> times series is a stationary.
- <u>The alternative hypothesis H_1:</u> times series is non-statinary.

⇨ The null hypothesis is accepted if the p-value associated with this test is greater than 5%. Alternatively, if the p-value is smaller than 5%, we can conclude that our data is stationary.

We get the below results by performing the Dicky- Fuller test on daily stock price returns to test the Hypothesis.

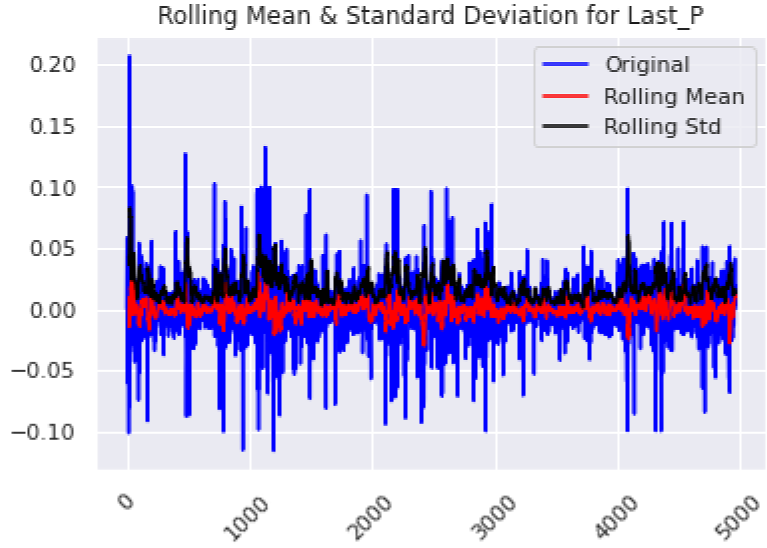

Figure 2: Rolling Mean & Standard Deviation for the Stock Returns of Stock AADC AB Equity

Table 3: Results of Dickey-Fuller Test

Test Statistic	-1,23E+01
P-value	**7,20E-23**
# Lags Used	2,20E+01

- ⇨ From the above results, we see that the percentage change in close prices is stationary as the p-value is much less than 5%.
- ⇨ For greater intuition, the graph above shows that the rolling mean and rolling standard deviation lines are almost flat, which means that they don't change in a way that is statistically significant over time.

III. The Models

We analyze the performance of five model specifications to test the fundamental research assumptions on forecasting stock price returns. In particular, we create linear and nonlinear

International Journal of Finance
ISSN 2520-0852 (Online)
Vol. 11, Issue No. 1, pp. 14 - 35, 2026

univariate (CAPM) and multivariate (3-factor and Multifactor) models. The five model categories are described below, and Table 3 provides a conceptual overview of each.

Let $(\Omega, \mathscr{F}, \mathbb{P})$ be the probability space defined by:

- The sample space Ω composed of all the public data available for the 212 listed corporations in the Stock Exchange of Saudi Arabia, Tadawul.
- The sigma algebra $\mathscr{F}$ as the smallest sigma algebra containing all the data described in table 1 and 2 that is provided for the listed equities available in the period starting from January 3rd 2009 to August 8th 2022
- The probability P defined by the risk-neutral probabilities of all possible future states of the stock returns occurring.

Table 4: Overview of the models used.

Probability Filtered Space	Linear Models	Non-Linear Models
$(\Omega, \mathscr{F}, F, \mathbb{P})$	UVL (CAPM)	UCNN
$(\Omega, \mathscr{F}, G, \mathbb{P})$	3F-L	3F-CNN
$(\Omega, \mathscr{F}, H, \mathbb{P})$		MV-CNN

In order to assess the proposed methodology, we use the data from the Tadawul Stock Market described above. The input data is divided into training and testing sets, with the first 70% of the data representing training and the remaining 30% representing testing. Several tests were carried out to optimize the forecasting models and improve their performance. The number of epochs, size of batches and other hyperparameters are all altered during the optimization process. Three performance measures are then utilized and assessed to compare the forecasting models and choose the model that accurately predicts the future stock returns of the Saudi equity market. The superiority of the prediction models is measured by how close its MAE, MSE and RMSE values are to zero.

