

Article

From Forecasting Accuracy to Trading Profitability: Evaluating Sequence Models for Stock Price Prediction

Carol Anne Hargreaves *  and Hieu Le Trung

Department of Statistics and Data Science, National University of Singapore, Singapore 117577, Singapore

* Correspondence: carol.hargreaves1@gmail.com

Abstract

Accurate stock price forecasting remains a challenging problem due to the noisy, nonlinear, and non-stationary characteristics of financial time series. Although recent advances in deep learning have improved predictive capabilities, most prior studies evaluate forecasting models primarily using statistical error metrics, with limited consideration of their practical value in trading and investment decision-making. This creates a gap between predictive performance and real-world economic utility. This study proposes a decision-oriented evaluation framework for multi-step stock price forecasting that jointly assesses predictive accuracy and trading profitability within a unified experimental setting. Using data from 91 ASX 100 stocks after data cleaning, with a testing period spanning 2019–2020, several deep learning architectures, including Multi-Layer Perceptron (MLP), Gated Recurrent Unit (GRU), Seq2Seq, and attention-based sequence models, are systematically compared under identical training and trading conditions. The results show that the Seq2Seq model achieved the best overall performance, obtaining the lowest average MAPE of 0.0293 and the highest ROI of 23.2%, while the attention-based model achieved a similar MAPE of 0.0294 but a lower ROI of 12.4%. Although differences in forecasting accuracy were relatively small, the Seq2Seq model achieved the highest observed trading profitability and generated a higher observed return than a passive market benchmark under the proposed evaluation framework. These findings suggest that evaluation based solely on prediction accuracy may not fully capture the practical value of forecasting models.

Keywords: financial time series forecasting; stock price prediction; decision-oriented evaluation; trading profitability; deep learning; sequence models; model evaluation framework

1. Introduction

1.1. Overview

Stock price forecasting is extremely challenging due to the volatility and nonlinear nature of stock price movements. The identification of high performing stock price forecasting models holds considerable value, as they enable high trading profits. To date, many researchers have studied the performance of a wide range of forecasting models to predict stock price movements. Among recent studies, deep learning models have shown good model performance results. Specifically, Long Short-Term Memory (LSTM) models [1] and the Gated Recurrent Unit (GRU) [2] have gained recognition for their ability to model temporal dependencies and to capture long-term patterns in sequential stock data.

Some new techniques such as the Sequence-to-Sequence (Seq2Seq) model [3] and attention mechanism [4] have achieved great success in language modeling tasks such as



Academic Editor: Steven Prestwich

Received: 7 May 2026

Revised: 8 July 2026

Accepted: 14 July 2026

Published: 18 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

machine translation and text generation. However, these architectures are still not very popular in time series research, especially in stock price prediction.

Recent studies in stock price prediction focus on a model and its accuracy, making it extremely difficult to compare model performance based on different deep learning architectures under the same experimental conditions. There are very few studies that have evaluated the performance of sequence-to-sequence models within the Australian stock market context.

This study focuses on the top 100 stocks in the Australian Stock Exchange and the performance of four deep learning architectures based on accuracy and trading profitability. A consistent dataset, evaluation framework, and trading strategy were applied to the MLP, GRU, Seq2Seq, and Seq2Seq with attention models. In this experimental study, both stock price forecasting accuracy and trading profitability were used to compare the four models. This study contributes greatly to the financial time series forecasting research area and confirms the use of sequence-based models and sequence-based models with the attention mechanism for real-world stock price forecasting accuracy and trading profitability. In addition, this study shifts the focus from developing increasingly complex models to understanding how existing models perform under realistic decision-making conditions, thereby complementing current state-of-the-art research.

1.2. Related Works

Traditionally, most stock price prediction models use linear models. While these methods offer interpretability, they usually do not perform well in predicting stock prices. Ref. [5] examined a wide range of linear models using popular predictors from existing literature and found that most linear models have poor out-of-sample forecasting performance. In contrast, advancement in deep learning has introduced many new neural network architectures that can capture nonlinear patterns in time series data without any prior assumptions about the data, making them appropriate for capturing the volatility of stock markets.

The simplest type of deep learning architecture is a Multi-Layer Perceptron (MLP), which consists of an input layer, at least one hidden layer, and an output layer. Each layer contains neurons that are calculated by a weighted linear combination of the outputs from the previous layer. Next, the neurons transform and use a nonlinear activation function, which helps the MLPs to capture complex nonlinear patterns in the data, making them appropriate for tasks where linear models fall short. MLPs have been applied to financial time series and have achieved positive results [6,7]. Hybrid approaches where MLP combine with traditional statistical models such as Auto-Regressive Integrated Moving Average (ARIMA) have been recommended to improve model performance [8].

Some networks, such as Recurrent Neural Networks (RNNs), are designed particularly for capturing temporal dynamic behaviors in sequential data. RNNs process each time step of a time series while retaining information from the previous steps in a hidden state, allowing the network to model time-dependent relationships.

However, standard RNNs face challenges in learning long-term dependencies due to issues like vanishing and exploding gradients. To address these limitations, variants of RNN such as Long Short-Term Memory (LSTM) [1] and Gated Recurrent Unit (GRU) [2] significantly improve the model's performance due to their ability to capture and learn longer sequences. Due to the advantages of LSTM and GRU models being designed for sequential data, they often perform better than MLP models in stock price prediction [9–11].

Ref. [3] used two LSTM or GRU networks as an encoder and decoder to effectively capture sequential input–output patterns. The encoder processes the input sequence while the decoder generates the output sequence. With a separate decoder network, each output

can be generated based on the context of the previous ones. Many machine translation and text generation models use the Seq2Seq architecture. For example, ref. [12] used Seq2Seq models to predict dynamic output sequences with varying lengths in financial time series prediction. Additionally, ref. [13] used dual information of news and time series market data in a Seq2Seq model to improve the forecasting performance.

A long-time series forecasting method by utilizing multi-scale feature extraction and a sequence-to-sequence (seq2seq) attention mechanism in the hidden state of Long Short-Term Memory (LSTM), which is named MS-LSTM. Concretely, MS-LSTM is inspired by the seq2seq attention mechanism to generate the output sequences by making the best of previous information [14,15].

Further, an attention-based CNN-LSTM and XGBoost hybrid model was proposed to predict the stock price. The model constructed in this paper integrates the time series model, the Convolutional Neural Networks with Attention mechanism, the Long Short-Term Memory network, and XGBoost regressor in a non-linear relationship, and improved the prediction accuracy [16]. Ref. [17] introduced neutrosophic-logic-based sentiment analysis, highlighting the importance of incorporating external sentiment data, which is not considered in the current study.

In addition, ref. [18] recommended a deep learning framework for performing prediction in the stock market by considering the sentiment text and historical information from social media whereby the preprocessed sentiment texts were given to the Bidirectional Encoder Representations from Transformers (BERT) model for retrieving the first set of features from the positive and negative sentiments and the deep features were retrieved from the data using a One-Dimensional Convolutional Neural Network (1DCNN), which was considered a second feature set from historical data. The two sets of features retrieved from the sentiment text and data were passed to the Dual Scale Adaptive Residual Long Short-Term Memory with Attention Mechanism (DSAResLSTM-AM) for stock market price prediction, where the attributes of the ResLSTM were tuned using the Enhanced Deep Sleep Optimizer (EDSO).

Ref. [19] proposed a hybrid Temporal Convolutional Neural Network (TCN)-LSTM model for financial forecasting, which represents a more advanced architecture and should be included in the comparative discussion.

Ref. [20] demonstrated that multimodal models can guarantee more accurate and robust estimation than single-source ones, both on forecasting accuracy and trend precision. Also, their study showed the need for text information sentiment and numeric data to be used together to develop systems that are intelligent and responsive in a manner of helping in decision making within an intelligent FinTech environment.

Although attention-based and transformer-inspired architectures have achieved strong performance in several time series forecasting applications, their effectiveness in financial markets remains an open question. Financial time series are characterized by high volatility, non-stationarity, and low signal-to-noise ratios, which may limit the ability of increasingly complex models to consistently outperform simpler architectures. Recent studies have suggested that improvements in model complexity do not always lead to meaningful gains in forecasting performance, highlighting the need for controlled comparative studies that evaluate both predictive accuracy and practical financial outcomes.

The above existing research demonstrated that deep learning models, particularly recurrent neural networks such as LSTM and GRU, are effective in capturing nonlinear temporal dependencies in stock market data and generally outperform traditional linear and shallow learning models. More recent studies have begun exploring sequence-to-sequence architectures and attention mechanisms, showing potential advantages in modeling complex input-output relationships. However, much of the current literature focuses on isolated

model implementations or specific datasets, limiting the comparability of results across studies. Additionally, few works jointly evaluate predictive accuracy and practical trading performance under a unified experimental framework.

While recent state-of-the-art methods, such as Transformer-based models, multimodal models and Temporal Convolutional Neural Network approaches, focus on improving predictive accuracy through increasingly sophisticated architectures, they often evaluate performance primarily using statistical error metrics. In contrast, this study emphasizes a decision-oriented perspective by integrating trading profitability into the evaluation framework.

Moreover, many existing studies do not control for differences in data preprocessing, feature engineering, and training procedures, making it difficult to isolate the true impact of model architecture. The controlled experimental design adopted in this study addresses this limitation and provides clearer insights into model behavior.

Although recent studies have proposed increasingly sophisticated deep learning architectures for stock forecasting, it remains unclear whether improvements in conventional accuracy metrics translate into meaningful economic gains. Moreover, differences in datasets, feature engineering, training procedures, and evaluation criteria make direct comparisons across existing studies difficult. This study addresses these limitations through a controlled decision-oriented evaluation framework that systematically compares multiple deep learning architectures under identical experimental conditions and evaluates their effectiveness using both forecasting accuracy and trading profitability.

1.3. Key Objectives and Contributions

This study makes the following contributions to the financial time series forecasting literature:

1. Decision-Oriented Evaluation Framework:

Unlike most existing studies that focus solely on predictive accuracy, this work proposes a unified evaluation framework that links forecasting performance with trading profitability. This enables a more realistic assessment of model utility in financial decision-making contexts.

