Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices

Standy Hartanto[1], Alexander Agung Santoso Gunawan[2]
Computer Science Department-Master of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia[1]
Computer Science Department-School of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia[2]

***Abstract*—The stock market represents the financial pulse of economies and is an important part of the global financial system. It allows people to buy and sell shares in publicly held corporations. It serves as a platform for investors to trade ownership in businesses, enabling companies to raise capital for expansion and operations. However, the stock market can be very risky for any investor because of the fluctuating prices and uncertainties of the market. Integrating deep learning into stock market analysis enables researchers and practitioners to gain a deeper understanding of the trends and variations that will improve investment decisions. Recent advancements in the area of deep learning, more specifically with the invention of transformer-based models, have revolutionized research in stock market prediction. The Temporal Fusion Transformer (TFT) was introduced as a model that uses self-attention mechanisms to capture complex temporal dynamics across multiple time-series sequences. This study investigates feature engineering and technical data integrated into the TFT models to improve short-term stock market prediction. The Variance Inflation Factor (VIF) was used to quantify the severity of multicollinearity in the dataset. Evaluation metrics were used to evaluate TFT models' effectiveness in improving the accuracy of stock market forecasting compared to other transformer models and traditional statistical Naïve models used as baselines. The results prove that TFT models excel in forecasting by effectively identifying multiple patterns, resulting in better predictive accuracy. Furthermore, considering the unique patterns of individual stocks, TFT obtained a remarkable SMAPE of 0.0022.**

Keywords—Time series forecasting; stock price prediction; capital market; technical analysis; TFT

I. Introduction

Stock market indices show the health of the economy. It allows people to trade in the shares of publicly held corporations. It serves as a platform for investors to trade ownership in businesses, enabling companies to raise capital for expansion and operations. Changes in the equity markets will indicate the economic situation, investors' sentiment, and expectations about future economic performance. This can be very risky for any investor because of the price fluctuations and uncertainties of the market. In addition, stock markets play a vital role in determining the companies' value. Prediction of stock prices is a very complex and highly challenging task due to the intrinsic volatility and multi-dimensionality of financial markets [1]. Indeed, most traditional models are challenged to identify exactly the complex trends and variables that impact stock price movements.

Research on stock market prediction has paid considerable attention to deep learning algorithms in recent years. Some techniques involve training models using large datasets to come up with complex patterns and correlations [2, 3, 4, 5]. These may also combine other data sources, like financial and non-financial data, in a model to be trained for the increase in prediction accuracy [6]. Researchers have just commenced exploring how Transformer-based models [7, 8] apply alongside Reinforcement Learning for forecasting trends in the stock market [9, 10, 11].

TFT is a new transformer-based model for handling multiple time series sequences with complicated temporal dynamics. By merging the LSTM Sequence-to-Sequence framework with the self-attention mechanism of Transformers, TFT adeptly captures temporal dependencies across varying scales while enriching temporal representations with static contextual information about measured entities. In contrast to RNN-based models, Transformers offer expedited processing by simultaneously ingesting all input, thereby bypassing the sequential nature of RNNs. Moreover, it is easier to train Transformers because they have fewer parameters compared to LSTM networks. Transfer learning is also possible with Transformers, which is not the case with LSTM networks. Notably, TFT fits the detailed subtleties within hydrographs, peaks, and transitional phases much more effectively than both LSTM and Transformers.

Generalizing models may not provide valuable insights to analysts and investors in view of the uniqueness of trends and patterns that individual stocks exhibit to make meaningful short-term decisions. It needs more comprehensive indicators that directly impact the market behavior. This involves identifying a few exogenous input features that help the model recognize the patterns of history, evaluate its performance against real market fluctuations so that it can be agile to volatility, and thereby provide valuable insights for short-term stock price predictions.

In this research, five years of stock market data from the Indonesia Stock Exchange (IDX) were used in analyzing with the TFT model, particularly in mining, communications, and industrial sectors. The main objective is to develop a comprehensive analysis of TFT with regards to short-term stock price prediction by using historical technical indicators.