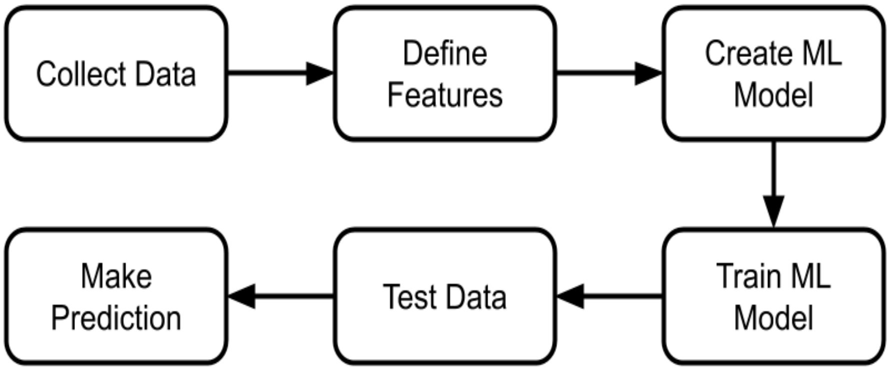

Figure 3: Proposed Model Block Diagram

Experimental Environment:

All experiments and models are implemented in Python. For various purposes, we relied on the following libraries:

• **Keras:** Keras is a well-known open-source framework that helps scientists and developers quickly define and train a large number of deep learning models. Its primary purpose is in the development and education of neural networks. It gives us an easy way to access libraries like Tensorflow, which in turn lets us take advantage of graphics processing units (GPUs) for quicker training and prediction. Keras allows for rapid testing, which is essential for gathering useful insights, which in turn leads to better deep learning models for stock prediction. To encourage the use of accuracy as the primary metric for CNNs trained on balanced datasets, the Keras team eliminated metrics including recall, precision, and fmeasure from version 2.0.

• **Tensorflow:** To conduct low-level operations like tensor products and convolutions, Keras relies on its backend, Tensorflow. Google is responsible for its creation and upkeep.

• **Matplotlib:** The actual time-series and forecast trends were plotted using Matplotlib.

• **Pandas:** The reading of data from csv files into DataFrames was accomplished with the help of Pandas.

• **Numpy:** Numpy was used to carry out matrix operations such as flip, reshape, and produce random matrices.

The input data was reconstructed into a group of subsequences, each consists of 30 time steps (30 working days) to forecast the price return of one day ahead in the future.

Linear Models

Univariate Linear Model – Single Factor Linear Model:

International Journal of Finance

ISSN 2520-0852 (Online)

Vol. 11, Issue No. 1, pp. 14 - 35, 2026 www.carijournals.org

What became known as the "capital asset pricing model" (CAPM) was created by (Sharpe, 1964), (Lintner, 1965), (Mossin, 1966), and (Treynor, 1962). This paradigm has enticing simplicity and insightful power. It showed that there is a cross-sectional relationship between expected returns and security risk for individual securities as assessed by beta, which is the covariance between the security return and the market return scaled by the market return variance. The CAPM model provided a new approach to predict the return of a risky asset by linking it directly to the risk that the asset bears. The risk premium which is correlated to the asset's beta is added to the risk-free rate to determine an asset's projected return. The realized returns of securities may be affected by a variety of risky events in this framework, but only beta risk is systematically priced. This model is built on the assumption that every risk or risky event; other than the risk captured by the beta, can be diversified away, hence they are not captured by the model, and they do not affect the expected return.

Let (Ω, $\mathscr{F}$, F, $\mathbb{P}$) be a filtered probability space endowed with the filtration F = $\{F_t : t \in T\}$, generated by the stochastic process $M = \{M_t : t \in T\}$ with M_t being the market risk premium at time t.