2. Controlled Comparative Analysis:

The study conducts a systematic comparison of multiple deep learning architectures under identical experimental conditions, including the same dataset, feature set, and training procedure. This controlled design isolates the effect of model architecture, which is often confounded in prior studies.

3. Insight into Model Complexity vs. Performance:

The findings demonstrate that increased architectural complexity, such as the inclusion of attention mechanisms, does not necessarily lead to improved performance in financial time series forecasting. This contributes to an emerging understanding that model effectiveness is context dependent.

4. Bridging Prediction and Economic Outcomes:

By integrating a trading strategy into the evaluation, the study highlights the disconnect between statistical accuracy and financial performance, providing practical insights for both researchers and practitioners.

5. Model-Agnostic and Extensible Framework:

The proposed framework is not limited to the models considered in this study and can be readily extended to evaluate more advanced architectures, including Transformer-based and diffusion-based models.

2. Methodology Overview

This study is designed to compare and evaluate the performance of four deep learning models for stock price forecasting accuracy and trading profitability. Four deep learning models, namely, MLP, GRU, Seq2Seq, and Seq2Seq with attention models were trained using the same datasets and experimental settings to ensure a fair comparison.

The overall methodology process consisted of (1) the collection of the raw stock data, (2) Stock data pre-processing, (3) Description of the four models, (4) Experimental Setup and Implementation, and (5) Model Configuration and Hyperparameter Setting.

Widely used open, close, high, low and volume stock data was collected and used as input data for the models. In addition, technical indicators as discussed in Section 2.2 were extracted and standardised to form features for the forecasting models too.

The accuracy of each model was evaluated using the mean absolute percentage error (MAPE) across a 5-day forecasting horizon. The practical usefulness of the models was evaluated using a rule-based trading strategy and the trading profitability of the models was compared. The above methodology enabled a comparison of both forecasting accuracy and real-world trading performance.

2.1. Recent Advances in Time Series Forecasting

Recent advances in time series forecasting have introduced a range of deep learning architectures that significantly extend beyond traditional recurrent and convolutional models. In particular, Transformer-based models have gained prominence due to their ability to capture long-range dependencies through self-attention mechanisms. Originally proposed for natural language processing [21], transformers have been adapted for time series forecasting, demonstrating strong performance across various domains.

Building on this paradigm, the Temporal Fusion Transformer (TFT) integrates attention mechanisms with recurrent layers and gating components, enabling both high predictive performance and interpretability in multi-horizon forecasting tasks [22]. TFT is particularly notable for combining static covariates, temporal features, and attention-based interpretability within a unified architecture.

More recently, PatchTST has been proposed as an efficient Transformer-based model that segments time series into patches, allowing for improved representation learning and scalability [23]. This approach has shown strong empirical performance, particularly in long-term forecasting benchmarks.

In addition, diffusion-based models have emerged as a probabilistic framework for time series forecasting. These models, inspired by generative diffusion processes, are capable of modeling complex data distributions and uncertainty, offering advantages in capturing stochastic behavior in financial time series [24,25].

While these architectures represent the current state of the art, their application in financial forecasting remains an evolving area, particularly due to challenges such as low signal-to-noise ratios, non-stationarity, and computational complexity.

2.2. Collection of the Raw Data

The dataset utilized in this study comprises trading records from 100 companies listed in the Australian Stock Market, specifically the ASX 100 index, which represents the top 100 companies by market capitalization as of December 2020 (source: Market Index (<https://www.marketindex.com.au/asx100>, accessed on 13 July 2026)). Only those companies with continuous and sufficient historical data over the full study period were included in the analysis. The data was collected through web scraping from Yahoo Finance (<https://au.finance.yahoo.com/>, accessed on 13 July 2026), covering a total of 1521 trading days, spanning from 2 January 2015 to 31 December 2020.

The ASX 100 was selected due to its liquidity, market capitalization, and data consistency, which reduce microstructure noise and missing data issues commonly observed in smaller-cap assets. Focusing on a single market allows for a controlled experimental environment in which model comparisons are not confounded by cross-market heterogeneity such as differences in regulation, trading hours, or currency effects. This design prioritizes internal validity over broad coverage, enabling clearer interpretation of model behavior under consistent conditions.

For each stock and trading day, the dataset includes key market metrics: the opening, high, low, and closing prices, along with the trading volume for that day (as shown in Table 1 below). Table 1 presents sample rows of the raw dataset, illustrating the structure and key market variables used in this study.

Table 1. Example of rows of the raw data.

	Date	Open	High	Low	Close	Volume	Code
0	2 January 2015	5.98	5.99	5.93	5.97	7,289,465	TLS
1	5 February 2015	5.97	6.00	5.94	5.98	11,299,156	TLS
2	6 January 2015	5.96	5.98	5.90	5.94	19,114,285	TLS
3	7 January 2015	5.93	5.97	5.90	5.95	16,098,189	TLS
4	8 January 2015	5.98	6.03	5.95	6.02	15,235,066	TLS

The dataset was cleaned up to ensure good-quality data. The final dataset included data from 91 stocks in Australia, since 9 stocks did not have continuity of data for the entire period under study. Table 2 below lists the stock names and their abbreviations.

Table 2. Stock Names and Abbreviation Codes.

1	AGL	AGL Energy Ltd.	47	MFG	Magellan Financial Group Ltd.
2	ALL	Aristocrat Leisure Ltd.	48	MGR	Mirvac Group
3	ALQ	Als Ltd.	49	MPL	Medibank Private Ltd.
4	ALU	Altium Ltd.	50	MQG	Macquarie Group Ltd.
5	ALX	Atlas Arteria	51	NAB	National Australia Bank Ltd.
6	AMC	Amtcor Plc	52	NCM	Newcrest Mining Ltd.
7	AMP	AMP Ltd.	53	NEC	Nine Entertainment Co Holdings Ltd.
8	ANN	Ansell Ltd.	54	NHF	NIB Holdings Limited
9	ANZ	Australia and New Zealand Banking Group Ltd.	55	NST	Northern Star Resources Ltd.
10	APA	APA Group	56	NXT	NEXTDC Ltd.
11	AST	Ausnet Services Ltd.	57	ORA	Orora Ltd.
12	ASX	ASX Ltd.	58	ORG	Origin Energy Ltd.
13	AWC	Alumina Ltd.	59	ORI	Orica Ltd.
14	AZJ	Aurizon Holdings Ltd.	60	OSH	Oil Search Ltd.
15	BEN	Bendigo and Adelaide Bank Ltd.	61	OZL	OZ Minerals Ltd.
16	BHP	BHP Group Ltd.	62	QAN	Qantas Airways Ltd.
17	BLD	Boral Ltd.	63	QBE	QBE Insurance Group Ltd.

Table 2. Cont.

18	BOQ	Bank of Queensland Ltd.	64	QUB	QUBE Holdings Ltd.
19	BPT	Beach Energy Ltd.	65	REA	REA Group Ltd.
20	BSL	Bluescope Steel Ltd.	66	RHC	Ramsay Health Care Ltd.
21	BXB	Brambles Ltd.	67	RIO	RIO Tinto Ltd.
22	CAR	CAR Group Limited	68	RMD	Resmed Inc.
23	CBA	Commonwealth Bank of Australia	69	SAR	Saracen Mineral Holdings Ltd.
24	CCL	Coca-Cola Amatil Ltd.	70	SCG	Scentre Group
25	CGF	Challenger Ltd.	71	SEK	Seek Ltd.
26	CHC	Charter Hall Group	72	SGP	Stockland
27	COH	Cochlear Ltd.	73	SGR	The Star Entertainment Group Limited
28	CPU	Computershare Ltd.	74	SHL	Sonic Healthcare Ltd.
29	CSL	CSL Ltd.	75	SKI	Spark Infrastructure Group
30	CWN	Crown Resorts Ltd.	76	SOL	Washington H Soul Pattinson & Company Ltd.
31	CWY	Cleanaway Waste Management Ltd.	77	STO	Santos Ltd.
32	DMP	Domino's PIZZA Enterprises Ltd.	78	SUN	Suncorp Group Ltd.
33	DOW	Downer EDI Limited	79	SYD	Sydney Airport
34	DXS	Dexus	80	TAH	Tabcorp Holdings Ltd.
35	EVN	Evolution Mining Ltd.	81	TCL	Transurban Group
36	FLT	Flight Centre Travel Group Limited	82	TLS	Telstra Corporation Ltd.
37	FMG	Fortescue Metals Group Ltd.	83	TPG	TPG Telecom Ltd.
38	FPH	Fisher & Paykel Healthcare Corporation Ltd.	84	TWE	Treasury Wine Estates Ltd.
39	GMG	Goodman Group	85	VCX	Vicinity Centres
40	GPT	GPT Group	86	WBC	Westpac Banking Corporation
41	IAG	Insurance Australia Group Ltd.	87	WES	Wesfarmers Ltd.
42	ILU	Iluka Resources Limited	88	WOR	Worley Ltd.
43	IPL	Incitec Pivot Ltd.	89	WOW	Woolworths Group Ltd.
44	JBH	JB Hi-Fi Ltd.	90	WPL	Woodside Petroleum Ltd.
45	JHX	James Hardie Industries Plc	91	XRO	Xero Ltd.
46	LLC	Lendlease Group			

2.3. Data Pre-Processing

2.3.1. Feature Extraction

The feature set in this paper is intentionally limited to commonly used technical indicators derived from price and volume data. This choice reflects a practical trading scenario in which such features are readily available and widely used. While incorporating additional data sources such as macroeconomic variables, fundamental indicators, or alternative data (e.g., news sentiment) may improve predictive performance, the use of a standardized feature set enables a fair and controlled comparison across models and isolates the contribution of model architecture.

For each stock, the input data for a given trading day is represented as a vector comprising 10 features. Five of these features are derived directly from the raw data: the opening, high, low, and closing prices, as well as the trading volume. The remaining five

features are technical indicators selected from the work of [26]. These technical indicators are calculated over a 5-day window, with an output sequence length of $n = 5$. The 5-day rolling window approach preserves temporal order and prevents information leakage.

The Simple Moving Average (SMA) is computed as shown in Equation (1).

Simple Moving Average (SMA):

$$SMA_n = \frac{1}{n} \sum_{i=0}^{n-1} C_{t-i} \quad (1)$$

where C_t is the close price at time t .