In addition, several feature engineering techniques, model architectures, and training strategies are also evaluated for their effects on the accuracy of prediction. Evaluation metrics were used against baselines composed of Transformer models and Naïve models, comparing the performance of TFT. Lastly, the model was tested under real-market conditions to evaluate its performance in generating accurate predictions on the stock exchange.

II. Related Studies

In this vast domain of time series forecasting, there exist a large number of theories and methodologies that act as the foundation for predictive analysis in different fields. These models provide insight into future trends and events that range from classic statistical approaches to modern deep learning algorithms [12].

Naïve models are very simple, fairly easy to use, and provide a simple starting point in developing or predicting stock prices. They are quick to calculate and inform us about the performance of a more complex method compared to something quite simple. While being relatively interpretable and tolerant of noise, they may miss small details that drive stock prices [13]. In this Naïve approach, each estimate is set based on the last observed value.

$$\hat{y}_{T+h|T} = y_T \tag{1}$$

Transformers have gained prominence in Natural Language Processing (NLP) and computer vision, their application in the realm of time-series data remains relatively unexplored. Our approach addresses this gap through a self-attention mechanism that helps identify complex nonlinear trends and intrinsic dynamics in time series data, which are consolidated under high volatility and nonlinearity. The predictive power of our model includes providing closing price forecasts for the next trading day with insights derived from multiple stock price inputs. Our model is rigorously validated by testing through four different error evaluation metrics. The fact that our model can predict the closing prices with a probability of more than 90% makes this model very useful for fintech [14].

Employing a combination of CNNs, RNNs, LSTMs, and BERT, alongside textual data from social media. It is posited that, by incorporating deep learning models with the state-of-the-art BERT word embedding model, classification performance will be improved. When such deep learning algorithms are combined with such a state-of-the-art natural language processing model, it incurs improvement in performance every time. In predicting stock directional movement, it leads to up to 96.26% accuracy performance [15].

LSTM neural network models are suitable for monitoring trends and capturing seasonality over long forecast periods. A study [16] reveals an increase in model performance with a new approach that uses six variables: High, Low, Open, Volume, HiLo, and OpSe. Give rise to the urge to explore new forecasting strategies with respect to the various scenarios that can be studied. These efforts can provide meaningful insights for investors and analysts who want to understand the working mechanisms of the stock market to better grasp future trends [17].

Recently, studies on the application of Transformer-based models and Reinforcement Learning (RL) models in stock market forecasting have already been initiated. The purpose of the survey is to consolidate the latest developments in methodologies like Transformers and RL with in-depth analysis and discourse on their implications and advancement in this domain [9, 10].

Temporal Fusion Transformer is a model architecture designed for time series forecasting. It intrinsically combines the concepts of transformers very successfully in natural language processing and related sequence data tasks with techniques specifically developed for dealing with temporal data [18, 19, 20]. The primary function of TFT is to enhance learnt temporal representations with static data about measured entities and to capture temporal dependencies at various time scales using a combination of the Transformer's Self Attention mechanism and the LSTM Sequence-to-Sequence [21]. Transformers process all of the input at once, making them faster than RNN-based models [22]. Compared to transformer networks, LSTM networks require longer training due to their significantly larger parameter set. Furthermore, transfer learning is not feasible with LSTM networks. TFT is more effective than LSTM or Transformers at capturing the subtleties of the hydrographs, such as the peaks and limbs.

A study in [21] proposed the TFT model as a solution for multi-horizon time series forecasting, where the goal is to predict multiple future time steps of a sequence simultaneously. The TFT architecture generalizes transformer attention mechanisms and encoder-decoder structures to capture complex temporal patterns in the data while offering interpretability through attention weights. This study encompasses data from four major categories: Electricity, Traffic, Retail, and Volatility. The regional variables in the volatility category are indices of the Americas, Europe, or Asia. There are 31 stock indices in total with open-to-close returns acting as supplementary exogenous inputs, and the time span was from 2000 to 2019. Comparisons against TFT were made with a number of models DeepAR, ConTrans, Seq2Seq, and MQRNN with respective results including 0.050, 0.047, 0.042, 0.042, and 0.039.

Acknowledging the distinct trends and patterns observed in individual stocks, the study aims to evaluate the effectiveness of the TFT model in analyzing short-term trends in Indonesian stock prices, particularly within the mining, communications, and industrial sectors. It seeks to determine whether TFT models can provide valuable insights to analysts and investors for short-term decision-making purposes.