We analyze the stock market return predictability in the context of the capital asset pricing theory. We define the expected returns conditional on F as:

$$\mathrm{E}(\, \boldsymbol{R_{i,t}} \mid \boldsymbol{F_t}) = \boldsymbol{r_{i,t}} = \alpha + \beta \boldsymbol{x_{i,t-1}} + \varepsilon_t$$

where $r_{i,t}$ is the period-t return on stock(i), $x_{i,t}$ is the predictor variable, and εt is a zero-mean disturbance term. It is straightforward to use equation [1.1] to generate an out-of-sample forecast of r_{t+1} based on $x_{i,t}$ and data available through period t

Multivariate Linear Model – 3 Factor Linear Model:

The increasing number of questions surrounding the CAPM model and the lack of empirical evidence supporting it, called for a new asset pricing model and in 1992, Fama and French invented the three-factor model. In essence, Fama and French merged earlier studies on the size impact, the value effect, and the overall market component into one cross-sectional equation. The difference in return between the cap-weighted market portfolio and the risk-free rate of interest is represented by the MKT factor (Market Premium), the difference in returns between a portfolio of small-cap stocks and a portfolio of large-cap stocks is represented by the SML factor (Size) and the difference in returns between a portfolio with high book values of equity relative to market values of equity BE/ME and a portfolio with low BE/ ME is represented by the HML factor (Value). Contrary to the CAPM case, the three-factor model, despite not being formally grounded in a theory, this model was well supported by empirical data. This paradigm became the basic norm for most subsequent quantitative equity studies.

Let (Ω, $\mathscr{F}$, G, $\mathbb{P}$) be a filtered space endowed with the filtration G = $\{G_t$: t ∈ T}, generated by the 3 stochastic processes:

$M = \{M_t : t \in T\}$, with Mt being the market premium at time t.

$MC = \{MC_t : t \in T\}$ with MC being the market capitalization of the stock at time t.

$PB = \{PB_t : t \in T\}$ with PB being the price to book ratio of the stock at time t.

We analyze the stock market return predictability in the context of Fama and French's 3-factor model. We define the expected returns conditional on G as:

$$E(R_{i,t} \mid G_t) = r_{i,t} = \alpha + \beta_i * M_{t-1} + \beta 2 * Pb_{t-1} + \beta * M\rho_{t-1} + \varepsilon_t$$

Shortcomings of linear regression

Table 5: Disadvantages of linear regression in time series modelling

Properties of Time Series Data	Meaning	Can OLS Deal With This	Why / Why Not?
Sequentially Ordered	Outcome Today Depends On Outcome Yesterday	No	• Order Of The Data Does Not Matter • OLS requires No Correlation between observations of the error term (no autocorrelation)
Displays Trends	Different Data Series May Show High Correlation Even If their Non-Trend Component Is Not Correlated (i.e. Spurious Correlation) Mean of The Data Changes Over Time	No	• OLS requires No Perfect Correlation Between Independent Variables (No Multicolinearrity)
May Not Be Stationary	Mean Of The Data Changes Over Time	No	• OLS requires That The Variance Of Error Does Not Change For Each Observation/A range Of Observations (No Heteroskedasticity)

1D-Convolutional Neural Networks .(CNN):

Time series forecasting is difficult. Unlike the simpler problems of classification and regression, time series problems add the complexity of order or temporal dependence between observations. This can be difficult as specialized handling of the data is required when fitting and evaluating models. This temporal structure can also aid in modeling, providing additional structure like trends and seasonality that can be leveraged to improve model skill. Traditionally, time series forecasting has been dominated by linear methods like linear regression and ARIMA because they are well understood and effective on many problems. But these classical methods also suffer from some limitations, such as:

- Focus on complete data: missing or corrupt data is generally unsupported.
- Focus on linear relationships: assuming a linear relationship excludes more complex joint distributions.
- Focus on fixed temporal dependence: the relationship between observations at different times, and in turn the number of lag observations provided as input, must be diagnosed and specified.
- Focus on univariate data: many real-world problems have multiple input variables.
- Focus on one-step forecasts: many real-world problems require forecasts with a long-time horizon.

Machine learning methods can be effective on more complex time series forecasting problems with multiple input variables, complex nonlinear relationships, and missing data. In order to perform well, these methods often require hand-engineered features prepared by either domain experts or practitioners with a background in signal processing simpler neural networks such as the Multilayer Perceptron or MLP approximate a mapping function from input variables to output variables. This general capability is valuable for time series for a number of reasons.

- Robust to Noise: Neural networks are robust to noise in input data and in the mapping function and can even support learning and prediction in the presence of missing values.
- Nonlinear: Neural networks do not make strong assumptions about the mapping function and readily learn linear and nonlinear relationships.