The technical indicators are defined by Equations (2)–(5).

Average True Range (ATR)

$$ATR_n = EMA_n(\max(H_t - L_t, |H_t - C_{t-1}|, |L_t - C_{t-1}|)) \quad (2)$$

where H_t, L_t is the high and low price at time t and $EMA_n = \sum_{i=0}^{n-1} \omega_i C_{t-i}$ with $\sum_{i=0}^{n-1} \omega_i = 1$.

Average Directional Index (ADX)

$$ADX_n = 100 * (DI_n^+ - DI_n^-) / (DI_n^+ + DI_n^-) \quad (3)$$

where

$$DI_n^+ = 100 * EMA_n(\max(C_t - C_{t-1}, 0)) / ATR_n$$

$$DI_n^- = 100 * EMA_n(\min(C_t - C_{t-1}, 0)) / ATR_n$$

Stochastic %K

$$\%K_n = 100 * (C_t - LL_n) / (HH_n - LL_n) \quad (4)$$

where HH_n and LL_n are the mean highest high and lowest low prices in the last n days.

Stochastic %D

$$\%D_n = EMA_3(\%K_n) \quad (5)$$

2.3.2. Input Preparation

To ensure consistency across multiple stocks, all input features are standardized to the same scale. We apply normalization by transforming each feature to have a mean of zero and a standard deviation of one. Specifically, for each stock, we standardize the data by subtracting the mean and dividing by the standard deviation of each feature.

The dataset is segmented into input–output pairs using the rolling window technique (see Figure 1). For each stock, a 25-day sliding window generates data points, where each input sequence consists of 20 days (approximately one month) of feature vectors, and each output sequence corresponds to the next 5 days (approximately one week) of closing prices.

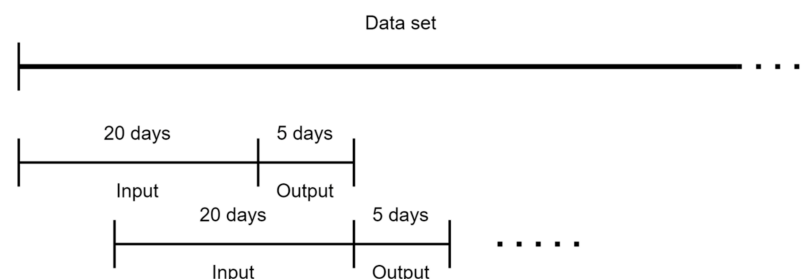


Figure 1. Generate input and output sequence using rolling window method.

The dataset is then split into training, validation, and test sets. The test set comprises 508 trading days, spanning from 2 January 2019 to 31 December 2020. The remaining data,

consisting of 1013 trading days from 2 January 2015 to 31 December 2018, is randomly divided into training (80%) and validation (20%) sets.

2.4. Description of the Four Models

For a given stock and time step, let $x_1, x_2, \dots, x_T$ be the sequence of input vectors with d features. We aim to find a model that produces the M -day-ahead close price prediction sequence $\hat{y}_{T+1}, \hat{y}_{T+2}, \dots, \hat{y}_{T+M}$ that is as close to the actual prices $y_{T+1}, y_{T+2}, \dots, y_{T+M}$ as possible.

2.4.1. Multi-Layer Perceptron

The Multi-Layer Perceptron (MLP) is one of the most widely used feedforward neural network architectures and has been extensively applied as a baseline model in financial time series forecasting. Due to its ability to approximate nonlinear functions, MLP has been shown to outperform traditional linear models in various stock price prediction tasks [6,7,11]. Although MLP does not explicitly model temporal dependencies, it provides a useful benchmark for evaluating the benefits of sequence-based architectures.

In this study, an MLP with a single hidden layer is employed to predict the next n -day closing prices. The input sequence is flattened into a single feature vector, which is then transformed through a nonlinear hidden layer to generate the output predictions.

An MLP with one hidden layer was applied to generate the closing prices of the next M days. See Equation (6) below. Each T -day series of input vector $x_1, x_2, \dots, x_T$ were first concatenated into a single vector x'_t of length $d_{in} = d \times T$. This vector was transformed into a hidden vector of size d_h and next transformed into the prediction of size M :

$$\hat{y} = W_o \text{RELU}(W_h x'_t + b_h) + b_o \quad (6)$$

where $W_h, \in \mathbb{R}^{d_{in} \times d_h}, b_h \in \mathbb{R}^{d_h}, W_o \in \mathbb{R}^{d_h \times M}, b_o \in \mathbb{R}^M$ are trainable parameters.

2.4.2. GRU Network

The Gated Recurrent Unit (GRU) is a recurrent neural network architecture designed to capture temporal dependencies in sequential data while mitigating the vanishing gradient problem commonly encountered in standard RNNs [2].

GRU models have been widely applied to financial time series forecasting and have demonstrated higher performance than the MLP model in capturing dynamic market behaviors [8–10]. In this study, the GRU model serves as a strong sequential baseline for comparison with the proposed sequence-to-sequence architectures.

The Gated Recurrent Unit (GRU) is a variant of the Recurrent Neural Network (RNN) designed to capture long-term dependencies in sequential data. Due to its specialized architecture for time series forecasting, GRU is expected to outperform Multi-Layer Perceptron (MLP) models in stock price prediction tasks. While GRU shares similarities with the Long Short-Term Memory (LSTM) model, it requires fewer computational resources, resulting in faster training times. GRU utilizes a single hidden state to propagate information from previous time steps, with two primary gates, the reset gate and the update gate, regulating the flow of information and controlling the network's learning process.

The reset gate of GRU (see Equation (7) below) discards some of the information from the previous hidden state h_{t-1} when the new memory $\tilde{h}_t$ (see Equation (8) below) is calculated, while the update gate z_t (see Equation (9) below) controls the update of the new hidden state $h_t \in \mathbb{R}^{d_h}$. See Equation (10) below.

$$r_t = \text{Sigmoid}(W_r[x_t, h_{t-1}] + b_r) \quad (7)$$

$$\tilde{h}_t = \tanh(W_n[x_t, r_t \times h_{t-1}] + b_n) \quad (8)$$

$$z_t = \text{Sigmoid}(W_z[x_t, h_{t-1}] + b_z) \quad (9)$$

$$h_t = (1 - z_t) \times \tilde{h}_t + z_t \times h_{t-1} \quad (10)$$

where $W_r, W_n, W_z \in \mathbb{R}^{(d+d_h) \times d_h}$, $b_r, b_n, b_z \in \mathbb{R}^{d_h}$ are trainable parameters.

The context vector is calculated as a weighted average of the encoder hidden states, as defined in Equation (11). The last hidden state h_T from the GRU summarizes the information from the input sequence, to generate the output vector $\hat{y} = (\hat{y}_{T+2}, \dots, \hat{y}_{T+M})$ through a linear layer

$$\hat{y} = W_o h_T + b_o \quad (11)$$

where $W_o \in \mathbb{R}^{d_h \times M}$, $b_o \in \mathbb{R}^M$ are trainable parameters.

2.4.3. Sequence-to-Sequence Model

To enhance the Seq2Seq architecture, an attention mechanism is incorporated into the decoder to allow the model to selectively focus on different parts of the input sequence when generating each prediction. Rather than relying solely on the final hidden state of the encoder, the attention mechanism computes a context vector as a weighted combination of all encoder hidden states. This enables the model to assign varying importance to different time steps in the input sequence, potentially improving predictive performance when relevant information is distributed across the input window.

The Seq2Seq architecture consists of two networks as shown in Figure 2: an encoder and a decoder. The encoder is a GRU network which processes each time step of the T -day input sequence $x_1, x_2, \dots, x_T$ and produces hidden states $h_1, h_2, \dots, h_T$. The decoder is also a GRU network which receives the last encoder's hidden state h_T as the initial state. The previous output of the decoder is used as the input to generate the next prediction. By having a GRU decoding network, the model produces sequential outputs.

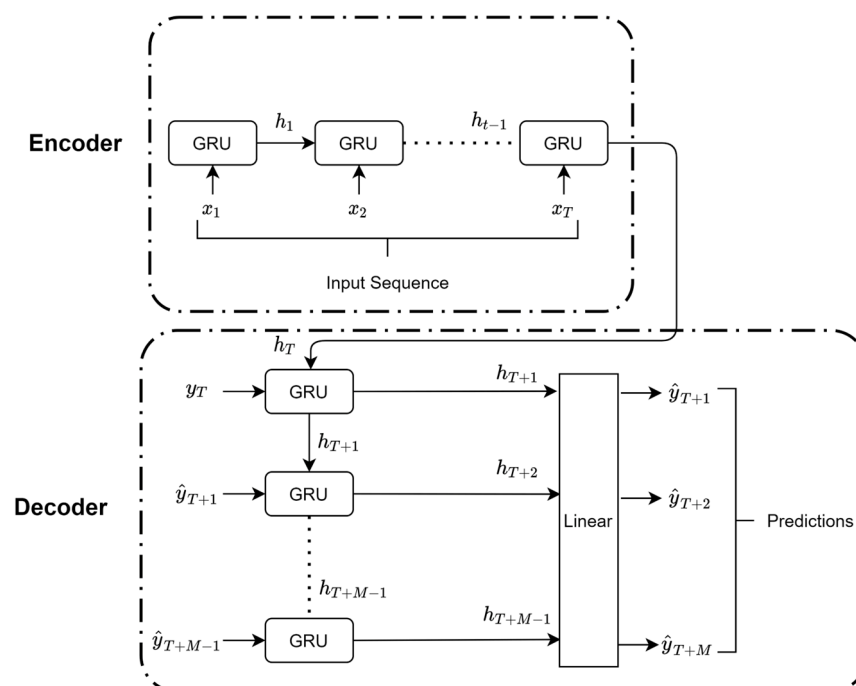


Figure 2. Seq2Seq model architecture.

Figure 3 below illustrates the architecture of the Seq2Seq model with the attention mechanism integrated into the decoder.