III. Research Methodology

Advanced methodology that drives our research is unveiled. It is imperative to establish a nuanced understanding of what kind of guiding principles and meticulous procedures we put in motion for an in-depth review, starting from careful data collection to rigorous analysis. Fig. 1 depicts the research stages.

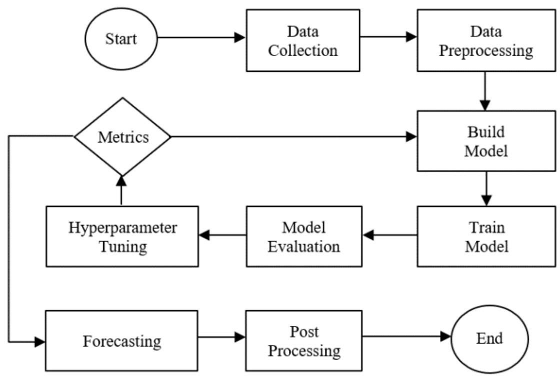

Fig. 1. Research stages.

A. Dataset

The landscape of this research utilized datasets sourced from emerging markets, the Indonesia Stock Exchange (IDX), in order to reflect a growing recognition of the significance of diversifying data sources to attain a more comprehensive understanding of global economic trends. This approach acknowledges the high probability of new trends, market behaviors, and investment patterns that might not be captured or be sufficiently represented within past research datasets.

The dataset is sourced from publicly available data provided by Yahoo!Finance [23]. General market stability and behavior will drive data requirements, in highly volatile markets or in exceptional circumstances such as an economic crisis, longer historical data may be required. However, longer periods of data might increase the risk of overfitting, where the model learns noise in the data rather than true patterns. The investigation contemplated the incorporation of data covering roughly the previous five years.

Each stock exhibits a unique pattern. To acquire comprehensive insights, three different stock analyses were made for Aneka Tambang (ANTM) in the mining sector, XL Axiata (EXCL) in the communication sector, and Astra International (ASII) in the industrial sector. The transaction was specified for the date range from March 27, 2019, to March 27, 2024.

Table I presents a selection of five examples taken from the ANTM stock dataset, showing which main features are detected on each trading day. The main variables regarding the dataset include Close and Volume.

TABLE I. SUMMARY OF STOCK EXTRACTION DATA

Date	Close	Volume
2024-03-15	1490	51840100
2024-03-18	1526	66602000
2024-03-19	1531	49266200
2024-03-20	1531	38425600
2024-03-21	1568	85481500

"Close" represents the price level at which the last trade took place when the market closed for the day. "Volume" shows the degree of activity or liquidity for that stock on the market.

B. Data Preprocessing

In this study, historical stock data was retrieved from Yahoo!Finance. After acquiring the data, some of the columns were removed as they were irrelevant for the analysis. Sorting the dataset in order of date to keep chronological order is a very critical factor in almost all forms of time series analysis. Adding more features to a model is one step toward better capturing the nuanced relationships and dependencies existing within financial markets and hence leads to more accurate and robust predictions of stock prices from the model. Other variables that were included in this dataset to enhance the predictive power of the model were the gap between the opening and closing prices and indicators for working days and months. These new variables provide insight into temporal trends and behavior of the markets, which could not have been done otherwise, and thus facilitate better predictions.

- Gap between opening and closing price: The model will understand days in which prices move highly compared with days when prices remain unchanged. This will help the model understand short-term price trend predictions.
- Working days and months: These variables enabled the model to accommodate known breaks or closures within the markets and seasonal adjustments in demand.

Despite TFT's capability to manage multicollinearity, VIF was utilized in this study to converge the results under the statistical requirements. The VIF was calculated for each predictor variable to measure the degree of multicollinearity. VIF refers to the measure of how much multicollinearity inflates the variance of a regression coefficient. Table II presents the results of VIF.

