Indian Stock Index Price Prediction using Deep Learning and Transformer Models

Hetvi Desai
MSc (Data Science)
DAIICT
Gandhinagar, Gujarat
202218012@daiict.ac.in

Shreya Arora
MSc (Data Science)
DAIICT
Gandhinagar, Gujarat
202218032@daiict.ac.in

Dhruv Solanki
MSc (Data Science)
DAIICT
Gandhinagar, Gujarat
202218053@daiict.ac.in

Jatan Sahu
MSc (Data Science)
DAIICT
Gandhinagar, Gujarat
202218061@daiict.ac.in

***Abstract*—In the dynamic realm of financial markets, the pursuit of accurate predictions and profound insights remains a critical challenge. This phase of the project aims to augment forecasting precision for the Indian Stock Market Index. The initial phase involved the development of an algorithm dedicated to detecting circuit occurrences in Indian stocks. Subsequently, in the second phase, attempts were made to predict circuit events using various features. Despite these efforts, the desired classification results were not achieved. To overcome this challenge, we shifted focus to constructing a model capable of accurately predicting stock prices, which could then be leveraged to forecast circuit occurrences. Conventional mathematical models proved inadequate in accurately predicting circuit prices. To address this limitation, we explored alternative approaches, including Facebook Prophet and Long Short-Term Memory (LSTM) models. While Facebook Prophet struggled to capture short-term trends, LSTM exhibited superior predictive capabilities. Building on these insights, the current Phase 3 of the project introduces the implementation of Deep Neural Networks (DNN) and Transformers. These advanced models are applied to predict the prices of key Indian Stock Market Indices such as Nifty50, Sensex and BankNifty. The objective is to refine forecasting precision and gain deeper insights into the intricate dynamics of the financial markets, ultimately contributing to more informed decision-making processes.**

***Index Terms*—Stock Market, Circuit Breakers, Stock Volatility, Deep Learning, Transformers**

I. Introduction

Stock markets have always been profitable investment options with excellent potential for wealth generation. However, stock markets are also volatile, and can experience sudden and dramatic price swings. These price swings can be caused by a variety of factors, including changes in economic conditions, news events, and investor sentiment. When stock prices experience a sudden and substantial decline, it can lead to panic selling. Panic selling is when investors sell stocks out of fear, without considering the underlying fundamentals of the company. Panic selling can quickly accelerate the decline in stock prices, leading to a market crash. The stock crashes can dislocate economic activities and disastrously affect the investors whose income and wealth are based on these financial assets.

The Nifty 50, also known simply as the Nifty, is a stock market index in India that serves as a benchmark for the country's equity market performance. The Nifty 50 index comprises 50 actively traded stocks from various sectors, representing a diversified and comprehensive view of the Indian stock market. These 50 stocks are selected based on various criteria, including liquidity, market capitalization, and sector representation. The Nifty 50 is widely used by investors, traders, and analysts as a key indicator of the overall health and direction of the Indian stock market, making it a crucial tool for financial analysis and decision-making.

The Sensex, short for the Sensitive Index, is the benchmark stock market index of the Bombay Stock Exchange (BSE) in India. It represents the performance of the 30 largest and most well-established companies listed on the BSE. These companies, often referred to as blue-chip stocks, are leaders in their respective sectors and play a crucial role in the Indian economy. The Sensex is a key indicator of the overall health and performance of the Indian stock market. Investors, analysts, and policymakers closely monitor the Sensex to gauge the sentiment and trends in the Indian equity market.

The Bank Nifty is another important stock market index in India, specifically designed to capture the performance of the banking sector. It comprises the most liquid and large capitalized banking stocks listed on the National Stock Exchange (NSE). The index includes both public and private sector banks, providing a comprehensive view of the banking industry's performance. As the banking sector is a vital component of the Indian economy, the Bank Nifty serves as a barometer for assessing the financial health and stability of the banking sector. Traders and investors often use Bank Nifty futures and options for derivative trading and hedging strategies related to banking stocks.

An increasing number of financial institutions are gravitating towards making investment decisions through the application of deep learning algorithms, surpassing reliance on human subjectivity. Given the intricate nature of financial markets, the amalgamation of deep learning techniques and financial time series forecasting stands out as one of the most compelling and appealing areas of exploration.