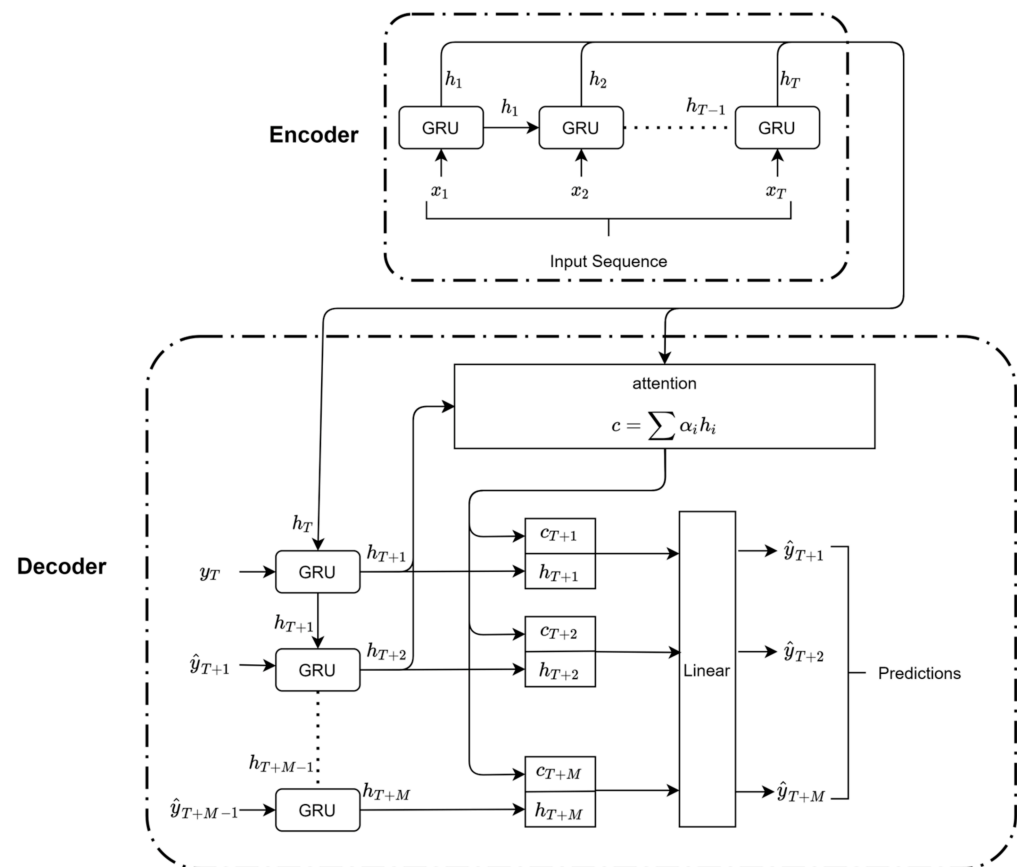


Figure 3. Seq2Seq model architecture with attention.

2.4.4. Attention Model

The attention mechanism is often applied in the decoder of a Seq2Seq model (Figure 3) to provide additional context for each prediction. It utilizes all the hidden states from the encoder to form a context vector. This vector helps the model focus more on certain parts of the input sequence for each stock and prediction time step. In detail, at time step $t, t = T, \dots, T + M - 1$ of the decoding process, each encoder's hidden state $h_i, i = 1, \dots, T$ is given a score $\beta_{t,i}$, calculated from h_i and the current hidden state h_t . See Equation (12) below.

$$\beta_{t,i} = v_a^T \tanh(W_a[h_i, h_t] + b_a) \quad (12)$$

where $W_a \in \mathbb{R}^{d_h \times d_h}, v_a, b_a \in \mathbb{R}^{d_h}$ are parameters to be trained.

Then, by using the softmax function, the scores $\beta_{t,i}$ are transformed into attention weights $\alpha_{t,i}$ such that each $\alpha_{t,i}$ (defined in Equation (13) below) is between 0 and 1 and $\sum_{i=1}^T \alpha_{t,i} = 1$

$$\alpha_{t,i} = \frac{\exp(\beta_{t,i})}{\sum_{j=1}^T \exp(\beta_{t,j})} \quad (13)$$

The context vector c_t (defined in Equation (14) below) is calculated by averaging the encoder's hidden states weighted by $\alpha_{t,i}$.

$$c_t = \sum_{i=1}^T \alpha_{t,i} h_i \quad (14)$$

Finally, the context vector c_t and the hidden state h_t are concatenated into a single vector and, through a linear layer, the close price prediction is generated.

2.5. Experimental Setup and Implementation

All experiments were conducted using the Python 3.11 programming language. The deep learning models were implemented using TensorFlow and Keras, while data processing and feature engineering were performed using NumPy and pandas. Model training and evaluation were carried out on a workstation equipped with an Intel-based CPU, 32 GB RAM, and an NVIDIA GPU to accelerate neural network training.

The models were trained using the Adam optimiser with a fixed learning rate, and early stopping was applied to prevent overfitting. To ensure a fair comparison, random seeds were fixed to improve the reproducibility of the experimental results. In addition, the hyperparameters were chosen based on preliminary experiments and the same hyperparameters were used across all the models.

2.6. Model Configuration and Hyperparameter Setting

To ensure a fair comparison, all the models used a consistent experimental framework. Hyperparameter values were selected based on preliminary exploratory experiments and commonly adopted settings reported in prior financial time-series forecasting studies. For example, the GRU and the Seq2Seq models used a batch size of 32 with 64 hidden units and a single recurrent layer to manage the overfitting risk. A dropout with a rate of 0.2 was applied to mitigate overfitting where applicable. The MLP baseline model also employed one hidden layer with 64 neurons.

The Adam optimiser with a learning rate of 0.001 was used for all models. The mean squared error was used as the loss function. Training of the models was conducted for a maximum of 100 epochs with early stopping based on the validation loss. Table 3 below shows the hyperparameter settings for all models.

Table 3. Hyperparameter Settings for All Models.

Model	Hidden Units	Layers	Optimizer	Learning Rate	Batch Size	Dropout
MLP	64	1	Adam	0.001	32	0.2
GRU	64	1	Adam	0.001	32	0.2
Seq2Seq	64	1	Adam	0.001	32	0.2
Seq2Seq with Attention	64	1	Adam	0.001	32	0.2

3. Model Results

3.1. Model Accuracy

All 4 models are trained on the Australian Stock Market data with $T = 20$, $M = 5$ and $d = 10$. The forecasting accuracy of the models was evaluated using the Mean Absolute Percentage Error (MAPE). MAPE was selected due to its easy interpretability and widespread use in financial forecasting studies. MAPE is defined in Equation (15) below as:

$$MAPE = \sum_{i=1}^T \left| \frac{\hat{y}_t - y_t}{y_t} \right| \quad (15)$$

where y_t denotes the actual closing price at time step t , $\hat{y}_t$ represents the predicted closing price at the same time step, and T is the total number of prediction points in the evaluation period.

Table 4 below presents the model accuracy, with the lowest errors highlighted in bold. The Seq2Seq model achieved the lowest average MAPE (0.0293), followed closely by the attention-based model (0.0294). Although these results suggest that sequence-based architectures may provide improved forecasting performance, the small magnitude of the differences indicates that formal statistical testing is required to determine whether

the observed performance gap is statistically significant. Although the Seq2Seq model achieved the lowest average MAPE, the numerical differences between the competing deep learning models were relatively small. Consequently, these results should be interpreted as descriptive empirical observations rather than statistically confirmed evidence of superior forecasting performance. Without formal statistical inference, such as bootstrap confidence intervals or hypothesis testing, it cannot be concluded that the observed differences exceed the level expected from random variation.

Table 4. Mean absolute percentage error for each prediction day.

Model \ Day	1	2	3	4	5	Average
MLP	0.0205	0.0291	0.0346	0.0428	0.0477	0.0349
GRU	0.0196	0.0268	0.0323	0.0381	0.0420	0.0318
Seq2Seq	0.0172	0.0240	0.0301	0.0351	0.0394	0.0293
Attention	0.0183	0.0241	0.0297	0.0349	0.0400	0.0294

The Seq2Seq with Attention model closely followed, with a MAPE of 0.0294. In contrast, the MLP model performed the worst, with a MAPE of 0.0349. It was noticed that the MAPE was the lowest on the first prediction day and tended to increase as the prediction horizon extended. The Seq2Seq model exhibited the smallest errors on days 1, 2, and 5, while the Seq2Seq with Attention model slightly outperformed the Seq2Seq model on days 3 and 4.

The model performances are aligned with our expectations. The GRU model, which is designed to handle sequential data, outperformed the MLP model. The Seq2Seq model had enhanced its performance by incorporating a GRU-based decoder, which enabled the Seq2Seq model to treat both the input and output sequences in a unified manner. However, the attention mechanism did not yield a significant improvement in performance, suggesting that its benefits may be context-dependent or require further optimization for this task. Table 4 below shows the mean absolute percentage error for each prediction day.

Seq2Seq and the attention-enhanced variant are relatively small. This suggests that the incremental predictive benefit of adding an attention mechanism in this context is limited.

However, this finding is itself informative. It indicates that, for short-term stock price forecasting using structured financial time series data, the additional complexity introduced by attention mechanisms does not necessarily translate into meaningful predictive gains. This contrasts with their strong performance in domains such as natural language processing, where long-range dependencies are more prominent.

All models performed optimally on the ‘AusNet Services Ltd.’ (AST) stock, achieving very low MAPE values: 0.0170 for MLP, 0.0150 for GRU, 0.0147 for Seq2Seq, and 0.0146 for the Attention model (Figure 4 below).

However, the models struggled to capture the sudden, unforeseen fluctuations in the time series. A notable event impacting performance was the COVID-19 pandemic, which began in early 2020 and caused significant disruptions across many stocks. For instance, the ‘Flight Centre Travel Group Limited’ (FLT) stock experienced a sharp decline in April 2020, which the MLP model failed to adapt to. Similarly, the GRU, Seq2Seq, and Attention models were unable to accurately model the abrupt changes in the ‘Magellan Financial Group Limited’ (MFG) stock data during the same period (Figure 5 below).