TABLE II. EVALUATION OF MULTICOLLINEARITY

Variables	Variance Inflation Factor (VIF)		
	ANTM	EXCL	ASII
Close	4.186406	5.989167	6.953210
Volume	2.413983	2.115669	3.727508
Gap_Open_Close	2.784901	2.486082	2.675353
Months	2.641024	3.239180	3.317118
Working_Days	2.482500	2.854557	2.880108

C. Proposed Method

In this section, a discussion and description regarding several deep learning methodologies are presented, followed by careful integration of these methods into the proposed model architecture.

Attention mechanisms are key components that allow the model to selectively focus on different segments in the input sequence while processing the temporal data for purposes of forecasting. Attentional mechanisms thus play a very important role in capturing complex temporal patterns, especially temporal interdependencies across a variety of time steps.

Recurrent Neural Network (RNN) stands as a deep learning model designed to process and transform sequential data inputs into specific sequential data outputs. Such sequential data typically encompasses words, sentences, or time-series data, where sequential elements are interconnected through complex semantic and syntactic rules.

Long Short-Term Memory (LSTM) is one such subtype of RNN, it is applied to sequence data to identify any underlying patterns within it. There may be present sequence data in the form of sensor readings, stock prices, or natural language. All these, while taking the position in the sequence of not only the actual value into account, are obtained during the prediction phase.

The Transformer, a deep learning architecture reliant on attention mechanisms [24], distinguishes itself by necessitating shorter training times compared to preceding recurrent neural architectures like LSTM. More precisely, this model accepts tokenized input tokens and, at each layer, contextualizes each token concurrently with other input tokens through attention mechanisms. Through their self-attention mechanisms, these models adeptly discern patterns spanning extensive sequences, effectively weighing the significance of each time step for accurate predictions. Parallel processing capabilities of Transformers expedite training and inference, useful for long time series. Moreover, by construction, Transformers inherently learn meaningful features from data, avoiding thorough manual feature engineering. By design ready to scale up and adapt, the Transformers are tailored to decode complex relationships in time and positions them as very powerful tools to uncover insights and predict trends within time series data.

The Temporal Fusion Transformer (TFT) represents a transformer-derived model utilizing self-attention mechanisms to grasp the intricate temporal variations across multiple time sequences. It stands as a potent tool for addressing multi-horizon and multivariate time series forecasting scenarios.

TFT uses time-dependent exogenous input features, which are made up of apriori unknown inputs (z) and known inputs (x), as well as static covariates (s), which offer contextual metadata about measured entities that is independent of time, to predict the future. Past target values (y) within a look-back window of length k are used as input. TFT uses quantiles to output prediction intervals rather than just a single value. At time t, every quantile q forecast of τ-step-ahead is expressed as follows:

$$\hat{y}_i(q,t,\tau) = f_q(\tau,\ y_{i,t-k:t},\ z_{i,t-k:t},\ x_{i,t-k:t+\tau},\ s_i) \quad (2)$$

Where, q: quantile, $y_{i,t-k:t}$: historical target values, $z_{i,t-k:t}$: unknown inputs, $x_{i,t-k:t+\tau}$: known inputs, s_i: static covariates.

The proposed method is visualized in Fig. 2.