More specifically, neural networks can be configured to support an arbitrary defined but fixed number of inputs and outputs in the mapping function. This means that neural networks can directly support:

- Multivariate Inputs. An arbitrary number of input features can be specified, providing direct support for multivariate forecasting.

We consider the 1-dimensional convolutional neural network as a potential solution to the shortcomings of the linear regression model in the prediction of stock market returns. OLS and linear models in general depend on hand-picked features that are manually and statically selected, however, CNNs can learn from unprocessed time series data with little to no need for human intervention in the form of feature extraction and engineering, while preserving the temporal nature of the data.

Table 6 compares key characteristics of linear regression and CNN models in the prediction of time series data.

Table 6: Comparison of OLS and CNN in Time Series Forecasting

Nature Of Time Series Data	OLS	CNN
Displays Trends	No	Yes
The Data May Not Be Stationary	No	Yes
Requires No.Of obs. To Be Less Than No.Of Independent Variables	No	Yes
Linear	Yes	No
Temporal Structure Of The Data	No	Yes
New Features Created	No	Yes

Definition of Convolutional Neural Networks

CNNs, or Convolutional Neural Networks, are a special type of neural network that has been developed for processing images. By obtaining state-of-the-art results on tasks like image classification and by offering a component in hybrid models for completely new problems like object geolocation, picture interpretation, and more, they have proven effective on difficult computer vision challenges.

They accomplish this by relying on raw data, such as raw pixel values, rather than domain-specific or constructed features extracted from raw data. Next, the model is taught how to automatically extract, from the raw data, the features that are immediately beneficial for the problem that is being addressed.

CNNs' capacity to learn and automatically extract characteristics from raw input data can be used to solve time series forecasting issues. A CNN model is able to read a series of observations, reduce them to their most important components, and then treat the observations as if they were a single-dimensional image.

CNN is comprised of a number of layers, which may be broken down into the following: the input layer, the convolutional layer, the pooling layer, the fully connected layer, and the output layer.

Data Preparation

A univariate or multivariate series needs to be prepared before it can be modeled. When fed a series of previous observations, the CNN model will learn to employ that data to predict the next observation. In order for the model to learn, the observational sequence must be converted into a set of independent examples. It is necessary to transform a time series into samples that include both input and output variables. The transformation tells the model what to learn and how we plan to use the model to make predictions in the future, such as what is needed to make a prediction (X) and what prediction is made (Y). The observations from earlier time steps, known as lag observations, are fed into a time series problem with a focus on making predictions for the current time step.

All CNN layers require a multidimensional input to operate effectively. These inputs have three dimensions:

- The Samples: We can think of one sample as one sequence. The term "batch" refers to a collection of samples.
- Time Steps: In statistical terms, one observation in the sample corresponds to one time step. Multiple time steps make up a single sample.
- Features: At each time step, one observation is counted as one feature. There are one or more features in a time step.

In our research, the input data was reconstructed into a group of subsequences, each consists of 30 timesteps.

Input Layer

In our study, only the input layer is updated between the univariate and multivariate models.

Univariate CNN

We are working with a univariate series, which means there is only one variable and one feature: which is the market risk premium. Let (Ω, $\mathscr{F}$, F, $\mathbb{P}$) be a filtered probability space endowed with the filtration $F = \{F_t : t \in T\}$, generated by the stochastic process $M = \{M_t : t \in T\}$ with M_t being the market risk premium at time t.

Our goal is to predict the expected stock returns conditional on F, defined by:

E $(\boldsymbol{R_{i,t}} \mid \boldsymbol{F_t}) = \hat{y} = f(X)$

where $R_{i,t}$ is the period-t return on stock(i), $\hat{y}$ is the predicted output variable, X the input component and f the transformation function.

Multivariate CNN

For the 3-factor and multivariate CNN, we perform the same approach described in the previous section, while changing the input variables.

3- Factor CNN:

Let (Ω, $\mathscr{F}$, G, $\mathbb{P}$) be a filtered space endowed with the filtration G = $\{G_t$: t ∈ T}, generated by the 3 stochastic processes: M, MC and PB.