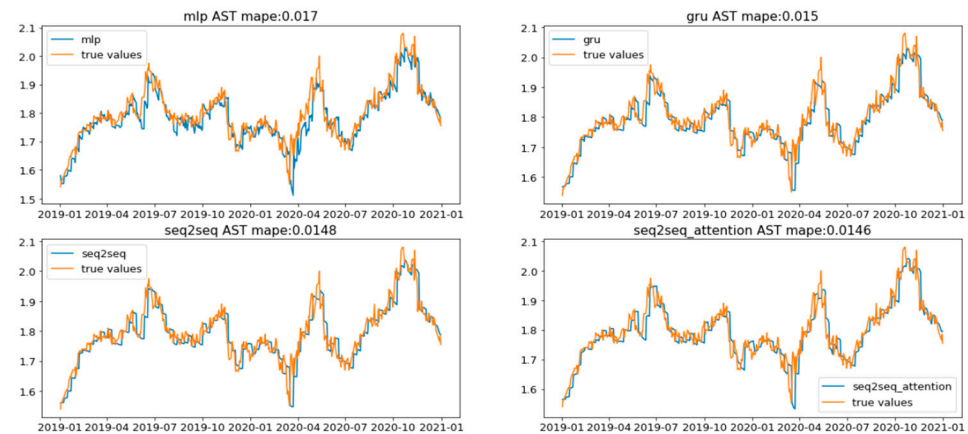


Figure 4. Test set predictions of AST stock.

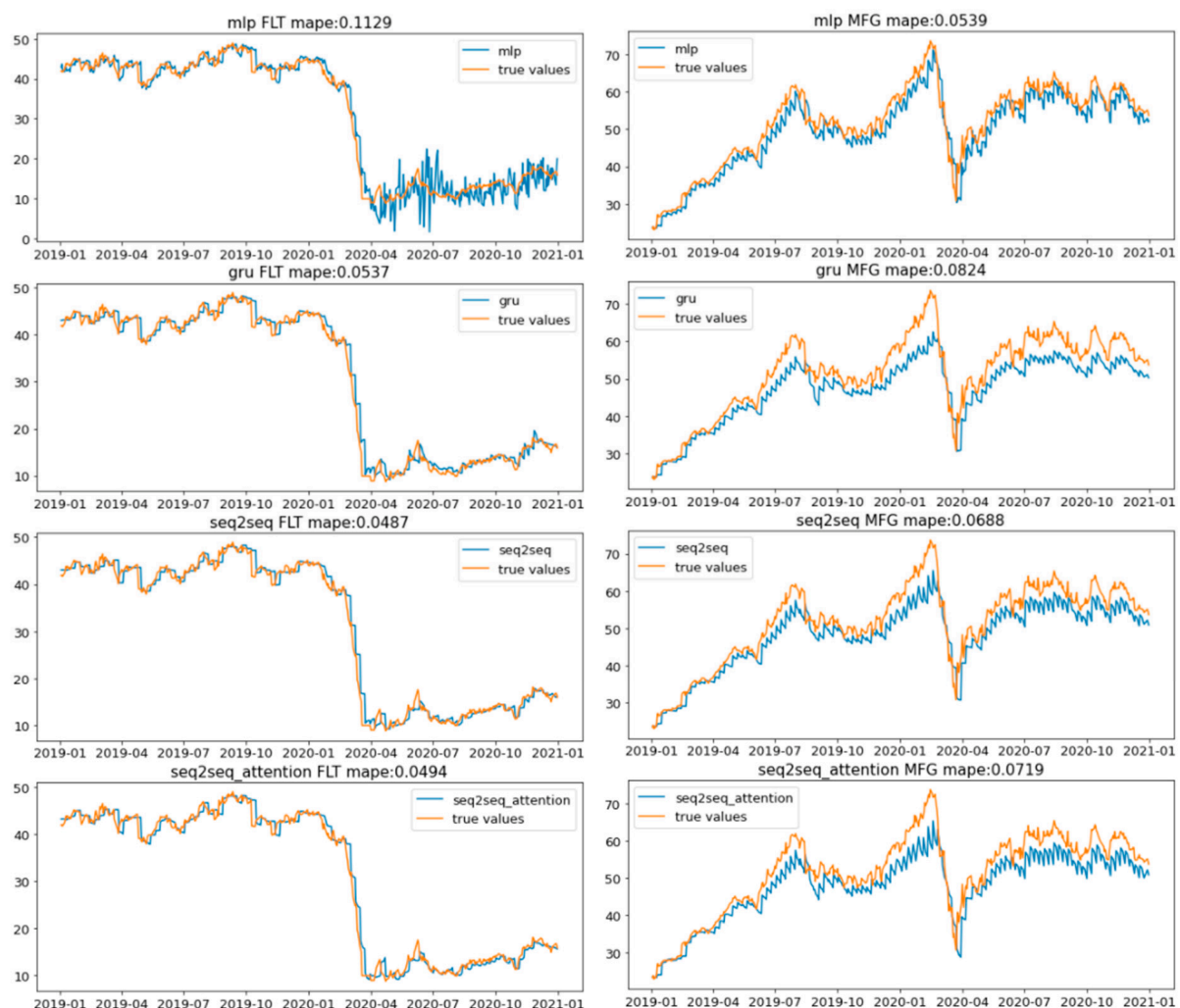


Figure 5. Test set predictions for the FLT and MFG stocks.

Although MAPE is used as the primary reported metric in this study, a comprehensive evaluation framework for financial time series forecasting should include additional metrics such as the Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE) to measure error magnitude, as well as the Hit Ratio to assess directional prediction accuracy. These metrics are particularly important in trading applications, where correctly predicting the direction of price movement can be more critical than minimizing numerical errors.

Although MAPE is the primary evaluation metric used in this study, directional metrics such as Hit Ratio are also important in financial forecasting because profitable trading often depends on correctly predicting price movement rather than minimizing absolute prediction error. The inclusion of Hit Ratio will be considered in future extensions of this framework.

3.2. Feature Sensitivity Analysis

Interpreting deep learning models is challenging due to their inherent complexity and the difficulty of directly analyzing internal parameters. In this study, a perturbation-based sensitivity analysis was employed as a proxy to assess the relative influence of input features on model outputs.

Specifically, small random noise sampled from a normal distribution was added independently to each input feature, and the resulting change in model predictions was measured using the Root Mean Squared Deviation (RMSD). Features that produced larger deviations in the output were considered more influential.

It is important to note that this approach measures sensitivity rather than true feature importance. The method captures how responsive the model is to perturbations in individual inputs, rather than providing a causal or attribution-based explanation of model behavior.

Figure 6 below illustrates the sensitivity derived from each model by averaging the scores across 5000 randomly selected samples from the training dataset. The x -axis represents the input features, while the y -axis denotes the days within the input sequence, with day 0 corresponding to the most recent day and day 19 representing the farthest past day. Lighter colors indicate higher sensitivity scores.

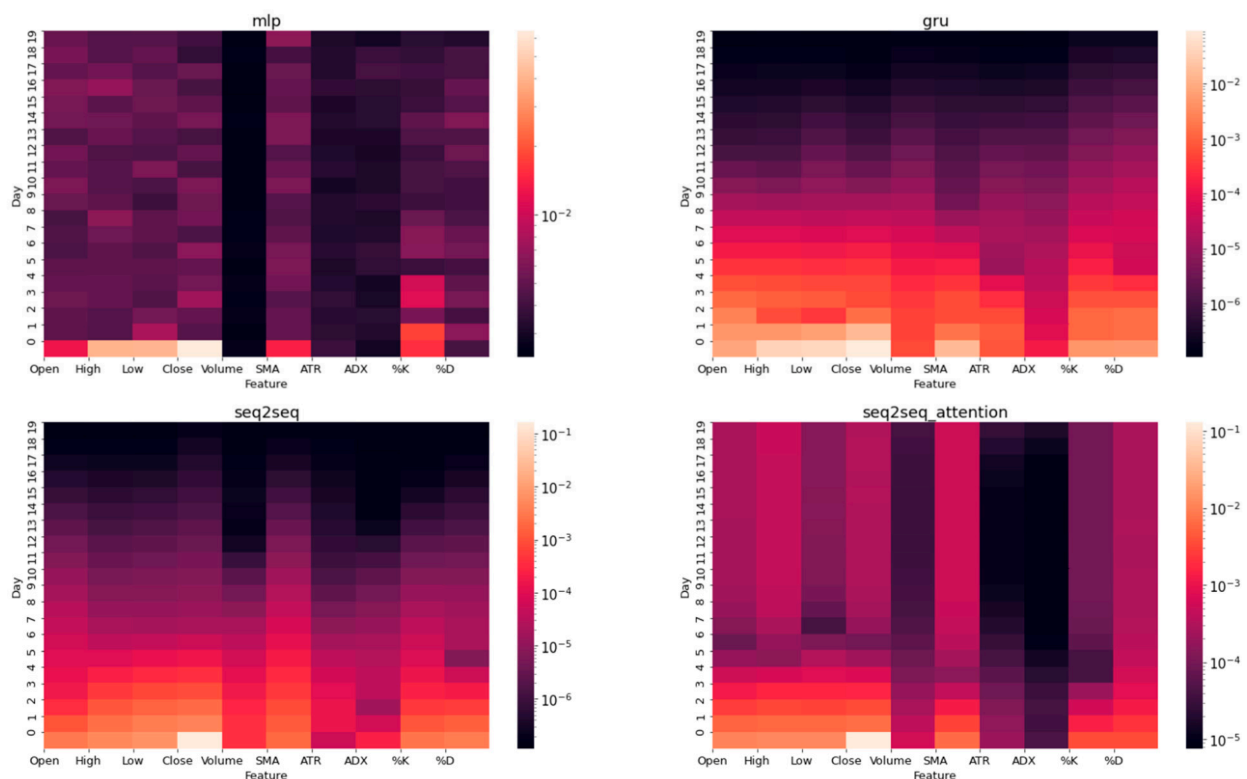


Figure 6. Sensitivity Scores.

The sensitivity magnitudes vary across models due to differences in their architectures and output response scales. Therefore, the absolute values should be interpreted within each model rather than used as direct comparisons between different model architectures.

Most dominant across all four models was the close price, particularly in the MLP, Seq2Seq, and Attention models. Other features, such as the open, high, and low prices, also exhibited substantial importance. Surprisingly, trading volume proved to be the least significant feature, exerting minimal influence on the model forecasts. Technical indicators generally showed comparatively lower importance. Specifically, the Simple Moving Average (SMA), Stochastic %K, and %D demonstrated moderate importance, while the Average True Range (ATR) and Average Directional Index (ADX) contributed minimally.