Numerous studies in the literature have explored the application of machine learning models such as AR, ARIMA, DNN models like LSTM, and CNN to predict stock prices, with notable success reported particularly for deep learning models. These endeavors have been instrumental in addressing the inherent limitations of traditional machine learning models. Nonetheless, even within the realm of deep learning, models such as CNN and RNN are not without their challenges. RNNs, for instance, contend with issues such as vanishing or exploding gradients, while CNNs, with their pooling layers, sacrifice information for the sake of computational efficiency.

In response to these challenges, transformers have emerged as a promising solution and have demonstrated efficacy across various domains. This project aims to leverage transformers to predict stock index prices and subsequently compare their performance with established deep learning approaches. By doing so, we seek to evaluate the effectiveness of transformers in mitigating the limitations encountered by traditional deep learning models, offering a comprehensive understanding of their applicability in financial time series forecasting.

II. Problem Statement

The challenge addressed by this project is the persistent ambiguity surrounding stock market volatility and the effectiveness of circuit breakers in managing losses during periods of unpredictable price fluctuations. Despite the successful development of a preliminary algorithm to identify circuit breaker instances, there remains a critical gap in understanding the intricate dynamics governing market volatility. Initially, our approach involved utilizing stock prices, fundamental financial ratios, and index prices to forecast circuit occurrences. Phase 2 explored machine learning algorithms for predicting circuits, treating it as a classification problem. Unfortunately, these algorithms struggled due to the significant impact of real-time market news. This realization led us to shift our focus to predicting stock prices, setting the stage for the current phase. This phase builds upon our earlier work, incorporating insights gained from phases 1 and 2. Recognizing the limitations of traditional approaches, we extend our investigation into Deep Neural Networks. In this report, we present a comprehensive comparative analysis of three distinct models: Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), and Transformer. Our objective is to decipher latent patterns within the Nifty 50, Sensex and BanNifty Indices, contributing valuable insights for investors. Our focus is on refining predictive precision for the Nifty 50 Index, combining lessons learned from previous phases with the potential of sophisticated models.

III. Literature Review

[2] In 2018, Hiransha M and others discussed the challenges of stock market forecasting using traditional linear models like AR, ARMA, and ARIMA, emphasizing their limitations in handling fluctuating and non-linear time series data. The application of deep learning models, particularly Artificial Neural Networks (ANNs), is proposed as a solution to address the complexities of stock market prediction. The success of ANNs in forecasting stems from their ability to approximate complex relationships and generalize to new test samples. The paper reviews the practical application of ANNs in stock market prediction over the past few decades and highlights a comparison study of different deep learning models. Additionally, it explores the use of Genetic Algorithms for feature discretization in ANN for stock price forecasting. The architectural aspects of ANNs, including input, hidden, and output layers, are explained, emphasizing non-linear activation functions. The study contributes insights into the effectiveness of deep learning models, particularly ANNs, in tackling the challenges of stock market forecasting.

[4] The traditional sequence transduction models rely on complex recurrent or convolutional neural networks with encoders and decoders, often incorporating attention mechanisms. In 2017, Ashish Vaswani and others introduced the Transformer, a novel architecture based solely on attention mechanisms, eliminating the need for recurrence and convolutions. Experimental results on machine translation tasks demonstrate superior quality, enhanced parallelizability, and reduced training time compared to existing models. Recurrent models, popular in sequence modeling, exhibit sequential computation limitations, hindering parallelization for longer sequences. The Transformer, devoid of recurrence, utilizes self-attention to establish global dependencies between input and output, enabling significant parallelization. The model achieves state-of-the-art translation quality after just twelve hours of training on eight P100 GPUs. The paper also outlines the Transformer's background, emphasizing the advantages of self-attention over other models. The model architecture involves stacked self-attention and fully connected layers for both encoder and decoder, facilitating residual connections for efficient information flow. All sub-layers and embedding layers produce outputs of dimension dmodel = 512 to support residual connections. This paper serves as the base for our Transformer model.

[5] Chaojie Wang and others, in 2022, provided an overview of traditional deep learning models used in stock market prediction, categorizing methods into fundamental and technical analysis. They further delved into classic models like CNN, RNN, and LSTM, setting the stage for subsequent model comparisons. Notably, the paper introduces the Transformer architecture, diverging from traditional models. Through experiments on major stock indices, the study showcases the Transformer model's superior performance. Compared to traditional models and a buy & hold strategy, the Transformer excels in prediction accuracy and net value analysis across various perspectives. This research signifies the Transformer's efficacy in enhancing stock market prediction outcomes. We intend to follow the similar approach in this ultimate phase of our project.