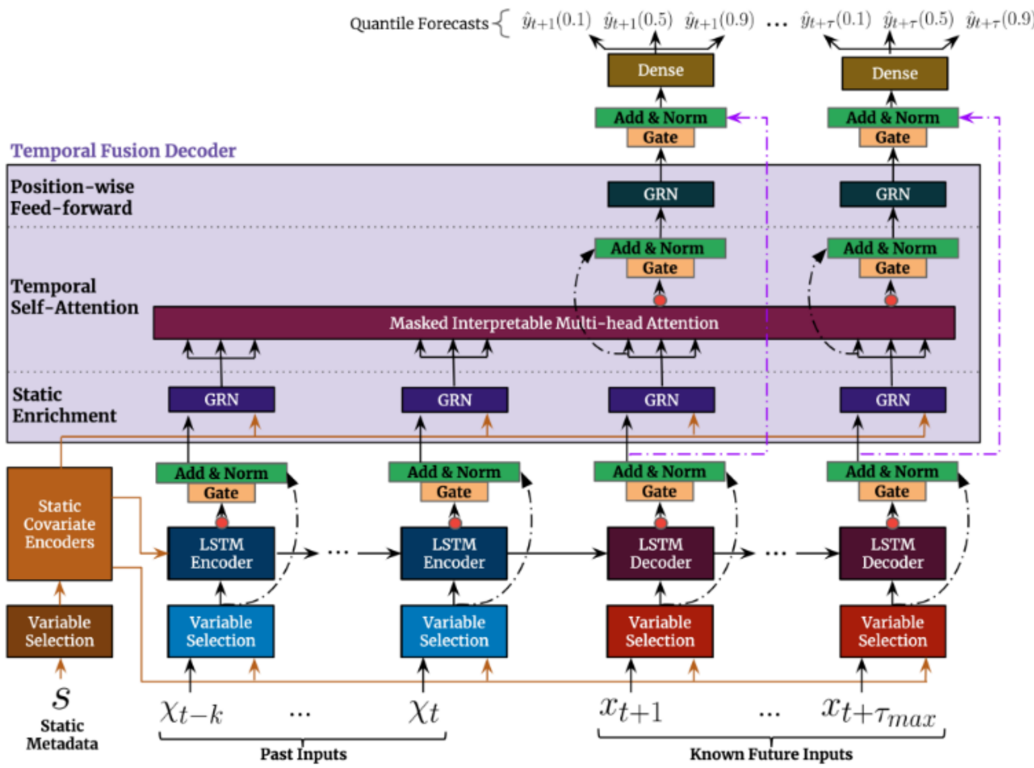

Fig. 2. The TFT Architecture [21].

To improve the flexibility of the TFT architecture, Gated Residual Networks (GRN) are incorporated into several layers of the architecture. They accomplish this by adding skip/residual connections, which transfer a layer's output to higher, non-adjacent levels in the network. As a result, the model has the ability to identify superfluous non-linear processing layers and exclude them. GRN dramatically lowers the number of parameters and processes needed while enhancing the model's generalization capabilities across a variety of application contexts. Fig. 3 illustrates the GRN architecture. ELU stands for Exponential Linear Unit activation function.

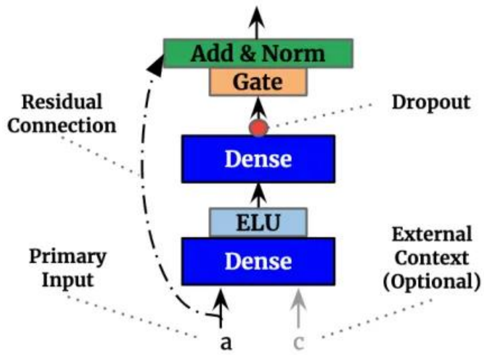

Fig. 3. Gated Residual Network [21].

In order to enable temporal variable selection, local temporal representation processing in the Sequence-to-Sequence layer, and static temporal representation enrichment, static covariate encoders obtain context vectors from static metadata and embed them into various TFT network segments. The conditioning of temporal representation learning with static data is made possible by this integration.

A different variable selection block is constructed for every type of input in the variable selection network, which includes static covariates, past inputs (both known and unknown that vary over time), and known future inputs. By learning to assess the importance of every input feature, these blocks allow the Sequence-to-Sequence layer that follows to handle the reweighted sums of the transformed inputs at each time step. Learned linear transformations of continuous data and entity embeddings of categorical features are examples of transformed inputs. Thus, the variable selection block of static covariates omits the external context vector, which is obtained from the output of the static covariate encoder block. Fig. 4 illustrates the Variable Selection architecture.

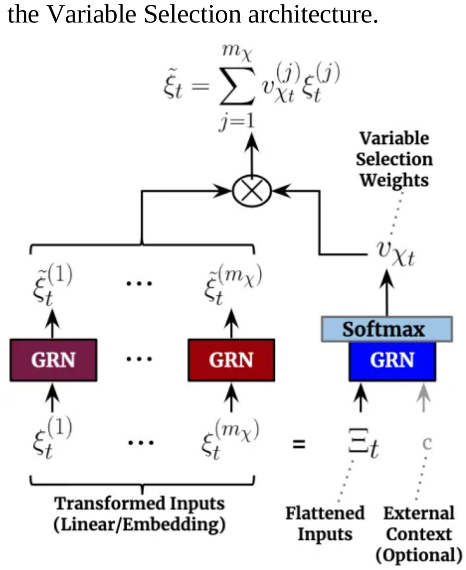

Fig. 4. Variable Selection Network [21].