The GRU, Seq2Seq, and Seq2Seq with Attention models, benefiting from the GRU's ability to capture long-term dependencies, assign higher importance to recent trading days, with the significance decreasing as the models consider more distant past data. Conversely, the MLP model primarily focuses on the most recent data point, effectively underutilizing the broader input sequence.

3.3. Effect of the Attention Mechanism

Since the Attention model did not yield any performance improvements over the Seq2Seq model, we conducted a deeper analysis of the model's outputs to gain further insights. Specifically, we examined the Attention weights assigned to each day in the input sequence during the calculation of the context vector. Although the Attention mechanism typically assigns individual scores to each input sequence, we were able to identify a general pattern by averaging the weights across all sequences.

Figure 7 below presents the averaged attention weights, computed from 5000 randomly selected data points from the training set. This is a **heatmap**, where the colours represent the magnitude of the values in each cell. The numbers printed in red are the actual values, while the colours provide a visual summary. The colours indicate the relative size of the values. Dark purple indicates the lowest values, teal indicates slightly lower values, green indicates the middle values and yellow the highest values.

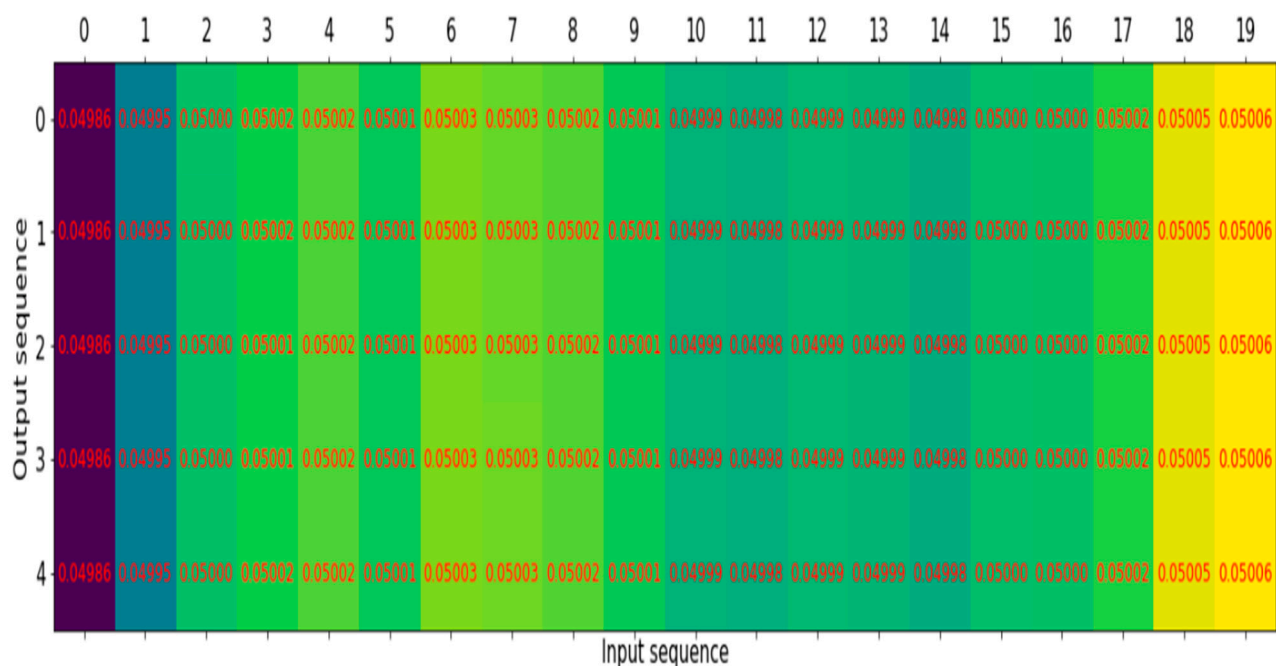


Figure 7. Attention weights.

The attention weights are generally close to the balanced value of 0.05, with a slight tendency for input days closer to the prediction date to have marginally higher weights. Conversely, input days further in the past had lower attention weights. In addition, the variation in attention weights across the days of each output sequence was minimal. As a result, the context vector generated by the Attention mechanism contributed very little additional information to the forecasts, which helped to explain why the performance of the Seq2Seq with Attention model closely aligned with that of the Seq2Seq model. Given the higher complexity of the Seq2Seq with Attention model, the Seq2Seq model is more favorable, offering similar forecasting accuracy and a lower computational overhead.

3.4. Computational Efficiency Results

Computational efficiency was evaluated in addition to forecasting accuracy to assess the practical feasibility of the models. The same computing environment was used for training and testing all the models. The training time was measured as the average time required to complete one full training process for a single stock, including the forward and backward passes.

The MLP model was computationally the most efficient due to its feedforward architecture. The GRU model required longer training times than the MLP model because of their sequential processing nature. Seq2Seq-based models, particularly those incorporating attention mechanisms, incurred the highest computational time and cost due to their encoder–decoder structure and additional attention calculations. Despite this, the increase in training time remained manageable and justified by the observed improvements in the forecasting accuracy.

Training times were averaged across multiple runs under identical experimental settings. Table 5 below displays the average training time per model.

Table 5. Average Training Time per Model.

Model	Avg. Training Time (s)
MLP	45
GRU	120
Seq2Seq	180
Seq2Seq + Attention	210

4. Trading Strategy Profitability

The primary objective of a stock market prediction model is its ability to generate profit. Since prediction errors alone do not capture this aspect, we further evaluate the models by implementing a trading strategy based on the 5-day-ahead predictions to generate buy and sell signals. The strategy is governed by a set of rules to determine the most suitable stocks to buy and sell. Only the top 80 stocks with the lowest MAPE performance in 2018 for each model were selected to execute trades during the 2019–2020 testing period.

The proposed trading strategy is designed as a controlled evaluation framework for comparing the relative economic utility of different forecasting models rather than as a fully deployable trading system. A fixed investment amount is allocated under identical trading rules for all models, ensuring that differences in profitability are attributable to variations in predictive signals rather than differences in trading assumptions. Transaction costs are incorporated to account for practical trading expenses; however, other market factors, such as slippage, liquidity constraints, and advanced portfolio risk management techniques, are not explicitly modeled.

The trading strategy was tested under the following assumptions, which can be easily generalized:

- All trades are executed at the end of each 5-day period, using the closing price of that day.
- A fixed budget of \$2000 is allocated to purchase stocks in each trade.
- A trading fee of \$20 per buy or sell action is applied, paid separately from the purchase budget.

Every 5-day interval, the model generates close price predictions for the next 5 days. Based on these predictions, the return for each stock is calculated by measuring the percentage change on the fifth day relative to the current closing price. Buy and sell actions are determined according to the following rules:

- Buy: The stock with the highest predicted positive return is purchased, provided at least 3 out of the 5 days show an upward trend compared to the previous day.
- Sell: The stock is sold if it shows a negative return, with at least 3 out of the 5 days demonstrating a downward trend compared to the previous day.

The final income was calculated by liquidating the remaining stocks in the portfolio. To provide additional context for evaluating the proposed trading strategy, we compare the model performance with a passive buy-and-hold investment in the S&P/ASX 200 Price Index over approximately the same evaluation period (January 2019–December 2020). The S&P/ASX 200 is widely used as the benchmark for the Australian equity market and provides an appropriate reference for assessing whether the proposed forecasting models generate returns beyond general market movements.

The Seq2Seq model achieved the highest observed ROI (23.2%) under the specified trading framework, while the Attention model achieved 12.4%. However, these differences should be interpreted with caution because no statistical significance testing was performed to determine whether the profitability differences exceed what may occur due to random market variation. While the Seq2Seq model achieved the highest observed return on investment under the proposed evaluation framework, these profitability differences should be interpreted with appropriate caution. Because formal statistical significance testing was not performed, it cannot be determined whether the observed differences in ROI represent genuine differences in predictive capability or are attributable to random variation within the evaluation period. Consequently, the reported trading results should be regarded as descriptive evidence rather than definitive proof of superior trading performance.

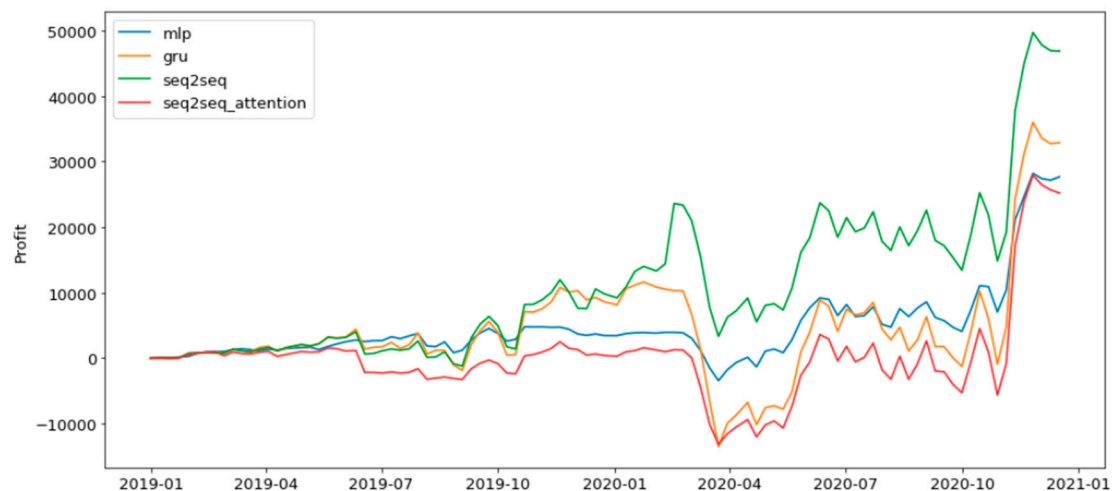
The comparison with the passive market benchmark provides additional context for interpreting the observed trading profitability. While the Seq2Seq model achieved the highest observed return under the proposed trading framework, the benchmark comparison indicates whether the model generated returns beyond those obtainable through passive exposure to the Australian equity market. Nevertheless, because the study did not perform formal statistical significance testing and employed a simplified trading strategy, these differences should be interpreted as descriptive rather than conclusive evidence of superior trading performance. Table 6 below displays the trading profits for the different models.