The expected returns that follow the 3-factor CNN model are defined by the following equation:

E $(\boldsymbol{R_{i,t}} \mid \boldsymbol{G_t}) = \hat{y} = f(X)$

Multivariate CNN:

Let (Ω, $\mathscr{F}$, H, $\mathbb{P}$) be a filtered space endowed with the filtration H = $\{H_t$: t ∈ T$\}$ defined on the measurable space (Ω, $\mathscr{F}$) with $H_t = G_t \vee L_t$

$$G_t = \sigma(M_t,\ MC_t, PB_t,\ t \in T\ )$$

$$L_t = \sigma(OP_t, HP_t, LP_t, CP_t, V_t, ROE_t, PM_t, NA_t, DY_t,\ \mathrm{t} \in \mathrm{T})$$

This means that L_t is generated by the 9 stochastic processes:

- $OP = \{OP_t : t \in T\}$, with OP_t being the open price of a certain stock at time t.
- $HP = \{HP_t : t \in T\}$, with HP_t being the high price of a certain stock at time t.
- $LP = \{LP_t : t \in T\}$, with LP_t being the low price of a certain stock at time t.
- $CP = \{CP_t : t \in T\}$, with CP_t being the close price of a certain stock at time t.
- $V = \{V_t : t \in T\}$, with V_t being the average daily traded volume of a certain stock at time t.
- $ROE = \{ROE_t : t \in T\}$, with ROE_t being the return on equity of a certain stock at time t.
- $PM = \{PM_t : t \in T\}$, with PM being the profit margin at time t.
- $NA = \{NA_t : t \in T\}$, with NA_t being the net assets of the traded company at time t.
- $DY = \{DY_t : t \in T\}$, with DY_t being the dividend yield at time t.

The model seeks to predict the expected returns conditional on H:

E ($\boldsymbol{R_{i,t}}$ | $\boldsymbol{H_t}$) = $\hat{y}$ = $f(X)$

Convolutional layer

The data convolution operation should be carried out by the convolutional layer. As a matter of fact, the input is a function, the filter is another function, and the convolution operation is a technique for quantifying the effects of applying a filter to an input. A filter's size is proportional to the area it can cover. Convolutional operations are performed by each filter using the same set of weights. During the training process, the weights will be adjusted.

Assume that FxF convolutional filters are utilized, and the input of layer is a NxN matrix. Then, we plug those numbers into the below equation to get layer l's input. A filter is applied to the input data to obtain the output value of $v1,1$. This value is then used in the following layer. Typically, an activation function is applied to the output of each filter before it is fed into the next layer. We use the non-linear activation function ReLu.

$$\hat{v}\,^{l}_{i,j} = \delta(\sum_{k=0}^{F-1}\sum_{m=0}^{F-1} w_{k,m} v^{l-1}_{i+k,j+m})$$

where $\hat{v}^{l}_{i,j}$ is the value at the ith row and jth column of layer l, $w_{k,m}$ is the weight at the kth row and column m in the filter and δ is the activation function.

Pooling Layer

The data is subsampled at the pooling layer. This procedure not only alleviates the strain on the computer during training, but it also helps with the overfitting issue that plagues convolutional neural networks. To overfit means that a trained model is overly tailored to the training data, to the point where it is unable to generalize to new, unexpected data. It is related to the size of the training set and the number of parameters used in the prediction model. Multiple parameters are common in deep models, such as CNNs. Overfitting occurs more frequently in deep models because of this. Overfitting can be avoided with the help of several proposed strategies. Overfitting can be mitigated by using pooling layers in CNNs. One value is generated from the concatenation of all the values contained inside a pooling window. The risk of overfitting is reduced since fewer parameters need to be learned by the model when the input to subsequent layers is shrunk via this change. We choose the most popular form of pooling, called "max pooling," in our research. It involves selecting the maximum value within a given time interval.

Fully Connected Layer

The MLP network used in the last layer of a CNN is referred to as the fully connected layer. Its job is to take all the features that were extracted in the preceding layers and turn them into a usable form for the output. The equation below describes the relationship between two consecutive layers:

$$\hat{v}^{j}_{i} = \delta(\sum_{k} v^{j-1}_{k} w^{j-1}_{k,i})$$

Where $\hat{v}^{j}_{i}$ represents the value of neuron i at layer j, δ represents the activation function, and $w^{j-1}_{k,i}$ represents the weight of connection between neuron k at layer j and neuron i at layer j.