[1] In 2021, Francisco J. Baldán and José M. Benítez addressed the challenge of Multivariate Time Series Classification (MTSC) with a focus on interpretability. While most existing approaches prioritize accuracy, the proposed method by them aims to create intelligible classifiers by transforming the original MTSC problem into a conventional classification task. This is achieved through a feature-based representation of time series, allowing the

application of traditional classification algorithms. Unlike prevalent techniques, the emphasis here is on user-friendly interpretation, enhancing end-user confidence. The paper reviews the MTSC landscape, highlighting limitations in early distance-based algorithms and advocating for the interpretability of results. It introduces a novel approach using feature extraction, departing from prevalent methods like shapelets and bag-of-words. The proposed method shows competitive accuracy with state-of-the-art techniques while prioritizing the creation of interpretable classifiers for complex tasks. The paper contributes to advancing MTSC methodologies by addressing both accuracy and interpretability concerns.

[3] Ruiz et al discussed advancements in Time Series Classification (TSC) and Multivariate Time Series Classification (MTSC). It notes the lack of standardized test problems for MTSC, highlighting the significance of the UEA archive with 30 MTSC problems. The review focuses on bespoke MTSC algorithms, including deep learning, shapelets, and bag-of-words approaches. The authors compare these algorithms to dimension-independent approaches on 26 equal-length problems from the UEA MTSC archive. Notably, the study finds that four classifiers, particularly ROCKET, outperform the benchmark dynamic time warping algorithm significantly and with a reduced computational time. The paper emphasizes the challenges and commonality of encountering multivariate TSC problems in real-world scenarios, providing an overview of MTSC and classifier evaluations. The research contributes valuable insights into MTSC algorithm effectiveness and efficiency which can be used in the future work of this project.

IV. Background

This section describes the deep learning methods that have been used in this project to predict the price of the index.

A. LSTM

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) architecture that is particularly well-suited for sequence prediction tasks, including time series forecasting such as predicting stock prices. LSTMs are designed to capture and learn patterns in sequences, making them effective for modelling temporal dependencies in time series data. An LSTM network consists of memory cells that can store and retrieve information over long sequences. The architecture includes three gates: the input gate, forget gate, and output gate. These gates control the flow of information and help the network learn which information to remember and which to forget. This architecture is represented in Figure 1.

For a sequence $\mathbf{X} = \{x_t : t = 1, \ldots, T\}$, where $x_t \in \mathbb{R}$ denotes the stock price at time t, the update steps of LSTM can be expressed as follows:

$$\begin{aligned}
f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f), \\
i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \\
\tilde{c}_t &= \tanh(W_c \cdot [h_{t-1}, x_t] + b_c), \\
c_t &= f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t, \\
o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \\
h_t &= o_t \cdot \tanh(c_t).
\end{aligned}$$

Here, W and b are weights and bias for the corresponding connection, σ and $\tanh$ represent the sigmoid function and the hyperbolic tangent function, and $[\cdot]$ denotes the concatenation operation which merges the two vectors together.

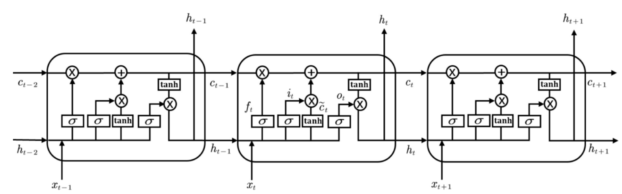

Fig. 1. Long-Short Term Memory Cells creating a Network

B. CNN

Convolutional Neural Networks (CNNs) are a class of deep learning models primarily designed for processing and analyzing visual data, such as images and videos. However, their versatility extends beyond traditional computer vision applications, and they have found utility in diverse fields, including financial time series forecasting, such as predicting Nifty 50 index prices.

In the context of predicting financial market values, CNNs can be applied to capture intricate patterns and relationships within historical price charts. The network's convolutional layers are adept at automatically learning hierarchical features, recognizing temporal dependencies, and identifying complex patterns in the time series data. The process typically involves transforming historical stock price data into a format suitable for input into a CNN. The network then learns to extract relevant features, recognizing patterns and trends that may influence future price movements. The trained CNN can subsequently be used for making predictions on unseen data.