The benchmark represents the approximate price return of the S&P/ASX 200 Price Index over the January 2019–December 2020 evaluation period based on official ASX historical monthly index values. It is provided solely as a passive market reference and is not directly comparable to the model trading strategy, which excludes dividend income and incorporates different trading assumptions.

Table 6. Trading Profits.

Model \ Metric	Profit (\$)	Profit/Trade (\$)	Return On Investment (%)
MLP	27,508	272	13.6%
GRU	32,705	323	16.2%
Seq2Seq	46,739	462	23.2%
Attention	25,067	248	12.4%
Buy-and-Hold (S&P/ASX 200 Price Index)	-	-	14%

To provide additional context for interpreting the trading results, the proposed forecasting models were compared with a passive buy-and-hold investment in the S&P/ASX 200 Price Index over the same evaluation period. The buy-and-hold benchmark achieved an approximate price return of 14.0%, compared with the observed returns of 13.6%, 16.2%, 23.2%, and 12.4% for the MLP, GRU, Seq2Seq, and Attention models, respectively. Under the proposed evaluation framework, the Seq2Seq model outperformed the passive benchmark, while the GRU model produced a modest improvement and the MLP and Attention models performed similarly to or below the benchmark. Nevertheless, because formal statistical significance testing was not performed and the proposed trading strategy is intentionally simplified, these observed differences should be interpreted as descriptive rather than conclusive evidence of superior trading performance. Figure 8 below illustrates the accumulated trading profits for each model over the testing period. Overall, the Seq2Seq model demonstrates consistently positive profit, exhibiting a generally upward trend.

**Figure 8.** Accumulated trading profit over time.

This sustained profit can be attributed to the model's effective buy and sell decisions. The Seq2Seq model successfully sold many shares before stock prices dropped in March 2020. The Attention model experienced losses due to poor decisions in 2019, further exacerbated by the crisis in 2020. However, as the stock market recovered toward the end of 2020, the GRU, Seq2Seq, and Attention models managed to make significant profits.

The proposed trading strategy is intentionally simplified to provide a consistent and controlled framework for comparing the relative economic value of different forecasting models. It does not aim to represent a fully realistic trading system, as factors such as slippage, portfolio diversification, position sizing, and advanced risk management are not considered.

5. Discussion

5.1. Interpretation of Results

The empirical results demonstrate that sequence-based models, particularly the Seq2Seq architecture, outperform the MLP baseline in forecasting accuracy and trading profitability within the ASX 100 dataset. This outcome is consistent with the expectation that models explicitly designed to capture temporal dependencies are better suited for financial time series, where sequential patterns and short-term momentum effects are prevalent.

However, the marginal difference in MAPE between the Seq2Seq and attention-based models suggests that increasing architectural complexity does not necessarily lead to proportional performance gains. This finding provides an important insight into the role of model complexity in financial forecasting. Although attention mechanisms have demonstrated remarkable success in domains such as natural language processing, their effectiveness depends on the existence of meaningful long-range dependencies and stable patterns that can be exploited by the model. In contrast, financial markets exhibit high levels of noise, non-stationarity, and low signal-to-noise ratios, which may limit the ability of additional model complexity to extract useful predictive information. Consequently, more sophisticated architectures may offer diminishing returns and, in some cases, may increase the risk of overfitting without improving practical trading outcomes.

More broadly, these findings challenge the assumption that increasingly complex deep learning architectures will consistently deliver superior results in financial forecasting, emphasizing that model suitability depends on the characteristics of the underlying data, the robustness of the evaluation framework, and the practical objectives of the forecasting task. The contribution of these findings is not that attention mechanisms are ineffective; rather, their effectiveness is context-dependent, and financial time series may not always contain sufficient exploitable structure for greater model complexity to provide additional economic value.

These findings also contribute to the broader discussion on financial market predictability. From the perspective of market efficiency, the presence of weak and time-varying predictive signals implies that improvements in statistical forecasting accuracy may not necessarily translate into consistent economic advantages. Financial markets are adaptive and non-stationary systems in which relationships between predictors and prices can evolve due to changes in economic conditions, investor behaviour, and market regimes. Consequently, while advanced deep learning models may uncover complex non-linear patterns, their practical benefits may be constrained by the limited persistence and stability of exploitable signals. This highlights the importance of evaluating forecasting models not only based on prediction error but also on their ability to support economically meaningful decisions.

Importantly, the results indicate that relatively small differences in forecasting accuracy were associated with larger observed differences in trading profitability under the proposed evaluation framework. However, because the study did not include formal statistical inference, these observations should not be interpreted as establishing a statistically significant relationship between forecasting accuracy and economic performance. Rather, they provide empirical evidence suggesting that such a relationship may exist and warrants further investigation using more rigorous statistical evaluation.

An additional comparison with a passive buy-and-hold investment in the S&P/ASX 200 benchmark provides further context for interpreting the trading results. Over the evaluation period (January 2019–December 2020), the benchmark achieved an approximate return of 14.0%, whereas the Seq2Seq model achieved an observed return on investment of 23.2% under the proposed trading framework. This comparison suggests that the Seq2Seq model generated higher observed returns than passive market exposure during

the evaluation period. Nevertheless, this finding should be interpreted with appropriate caution. Because the study did not include formal statistical significance testing or risk-adjusted performance evaluation, it cannot be concluded that the observed outperformance reflects a statistically significant or consistently superior trading strategy. Instead, the benchmark comparison provides descriptive evidence that the forecasting model may have captured useful predictive signals beyond those reflected in the overall market movement.

5.2. Comparison with Existing Literature

The findings align with prior research showing that recurrent and sequence-based models outperform traditional feedforward architectures in time series forecasting (e.g., [22]). However, unlike studies that report significant gains from attention-based mechanisms, the results here indicate that such benefits may be context-dependent.

Recent work has suggested that Transformer-based models do not consistently outperform simpler architectures in financial time series forecasting, particularly in environments characterized by high noise and non-stationarity [11]. The results of this study provide further empirical support for this perspective, demonstrating that the added complexity of attention mechanisms does not necessarily yield improved performance in short-term stock price prediction.

Furthermore, this study extends existing literature by linking predictive performance to trading outcomes, an aspect that is often overlooked in model comparison studies. This provides a more practical evaluation of model effectiveness in real-world financial applications.

5.3. Practical Implications

The findings have several important implications for practitioners and researchers. First, the results suggest that model selection should consider not only predictive accuracy but also economic impact. Models with similar error metrics may produce significantly different trading outcomes, highlighting the importance of decision-oriented evaluation.

Second, the limited benefit of attention mechanisms in this context suggests that simpler models may be preferable in practice, particularly when computational efficiency and interpretability are important considerations.

Third, although the study reports differences in forecasting accuracy and trading profitability across competing models, formal statistical inference was not performed. Consequently, it cannot be determined whether the observed differences are statistically significant or reflect random variation. Future research should incorporate directional accuracy measures, bootstrap confidence intervals, and hypothesis-testing procedures, such as the Diebold–Mariano test, to provide a more rigorous assessment of model performance.

The findings of this study suggest that improvements in model architecture alone may not be sufficient to achieve meaningful gains in financial applications. While state-of-the-art models aim to maximize predictive accuracy, this study demonstrates that practical performance depends on how predictions translate into decisions. This highlights the need for evaluation frameworks that go beyond accuracy metrics and incorporate economic outcomes such as profitability outcomes, an aspect that remains underexplored in much of the current literature.

For practitioners, the findings suggest that model selection should be aligned with the intended decision-making objective. In particular, models should not be selected solely based on lower forecasting errors, as small improvements in metrics such as MAPE may not correspond to superior financial outcomes. When deploying forecasting systems in practical trading environments, considerations such as trading profitability, computational efficiency, model complexity, and implementation costs should be evaluated jointly.

In addition, the comparison with a passive market benchmark further illustrates the practical relevance of evaluating forecasting models using both predictive accuracy and investment performance. While conventional error measures indicated only marginal differences between competing models, the Seq2Seq model achieved a higher observed return than the passive benchmark under the proposed evaluation framework. This observation reinforces the importance of complementing statistical forecasting metrics with economically meaningful evaluation measures when assessing financial forecasting models.

5.4. Limitations

This study has several limitations that should be acknowledged. First, the analysis is restricted to the ASX 100, which may limit the generalizability of the findings to other markets with different characteristics. Second, the feature set is limited to technical indicators, excluding potentially informative variables such as macroeconomic data, firm fundamentals, and alternative data sources.

Third, the evaluation relies primarily on MAPE, without formal statistical validation such as confidence intervals or hypothesis testing. Although model performance is consistent across multiple stocks, the absence of statistical inference limits the strength of the conclusions.

Fourth, the feature sensitivity analysis is based on a perturbation approach, which may be affected by feature correlation and the choice of perturbation magnitude. As a result, the findings should be interpreted as indicative rather than definitive.

Fifth, the trading strategy used for evaluation is simplified and does not incorporate real-world constraints such as transaction costs, slippage, or risk management, which may affect the practical applicability of the results.

Sixth, another limitation of this study concerns the data splitting strategy. Although the final testing period was maintained as a strictly out-of-sample chronological evaluation, the training and validation datasets were randomly separated from the pre-test period. While this design facilitated model tuning under a consistent experimental framework, it may introduce temporal leakage due to the sequential nature of financial data. Future research should adopt rolling-origin, expanding-window, or walk-forward validation approaches to provide a more realistic evaluation of forecasting models in dynamic financial markets.

Seventh, another limitation of this study is that the experimental framework relies exclusively on historical OHLCV data and technical indicators. While this controlled feature design enables a fair comparison of different deep learning architectures, it does not capture the broader information environment increasingly used in modern financial forecasting, such as textual sentiment, macroeconomic indicators, options-derived measures, and high-frequency market microstructure data. Future research may extend the proposed framework to multimodal and cross-sectional datasets to investigate whether richer information sources alter the relative benefits of advanced architectures, including attention-based and Transformer-based models.