Dropout

We have also employed a technique termed dropout, which was originally designed for use in the training of deep neural networks, in addition to pooling. The dropout strategy was developed so that a model wouldn't over-learn its training data. Therefore, each neuron in the network has a probability, proportional to the dropout rate, of not being trained in a given training cycle. This keeps the model from being very flexible, which aids the learning process in finding a solution that is not overly specific to the training data and can generalize well for predicting future data that has not been labeled.

International Journal of Finance
ISSN 2520-0852 (Online)
Vol. 11, Issue No. 1, pp. 14 - 35, 2026 www.carijournals.org

Model Accuracy Evaluation

The performance of the designed models for predicting the stock market will be measured against a variety of parameters. These metrics are used to evaluate the accuracy with which generated models predict experimental results. These include:

Mean squared error (MSE): $\text{MSE} = \frac{\sum(Y_t - \hat{Y}_t)^2}{n}$

Where $\hat{Y}_t$ is the forecasted value of stock price returns at time t, Y_t the actual value of stock price returns at time t and n: the number of observations. This metric determines the average squared deviation between the values that were observed and those that were forecasted.

Root Mean Square Error (RMSE): $\text{RMSE} = \sqrt{MSE} = \sqrt{\frac{\sum(Y_t - \hat{Y}_t)^2}{n}}$

The Root Mean Square Error provides the square root of the average squared difference between the values predicted for a dataset and the values that were actually observed.

Mean Absolute Error (MAE): $\text{MAE} = \frac{\sum|Y_t - \hat{Y}_t|}{n}$

Without taking into account which direction the errors are going; the MAE calculates the average magnitude of the errors found in a series of projections.

IV. Results & Findings

Table reports the averages of the three prediction measures for all five forecasting models of the stock returns of 116 public companies listed on the Saudi stock exchange between the period starting 3rd January 2009 to 8th August 2022.

Table 7: Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) for the 5 proposed models

	MAE	MSE	RMSE
UVL	0,01316	0,000466	0,02115
3F-L	0,01302	0,000461	0,021
UV-CNN	0,01297	0,000395	0,01934
3F - CNN	0,01290	0,000384	0,01932
MV-CNN	0,012768	0,000370	0,01938

Comparison of Linear Models and CNN models

We find that the univariate CNN models have lower MAE, MSE and RMSE values than the univariate linear models. Accordingly, we draw the conclusion that the univariate CNN nonlinear models are superior to the univariate linear models.

International Journal of Finance
ISSN 2520-0852 (Online)
Vol. 11, Issue No. 1, pp. 14 - 35, 2026 www.carijournals.org

When evaluating the 3-factor linear and CNN models using the same three metrics, we find that they provide quite similar outcomes. The 3-factor model built with a convolutional neural network approach has better mean absolute error, mean squared error and root mean squared error values than the 3-factor linear model. Therefore, we draw the conclusion that 3-factor CNN models are more accurate at making predictions than 3-factor linear models.

Comparison of Univariate Linear Model and 3-factor Linear Model

We find that all three prediction accuracy measures (MAE, MSE and RMSE) of the 3-factor model are lower than the univariate linear model. This means, and we concur, that switching from the CAPM to the 3-factor model does make forecasts more accurate as the addition of other financial variables improves the accuracy of the forecast.

Comparison between Non-Linear Univariate (U-CNN) and Non-Linear 3-factor model (3F-CNN)

The comparison of the univariate and multivariate models when the convolutional neural network approach is used yields similar results. We find that all three prediction accuracy measures (MAE, MSE and RMSE) of the 3-factor non-linear model are lower than the univariate non-linear model. This means that the addition of other financial variables improves the accuracy of the forecast whether in linear modeling or non-linear modeling.

- ⇨ The models are consistent: the 3-factor model is preferred for stock market returns in the Tadawul stock exchange

Comparison of Non-Linear Univariate (U-CNN) and Non-Linear 3-factor model (3F-CNN) with the multivariate non-linear model (MV-CNN)

The multivariate CNN model outperforms both univariate models and 3-factor models. This shouldn't come as a surprise as deep learning works best with big datasets through its ability to uncover hidden patterns and relationships that will enable investors to exploit inefficiencies in the stock market.