The output z_i at position i in a convolutional layer can be expressed as follows:

$$z_i = f(x_i, w) = \sum_{j=1}^{m} x_{i+j-1} \cdot w_j + b$$

where:

- z_i is the output at position i,
- x_i is the input at position i,
- w represents the convolutional filter,
- b is the bias term.

Figure 2 illustrates the CNN architecture employed in this project. The architecture features convolutional layers

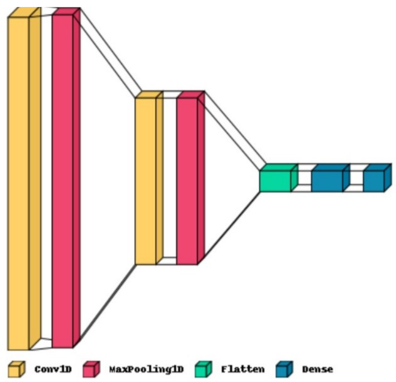

Fig. 2. CNN Architecture to predict Index Prices

(depicted in yellow) with a kernel size of 3 and a stride equal to 1. The pooling layer is represented in red. Following this, the structure is flattened to a one-dimensional format, and dense layers (represented in blue) are utilized to produce the ultimate output.

C. Transformers

Transformer methods represent a revolutionary advancement in the field of deep learning, initially introduced for natural language processing tasks. Unlike traditional sequential models such as recurrent neural networks (RNNs), transformers utilize a self-attention mechanism, allowing them to capture long-range dependencies and relationships within sequences. This architecture has proven highly effective in various domains beyond NLP, including computer vision, speech recognition, and notably, financial time series analysis.

Transformers have excelled in tasks like language translation, sentiment analysis, and document summarization due to their ability to capture contextual information effectively. In image and video analysis, transformers, especially in the form of Vision Transformers (ViTs), have shown remarkable performance in tasks such as image classification and object detection. Transformers are increasingly being employed for analyzing time series data, including financial markets. Their capacity to discern intricate patterns and dependencies makes them valuable for forecasting stock prices and understanding market trends.

The transformer's attention mechanism allows it to assign different weights to different parts of the input sequence, enabling it to focus on relevant information and capture intricate patterns. This makes transformers particularly well-suited for handling the complex and dynamic nature of financial markets.

In the context of predicting index prices, transformers bring several advantages. Transformers can capture relationships and dependencies over extended periods, enabling them to discern complex patterns in historical index price movements. They can seamlessly integrate diverse types of data, such as historical prices, trading volumes, economic indicators, and news sentiment, providing a holistic view for more accurate predictions. Transformers are adept at learning non-linear relationships, accommodating the dynamic and non-linear nature of financial markets.

To predict Index prices, self-attention mechanisms are employed to capture long-range dependencies. The attention mechanism of the transformer learning approach can be described by:

$$Attention(Q, K, V) = softmax\left(\frac{1}{\sqrt{d_k}} QK^T\right) V$$

Where:

- Q, K, V are the query, key, and value matrices, respectively.
- $softmax$ is the softmax function.
- d_k is the dimension of the key vectors.

Within the multi-head attention mechanism, each attention function operates independently in parallel with its corresponding projected versions of the query, key, and value matrices. Subsequently, the outputs of all attention functions are concatenated to form the final result, achieved through a linear layer. It can be expressed using the following expression:

$$MultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = Concat(h_1, h_2, \ldots, h_h)\mathbf{W}_O,$$

$$h_i = Attention(\mathbf{Q}\mathbf{W}_{Qi}, \mathbf{K}\mathbf{W}_{Ki}, \mathbf{V}\mathbf{W}_{Vi}),$$

where $i = 1, \ldots, h$. The weights $\mathbf{W}_{Qi}$, $\mathbf{W}_{Ki}$, $\mathbf{W}_{Vi}$, $\mathbf{W}_O$ correspond to the networks.

The Transformer architecture used has been illustrated in Figure 3. In our configuration, the multi-head attention mechanism employs four heads. Each layer within the model incorporates a dropout rate of 0.1, and the kernel size is set to 1.