Eighth, another limitation of this study is the absence of formal statistical inference when comparing forecasting accuracy and trading profitability across models. Consequently, although the Seq2Seq model achieved the highest observed return on investment under the proposed evaluation framework, it cannot be concluded that the observed differences are statistically significant or exceed the variation expected by chance.

Finally, a further limitation of this study is that the comparative evaluation does not include recent Transformer-based and diffusion-based forecasting architectures. Although the selected models provide a controlled progression from feedforward networks to recurrent sequence modelling and attention mechanisms, future studies should extend the proposed evaluation framework to contemporary architectures, such as Transformers, TFT,

PatchTST, and diffusion-based models, to determine whether their increased representational capacity translates into statistically and economically meaningful improvements in financial forecasting.

5.5. Future Research Directions

Future research will extend this work in several directions. First, additional evaluation metrics and statistical validation techniques, such as bootstrap confidence intervals and Diebold–Mariano tests, will be incorporated to strengthen the robustness of the findings.

Second, the analysis will be expanded to include multiple markets and asset classes to assess generalizability. Third, more advanced architectures, including Transformer-based and diffusion-based models, will be evaluated within the proposed framework.

Fourth, future research should incorporate bootstrap confidence intervals, hypothesis-testing procedures (e.g., Diebold–Mariano tests), and risk-adjusted performance evaluation to determine whether the observed differences remain statistically robust under more rigorous experimental conditions.

Fifth, although a passive market benchmark has been incorporated for comparison, future studies should further evaluate forecasting models using additional benchmark portfolios together with risk-adjusted performance measures such as the Sharpe Ratio, Sortino Ratio, and Maximum Drawdown.

Sixth, future research should include a time-series validation. Finally, future studies will incorporate more realistic trading environments, including transaction costs, risk management strategies, and portfolio-level optimization, to better reflect real-world applications.

6. Conclusions

This study investigated the performance of multiple deep learning architectures for multi-step stock price forecasting within the ASX 100, using a unified experimental framework that integrates predictive accuracy and trading profitability. The results demonstrate that the Seq2Seq model achieved the highest observed trading profitability under the proposed evaluation framework. However, because formal statistical inference was not performed, these results should be interpreted as descriptive empirical observations rather than statistically confirmed evidence of superior model performance.

A key finding of this study is that improvements in traditional error metrics, such as MAPE, do not fully capture the practical value of forecasting models. Despite only marginal differences in forecasting accuracy, noticeable differences in trading profitability were observed across the competing models. In particular, the Seq2Seq model achieved an observed return on investment of 23.2%, compared with an approximate 14.0% return from a passive buy-and-hold investment in the S&P/ASX 200 benchmark over the same evaluation period. This comparison suggests that improvements in forecasting performance may translate into economically meaningful gains under the proposed evaluation framework. However, because formal statistical significance testing and risk-adjusted performance evaluation were not performed, these findings should be interpreted as descriptive empirical observations rather than conclusive evidence of superior trading performance. In addition, the results indicate that increased model complexity does not necessarily lead to improved performance. The attention-based model did not provide significant advantages over the standard Seq2Seq architecture, suggesting that the effectiveness of advanced mechanisms such as attention is highly context-dependent. This finding contributes to the growing body of evidence that model selection should be guided by the characteristics of the data and the application domain, rather than by architectural sophistication alone.

From a broader perspective, this study contributes to the literature by proposing a decision-oriented evaluation framework that bridges the gap between predictive modeling and financial application. The framework is model-agnostic and can be extended to incorporate more advanced architectures and richer datasets, providing a foundation for future research in financial machine learning.

The findings also have practical implications for practitioners. Simpler sequence-based models may offer a favorable balance between performance, interpretability, and computational efficiency, making them suitable for real-world deployment in resource-constrained environments. Moreover, the emphasis on trading performance highlights the importance of aligning model evaluation with end-use objectives.

However, the results should be interpreted within the context of the study's limitations, including the focus on a single market, the use of a limited feature set, and the absence of statistical validation and realistic trading constraints. Addressing these limitations represents an important direction for future research.

Future work will extend this framework across multiple markets and asset classes, incorporate more comprehensive feature sets, and evaluate advanced architectures such as Transformer-based and diffusion-based models. In addition, integrating risk-adjusted performance metrics and realistic trading assumptions will further enhance the applicability of the proposed approach.

Overall, this study highlights the importance of moving beyond purely accuracy-based evaluation toward a more comprehensive decision-oriented assessment of forecasting models. By integrating predictive accuracy with trading profitability and benchmark comparison, the proposed evaluation framework provides a more practical perspective on model performance in financial forecasting. Future research should extend this framework by incorporating formal statistical inference, risk-adjusted performance measures, and more realistic trading environments to determine whether the observed economic benefits remain robust under broader market conditions.

Author Contributions: Conceptualization; Methodology; Validation; Investigation; Data-curation; Writing—review and editing; Project Administration; Supervision, C.A.H.; Conceptualization; Methodology; Software; Formal Analysis; Writing—Original draft; Visualization, H.L.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in the study are openly available at <https://github.com/th190/ASX100> (accessed on 13 July 2026).

Acknowledgments: ChatGPT-5.5 was used to improve English grammar and spelling. We would like to thank the National University of Singapore for providing the research resources.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
2. Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014. [[CrossRef](#)]
3. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014.
4. Luong, M.T.; Pham, H.; Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015. [[CrossRef](#)]
5. Welch, I.; Goyal, A. A Comprehensive Look at The Empirical Performance of Equity Premium Prediction. *Rev. Financ. Stud.* **2008**, *21*, 1455–1508.

6. Dhar, S.; Mukherjee, T.; Ghoshal, A.K. Performance evaluation of Neural Network approach in financial prediction: Evidence from Indian Market. In Proceedings of the International Conference on Communication and Computational Intelligence (INCOCCI), Perundurai, India, 27–29 December 2010.
7. Martinez, L.C.; da Hora, D.N.; de Palotti, M.J.R.; Meira, W.; Pappa, G.L. From an artificial neural network to a stock market day-trading system: A case study on the BM&F BOVESPA. In Proceedings of the 2009 International Joint Conference on Neural Networks, Atlanta, GA, USA, 14–19 June 2009. [\[CrossRef\]](#)
8. Zhang, G.P. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing* **2003**, *50*, 159–175. [\[CrossRef\]](#)
9. Zhao, J.; Zeng, D.; Liang, S.; Kang, H.; Liu, Q. Prediction model for stock price trend based on recurrent neural network. *J. Ambient Intell. Humaniz. Comput.* **2020**, *12*, 745–753. [\[CrossRef\]](#)
10. Nelson, D.M.Q.; Pereira, A.C.M.; de Oliveira, R.A. Stock market's price movement prediction with LSTM neural networks. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017. [\[CrossRef\]](#)
11. Fischer, T.; Krauss, C. Deep learning with long short-term memory networks for financial market predictions. *Eur. J. Oper. Res.* **2017**, *240*, 654–669.
12. Harmon, M.; Klabjan, D. Dynamic Prediction Length for Time Series with Sequence-to-Sequence Networks. *arXiv* **2019**, arXiv:1807.00425v2.
13. Liu, G.; Wang, X. A numerical-based attention method for stock market prediction with dual information. *IEEE Access* **2018**, *7*, 7357–7367. [\[CrossRef\]](#)
14. Wang, X.; Cai, Z.; Luo, Y.; Wen, Z.; Ying, S. Long time series deep forecasting with multiscale feature extraction and Seq2seq attention mechanism. *Neural Process. Lett.* **2022**, *54*, 3443–3466. [\[CrossRef\]](#)
15. Dash, A.; Singh, A.; Jain, A.; Shukla, A.; Mishra, H.; Vyas, P.; Kaur, S. Stock price analysis and prediction using seq2seq lstm. In *International Conference on Data Analytics and Insights*; Springer Nature: Singapore, 2023; pp. 655–666.
16. Shi, Z.; Hu, Y.; Mo, G.; Wu, J. Attention-based CNN-LSTM and XGBoost hybrid model for stock prediction. *arXiv* **2023**, arXiv:2204.02623.
17. Abdelfattah, B.A.; Darwish, S.M.; Elkaffas, S.M. Enhancing the Prediction of Stock Market Movement Using Neutrosophic-Logic-Based Sentiment Analysis. *J. Theor. Appl. Electron. Commer. Res.* **2024**, *19*, 116–134. [\[CrossRef\]](#)
18. Gnanavel, R.; Gnanasekar, J.M. An Innovative Sentiment Influenced Stock Market Prediction Based on Dual Scale Adaptive Residual Long Short-Term Memory with Attention Mechanism. *Comput. Intell.* **2025**, *41*, e70073. [\[CrossRef\]](#)
19. Nasar, M. Forecasting Week-Ahead Closing Price of Muscat Securities Market Using Hybrid TCN-LSTM Model. 2025. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5286017 (accessed on 30 November 2025).
20. Muqtadir, A.; Nasar, M. Multimodal Deep Learning for Stock Market Forecasting in Smart FinTech Systems. 2025. Available online: <https://ieeexplore.ieee.org/abstract/document/11390091/authors#authors> (accessed on 28 February 2026).
21. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
22. Sercan, B.L.; Arik, O.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *Int. J. Forecast.* **2021**, *37*, 1748–1764. [\[CrossRef\]](#)
23. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. A Time Series Is Worth 64 Words: Long-Term Forecasting with Transformers. 2023. Available online: <https://arxiv.org/abs/2211.14730> (accessed on 5 December 2025).
24. Rasul, K.; Seward, C.; Schuster, I.; Vollgraf, R. Autoregressive Denoising Diffusion Models for Multivariate Probabilistic Time Series Forecasting. 2021. Available online: <https://www.semanticscholar.org/reader/2ec659c6699933567a0924053de48359ae0eb00e> (accessed on 7 December 2025).
25. Tashiro, Y.; Song, J.; Song, Y.; Ermon, S. CSDI: Conditional Score-Based Diffusion Models for Probabilistic Time Series Imputation. 2021. Available online: <https://arxiv.org/abs/2107.03502> (accessed on 19 December 2025).
26. Shynkevich, Y.; McGinnity, T.; Coleman, S.A.; Belatreche, A.; Li, Y. Forecasting price movements using technical indicators: Investigating the impact of varying input window length. *Neurocomputing* **2016**, *264*, 71–88. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.