V. Conclusion

This study examines the predictive performance of linear and nonlinear modeling approaches for forecasting stock returns in the Saudi equity market. Specifically, it compares univariate and multivariate linear models rooted in traditional asset-pricing theory with convolutional neural network (CNN) architectures that are capable of capturing complex temporal and nonlinear dynamics. Using daily data for 116 listed equities over the period 2009–2022, model performance is evaluated through multiple accuracy metrics to ensure robustness of the findings.

The empirical results provide clear and consistent evidence that convolutional neural networks outperform traditional linear models in predicting stock returns in the Saudi stock market. Across all specifications, CNN-based models exhibit lower forecasting errors, as measured by mean absolute error, mean squared error, and root mean squared error. This superior performance can be attributed to the ability of CNNs to automatically extract informative features from raw time-series data and to model both linear and nonlinear relationships without imposing restrictive parametric assumptions. Unlike linear regression frameworks, CNNs effectively exploit the temporal structure of financial data and adapt to changing market dynamics.

The findings further indicate that multivariate models deliver more accurate forecasts than univariate specifications, regardless of whether linear or nonlinear techniques are employed. The inclusion of additional market and firm-level information enhances predictive power, highlighting the importance of information richness in stock return forecasting. Among all models considered, the multivariate convolutional neural network achieves the highest predictive accuracy, underscoring the value of deep learning methods when applied to high-dimensional financial datasets.

From a broader perspective, the results suggest that deep learning approaches are particularly well suited for emerging markets such as Saudi Arabia, where market inefficiencies, structural changes, and nonlinear dynamics are more pronounced than in developed markets. The ability of CNNs to uncover hidden patterns in noisy financial environments provides investors and practitioners with a powerful tool for improving return forecasts and potentially enhancing investment decision-making.

Future research could extend this framework by incorporating additional sources of information, such as trading volume, macroeconomic indicators, commodity prices, or alternative data, to further improve predictive performance. Moreover, exploring other deep learning architectures and evaluating their economic significance in portfolio construction and risk management contexts would offer valuable avenues for continued investigation. Overall, the study reinforces the growing role of deep learning techniques as a central component of modern financial forecasting and empirical asset pricing.

References

Alfreedi, A. A. (2018). Shocks and Volatility Spillover Between Stock Markets of Developed Countries and GCC Stock Markets. *Journal of Taibah University for Science, 13*(1), 10.

Alotaibi, T., Nazir, A., Alrobaea, R., Alotibi, M., Alsubeai, F., Alghaldi, A., & Alsulimani, T. (2018). Saudi Arabia Stock Market Prediction Using Neural Network. *International Journal on Computer Science and Engineering, 9*(2), 62-70.

Banz, R. W. (1981). The Relationship between Return and Market Value . *Journal of Financial Economics , 9*(1), 3-18.

Basu, S. (1977). Investment Performance of Common Stocks in Relation to Their Price-Earnings Ratios: A Test of the Efficient Market Hypothesis. *Journal of Finace , 32*(3), 663-82.

Batres-Estrada, G. (2015). *Deep Learning for Multivariate Time Series.* Technical Report, Stockholm.

Browlee, J. (2018). *Deep Learning for Time Series Forecasting.* Machine Mastery.

Carhart, M. (1997). On Persistence in Mutual Fund Performance . *Journal of Finance , 52*(1), 57-82.

Chong, E., Han, C., & Park, F. C. (2017). Deep Learning Networks for Stock Market Analysis and Prediction: Methodology, Data Representations and Case Studies. *Expert Systems with Applications, 83*, 187-205.

Connor, G. (1995). The Three Types of Factor Models:A Comparison of Their Explanatory Power . *Financial Analysts Journal , 51*(3), 42-46.

Daniel, K., & Moskowitz., T. J. (2016). Momentum Crashes. *Journal of Financial Economics, 122*(2), 221-47.

Di Persio, L., & Honchar, O. (2016). Artificial Neural Networks Architectures for Stock Price Prediction: Comparisons and Applications. *International J. Circuits, Systems and Signal Processing, 10*, 403-413.

Dingli, A., & Fourniner, K. S. (2017). Financial Series Forecasting - A Deep Leaarning Approach. *International J. Machine Learning and Computing, 7*(5), 118-122.