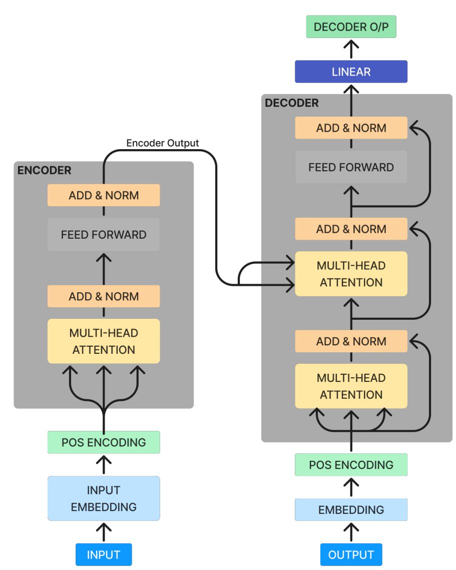

Fig. 3. Transformer Architecture for Index Price Prediction

The transformer architecture serves as a potent tool for time series forecasting, specifically tailored to capture and predict the dynamic movements of financial indices. The input to the model represents historical closing price data of the index. The input sequence undergoes initial processing through an embedding layer, followed by a positional encoding layer to retain the temporal order of the historical prices. The encoder processes this input sequence, generating a series of hidden representations and dependencies within the historical price movements. These representations are subsequently employed by the decoder to generate the predicted output sequence, representing the forecasted prices.

The decoder in this financial forecasting application utilizes the previously predicted output sequence from the preceding time step as an additional input to generate the subsequent output. This iterative process enhances the model's ability to capture temporal dependencies and trends in the price data.

To ensure stable training and optimal performance, the model incorporates residual connections and layer normalization. These mechanisms contribute to the overall effectiveness of the transformer architecture in predicting index prices, aligning with the unique challenges posed by financial time series forecasting.

V. Methodology

For this project, we model the one-dimensional time series of daily closing prices of the index over a period of 16 years from September 17, 2007 to December 8, 2023. We implement these models on three indices namely Nifty50, Sensex and BankNifty. Initially, we partition the entire data set into training and testing sets. The training set encompasses the initial 95% of the data, utilized for training model parameters, while the final 5% constitutes the testing set, employed for evaluating model performance.

To enhance data set stability and foster a robust model, we normalize the original data, which includes both the training and testing sets, as follows:

$$\hat{x}_t = \frac{x_t - \mu}{\sigma},$$

where $\hat{x}_t$ is the normalized price at time t, μ and σ are the sample mean and sample standard deviation of the training set.

The models are being evaluated on the basis of prediction errors. The true values are compared with the prediction values in the test set. The metrics used are listed below:

- Mean Absolute Error (MAE): The Mean Absolute Error is a measure of the average absolute difference between predicted and actual values.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

where $\hat{y}_i$ is the predicted value, y_i is the true value, and n is the sample size.

- Mean Squared Error (MSE): The Mean Squared Error is a measure of the average squared difference between predicted and actual values.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

where $\hat{y}_i$ is the predicted value, y_i is the true value, and n is the sample size. MSE has been used as the loss function.

- Mean Absolute Percentage Error (MAPE): The Mean Absolute Percentage Error is a measure of the average percentage difference between predicted and actual values.

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\%$$

where $\hat{y}_i$ is the predicted value, y_i is the true value, and n is the sample size.

VI. Results and Discussion

Table 1 presents the prediction errors observed in forecasting index prices. Notably, the transformer model consistently surpasses classical deep learning techniques across all three metrics for all the indices. The superior performance of the transformer model can be attributed to its ability to capture intricate patterns and long-range dependencies in the time series data. The multi-head attention mechanisms and self-attention mechanism in transformers facilitate a more comprehensive understanding of the complex relationships within the financial time series.

TABLE I
Performance Metrics for Different Models

Index	Model	MAE	MSE	MAPE (%)
Nifty50	LSTM	242.93	96836.25	1.28
	CNN	256.56	103121.44	1.36
	Transformer	240.31	91992.64	1.28
Sensex	LSTM	753.55	940670.92	1.21
	CNN	1177.05	1872449.36	1.88
	Transformer	739.97	867285.08	1.19
BankNifty	LSTM	595.36	620808.79	1.35
	CNN	747.43	913992.90	1.70
	Transformer	523.74	494531.11	1.20

In Figure 4, the fitted curves produced by CNN, LSTM, and Transformer for the Nifty 50 index are presented. It is observed that the predicted values closely align with the actual data in both the training and testing sets across all models. Notably, among the compared methods, the Transformer model exhibits the highest prediction accuracy across all data sets.