The TFT network substitutes a Sequence-to-Sequence layer for the positional encoding commonly found in Transformers in the Sequence-to-Sequence component. Due to its ability to capture local temporal trends through recurrent connections, this adaptation is more suited for time series data. This block uses context vectors to initialize the first LSTM unit's cell state and concealed state. Additionally, they add to the static enrichment layer by adding static data to the temporal representation that was learned from the Sequence-to-Sequence layer.

Value relevance is evaluated by the Interpretable Multi-head attention mechanism on the basis of the connections between keys and queries. It works similarly to information retrieval in that it finds the most pertinent documents (values) by comparing a search query (query) to document embeddings (keys) [25]. Fig. 5 shows the adjustments made by the TFT to ensure interpretation. Instead, it shares many head-specific weights for values across all the attention heads.

$$InterpretableMultiHead\ (Q,K,V) = \frac{1}{h}\sum_{i=1}^{h} head_i W_H$$

$$where\ head_i = Attention\left(QW_Q^{(i)}, KW_K^{(i)}, VW_V\right) \quad (3)$$

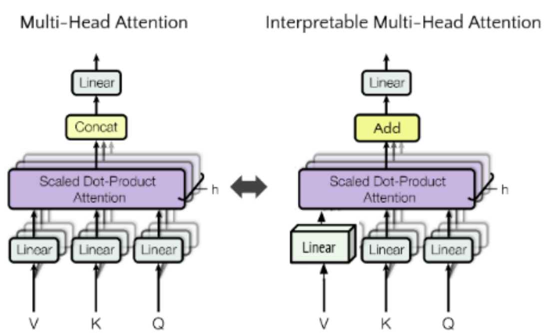

Fig. 5. Interpretable Multi-Head Attention [25].

Using a combination of the Transformer's Self Attention mechanism and the LSTM Sequence-to-Sequence, TFT was utilized to augment learnt temporal representations with static data about measured entities and to capture temporal dependencies at various time scales. In this study, a historical data window size of 12 was employed, with a prediction horizon of 3 for forecasting stock prices.

Table III presents a comprehensive outline of the TFT method utilized for forecasting stock prices.

TABLE III. TFT ALGORITHM

Algorithm 1: TFT	
Input	: Dataset [Close, Volume, GapOpenClose, Month, Day]
Output	: Prediction Result [Closing Price]
1.	**Start**:
2.	Load the dataset
3.	Preprocess the dataset
4.	Split dataset into data(train), data(val)
5.	WS = Initialize window size
6.	H = Initialize horizon
7.	Model ← build_model(TFT)
8.	Model ← train_model(data(train))
9.	Model ← optimize_hyperparameters(data(val))
10.	Model ← evaluate the model's performance
11.	Model ← save the best model
12.	MAE, MAPE, SMAPE ← (Model, data(val))
13.	Prediction ← (Model(WS,H), data(val))
14.	**Return Prediction**

D. Evaluation Metrics

This research incorporates prevalent loss functions for time-series forecasting, MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), and SMAPE (Symmetric Mean Absolute Percentage Error). The respective equations for each loss function are computed as follows:

- MAE measures the average absolute difference between the predicted values and the actual values.

$$MAE = \frac{1}{N}\sum_{i}^{N}|y_i - \hat{y}_i| \tag{4}$$

- where, N: number of observation, y_i: the actual value of the ith observation, $\hat{y}_i$: the predicted value of the ith observation.

- MAPE measures the average absolute percentage difference between actual and predicted values [26].

$$MAPE = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{A_t - F_t}{A_t}\right| \tag{5}$$

- where, N is the number of data points, At and F_t denote the actual and forecast values at data point $_t$, respectively.

- SMAPE calculates the percentage error for each data point, but it takes into account the scale of the actual and forecasted values by using their average.

$$SMAPE = \frac{100\%}{n}\sum_{t=1}^{n}\frac{|F_t - A_t|}{(|A_t|+|F_t|)/2} \tag{6}$$

- where, n is the number of data points, F_t is the forecasted value, A_t is the actual value.