Fama, E. F., & French, K. R. (1989). Business Conditions and Expected Returns on Stocks and Bonds. *Journal of Financial Economics, 25*(1), 23-49.

Fama, E. F., & MacBeth, J. D. (1973). Risk, Return, and Equilibrium:Empirical Tests. *Journal of Political Economy, 81*(3), 607-36.

Fisher, L., & Lorie, J. H. (1964). Rates of Return on Investments in Common Stocks. *Journal of Business, 37*(1), 1-21.

Gardner, M., & Dorling, S. (1998). Artificial Neural Networks (the Multilayer Perceptron) a Review of Applications in the Atmospheric Sciences. *Atmospheric Environment , 32*(14-15), 2627-2636.

Harvey, C. (1995). Predictable Risk and Returns in Emerging Markets. *The Review of Financial Studies, 8*, 773-816.

Heaton, J., Polson, N., & Witte, J. (2016). Deep Learning in Finance.

Jegadeesh, N., & Titman, S. (1993). Returns to Buying Winners and Selling Losers: Implications for Stock Market Efficiency. *Journal of Finance, 48*(1), 65-91.

Jia, H. (2016). Investigation into the Effectiveness of Long Short Term Memory Networks for Price Stock Prediction.

Kaastra, I., & Boyd, M. (1996). Designing a Neural Network for Forecasting Financial and Economic Times Series. *Neurocomputing, 10*(3), 215-236.

LeCun, Y., Bengio, Y., & Hinton, G. (1995). Convolutional Neural Networks for Images, Speech, and Times Series. *The Handbook of Brain Theory and Neural Networks, 3361*(10).

Lintner, J. (1965). The Valuation of Risk Assets and the Selection of Risky Investments in Stock Portfolios and Capital Budgets. *Review of Economics and Statistics, 47*(1), 13-37.

Malkiel, B. G. (1989). Efficient market hypothesis. *The New Palgrave: Finance.*, 127-134.

Markowitz, H. (1952). Portfolio Selection. *Journal of Finance, 7*(1), 77-91.

Mossin, J. (1966). Equilibrium in a Capital Asset Market. *Econometrica, 34*(4), 768-83.

Novy-Marx, R. 2. (2013). The Other Side of Value: The Gross Profitability Premium. *Journal of Financial Economics , 108*(1), 1-28.

Qi, M. (1999). Nonlinear Predictability of Stock Returns Using Financial and Economic Variables. *17,* 419-429.

Qing, C., Leggio, B. K., & Schniederjans, M. J. (2005). A Comparison between Fama and French's Model and Artificial Neural Networks in Predicting the Chinese Stock Market. *Computers and Operations Research, 32*(10), 2499-2512. doi:10.1016/j.cor.2004.03.015

Reinganum, M. B. (2011). Active Equity Portfolio Strategies:Dynamic Quantitative Models. 161-179.

Reinganum, M. R. (1981a). Misspecification of Capital Asset Pricing:Empirical Anomalies Based on Earnings'Yields and Market Values . *Journal of Financial Economics, 9*(1), 19-46.

Roman, J., & Jameel, A. (1996). Backpropagation and Recurrent Neural Networks in Financial Analysis of Multiple Stock Market Returns. *Proceedings of the Twenty-Ninth Hawaii International Conference. 2,* pp. 454-460. IEEE.

Rosenberg, B., Reid, K., & Lanstein., R. (1985). Persuasive Evidence of Market Inefficiency. *Journal of Portfolio Management, 11*(3), 9-16.

Sharpe, W. (1964). Capital Assert Prices. *Journal Of Finance, 19*(3), 425-42.

Treynor, J. (1962). Toward a Theory of Market Value of Risky Assets . *Asset Pricing and Portfolio Performance* , 15-22.

Tsantekidis, A., Passalis, N., Tefas, A., Kanniainen, J., Gabbouj , M., & Iosifidis, A. (2017). Forecasting Stock Prices from the Limit Order Book using Convolutional Neural Networks. *IEEE 19th Conf. Business Informatics.*

Turban, R. R. (1996). Neural networks in finance and investing. Using artificial intelligence to improve realworld performance.

White, H. (1988). *Economic Prediction Using Neural Networks: The Case of IBM Daily Stock Returns.* University of California San Diego, Department of Economics.