VII. Conclusion and Future Scope

This project assesses the efficacy of LSTM, CNN, and Transformer models in the context of stock market prediction, with a specific focus on experiments conducted on the stock market indices. The findings highlight the Transformer model's superior performance compared to traditional deep learning models, underscoring its potential in financial time series forecasting. The study suggests that investors stand to achieve higher excess earnings by leveraging the predictive

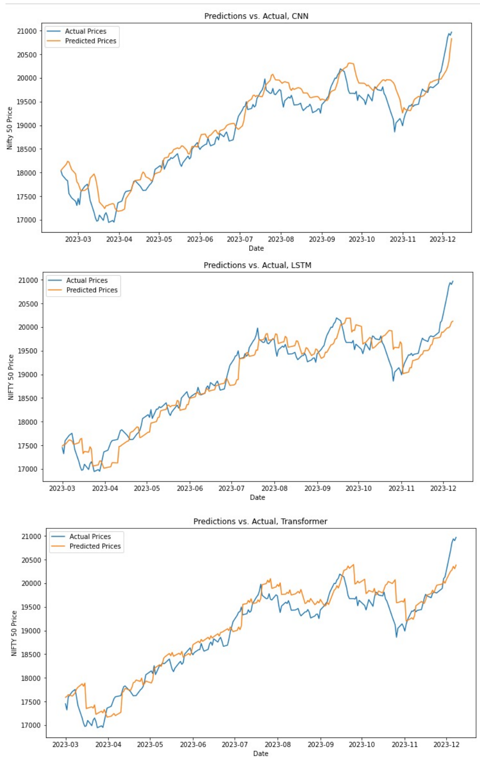

Fig. 4. The predicted curves of the CNN, LSTM and Transformer model, respectively. Here the blue lines mean the actual stock index prices data. The orange lines represent the predicted values.

capabilities of the Transformer architecture in practical scenarios.

The current study identifies certain limitations that pave the way for future research directions. The exclusive focus on predicting a single stock market index as a one-dimensional financial time series is recognized as a constraint. Recognizing that global financial markets demonstrate high correlation, future endeavors will delve into strategies for harnessing mutual information and jointly modeling high-dimensional time series data. Furthermore, the untapped potential of applying Transformer models in diverse financial markets, including commodities futures and the bond market, presents promising avenues for exploration in forthcoming research initiatives. These avenues hold the potential to broaden the understanding and applicability of Transformer architectures in the realm of financial forecasting.

Furthermore, the scope for enhancement extends to incorporating additional dimensions by integrating fundamental and sentimental analyses derived from diverse sources such as news articles, research papers, and social media posts (tweets). The inclusion of qualitative data alongside quantitative measures can potentially enrich the predictive models and offer a more comprehensive understanding of market dynamics. This expansion into multiple dimensions aligns with the evolving landscape of financial forecasting and sets the stage for a more holistic approach in future project developments.

References

[1] Francisco J. Baldán and José M. Benítez. “Multivariate times series classification through an interpretable representation”. In: *Information Sciences* 569 (2021), pp. 596–614. ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2021.05.024. URL: https://www.sciencedirect.com/science/article/pii/S0020025521004825.

[2] Hiransha M et al. “NSE Stock Market Prediction Using Deep-Learning Models”. In: *Procedia Computer Science* 132 (2018). International Conference on Computational Intelligence and Data Science, pp. 1351–1362. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2018.05.050. URL: https://www.sciencedirect.com/science/article/pii/S1877050918307828.

[3] Alejandro Pasos Ruiz et al. “The Great Multivariate Time Series Classification Bake Off: A Review and Experimental Evaluation of Recent Algorithmic Advances”. In: *Data Mining and Knowledge Discovery* 35.2 (Mar. 2021), pp. 401–449. ISSN: 1573-756X. DOI: 10.1007/s10618-020-00727-3. URL: https://doi.org/10.1007/s10618-020-00727-3.

[4] Ashish Vaswani et al. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL].

[5] Chaojie Wang et al. “Stock market index prediction using deep Transformer model”. In: *Expert Systems with Applications* 208 (2022), p. 118128. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2022.118128. URL: https://www.sciencedirect.com/science/article/pii/S0957417422013100.