E. Training Procedure and Computational Cost

Data splitting was carried out based on window size and horizon. It commences by ordering the dataset based on transaction dates. Subsequently, the dataset undergoes segmentation into training and validation sets. Spanning 5 years, the dataset comprises 1227 rows per individual stock, divided into 90% for training and 10% for validation.

It used a window size of 12 days, refers to the number of prior time steps in consideration while predicting future time steps. The horizon parameter was set to three days, and its meaning was how far the forecasting horizon was to be projected into the future.

Close, Volume, and GapOpenClose were used as exogenous inputs, complemented by working days and months as known categoricals, which are indispensable for predicting closing price as the output target. The training and validation processes were executed on a computer equipped with a 2.3 GHz Intel Core i7 quad-core CPU and 16GB of RAM. It was estimated that each of the individual final models would complete training in less than 30 minutes and use approximately 89% of the CPU's computational resources. Variability of patterns between the different stocks posed a challenge because optimality in hyperparameters configuration had to be identified. Moreover, the extended duration of model training posed a significant obstacle.

IV. Result and Discussion

Based on research findings, TFT models have been very effective in problems of time series forecasting, especially in short-term stock price prediction. It has been shown that the model is capable of handling complicated and dynamic temporal patterns in stock price data [21], drawing from information in multiple variables including seasonality. The superiority of TFT is further manifested in its flexibility when trends change. Compared with Transformer models and Naïve models, the TFT models perform better and provide more accurate predictions. TFT is an effective and sophisticated way to increase accuracy and precision in time series forecasting analysis [21].

While TFT has demonstrated significant effectiveness in resolving time series forecasting issues, it is necessary to admit that some element of uncertainty always remains in the stock market. The dynamics of the market may alter due to some unpredictable events, sudden economic changes, or other external events that may remain hidden in historical data alone [27]. Other complementary strategies would be the incorporation of real-time market sentiment analysis, macroeconomic indicators, or geopolitical events into the model in order to increase its performance [28, 29]. This would yield an all-inclusive view of dynamic market conditions to TFT and help in making better decisions due to the uncertainties that characterize changes in stock prices. Table IV presents the comparison metrics used for TFT, Transformer, and Naïve models.

TABLE IV. PERFORMANCE EVALUATION METRICS

Ticker	Model	Evaluation Metrics		
		MAE	***MAPE***	***SMAPE***
ANTM	TFT	3.3324	0.0022	0.0022
	Transformer	43.6452	0.0289	2.8845
	Naïve	35.0000	2.1027	2.0805
EXCL	TFT	9.7546	0.0041	0.0041
	Transformer	57.3425	0.0265	2.6567
	Naïve	115.4754	4.8732	4.7502
ASII	TFT	38.2241	0.0078	0.0078
	Transformer	84.3230	0.0167	1.6806
	Naïve	125.2116	2.5817	2.6285

The TFT models use multivariate data: Close, Volume, GapOpenClose, Month, Day. At the same time, Transformer and Naïve models use univariate data: Close. Based on the above-presented evaluation metrics, TFT models significantly outperform Transformer and Naïve models.

The closing price informs about performance and price trends, the volume conveys relevant information about trading activity. GapOpenClose allows highlighting of days with large price fluctuations, Time_Idx puts information into the context of time, while months and working days serve to enable the model to capture seasonal and daily trends. Based on several effective variables, the TFT model has proved to be very effective in predicting short-term stock prices, as it indeed picked up complex patterns hidden within the stock price time series data.

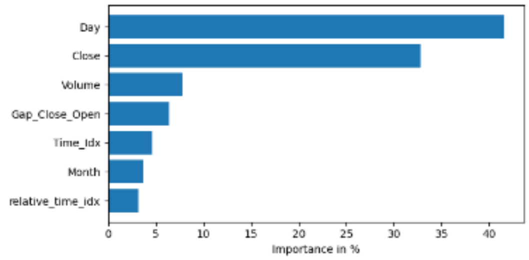

Fig. 6. ANTM encoder variables importance.

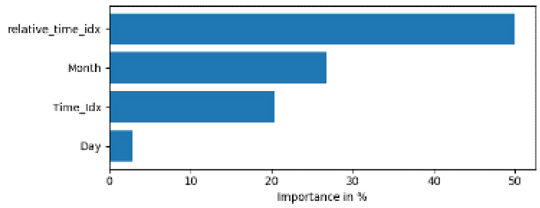

Fig. 7. ANTM decoder variables importance.

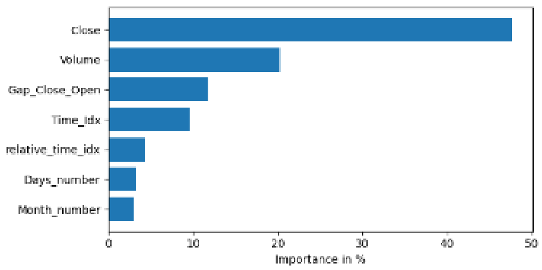

Fig. 8. EXCL encoder variables importance.

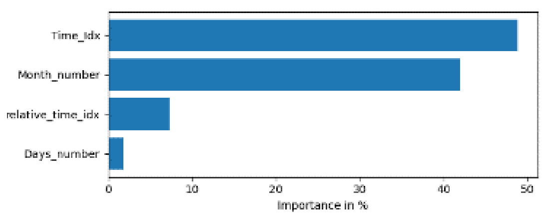

Fig. 9. EXCL decoder variables importance.

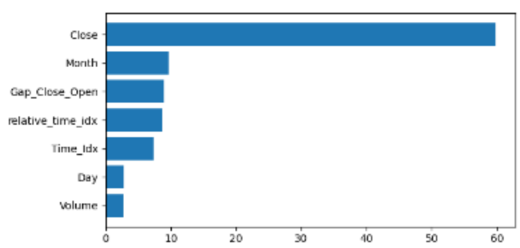

Fig. 10. ASII encoder variables importance.

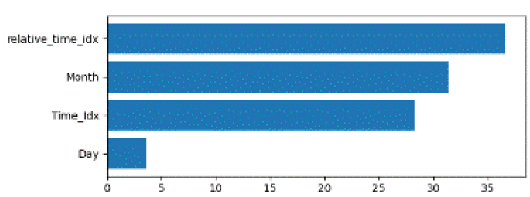

Fig. 11. ASII decoder variables importance.

Fig. 6 to Fig. 11 present evaluations of variables importance used in the encoder and decoder of the TFT models for forecasting ANTM, EXCL and ASII stocks. Encoder variables importance refers to how influential or informative the input variables are in the prediction task. It measures the effect of these variables on how well the model can capture and understand the patterns of data during it's encode phase. Decoder variables importance refers to the relevance of different features used during decoding.

Since every stock trend and patterns are different, the importance of encoder and decoder variables underline different priorities for each individual stock. This underlines the fact that customized approaches must be addressed for each individual stock.

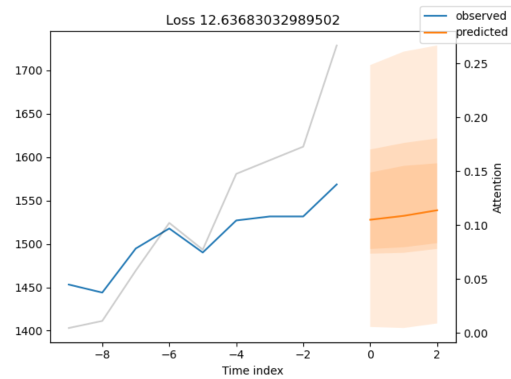

Fig. 12. ANTM prediction results.

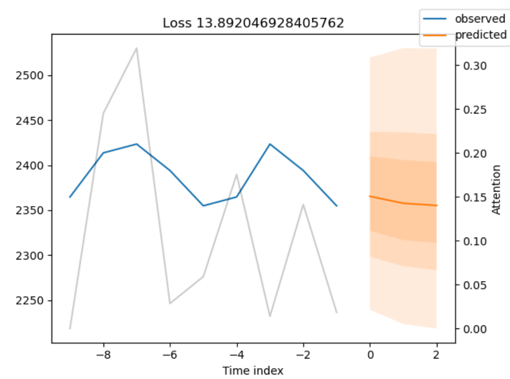

Fig. 13. EXCL prediction results.

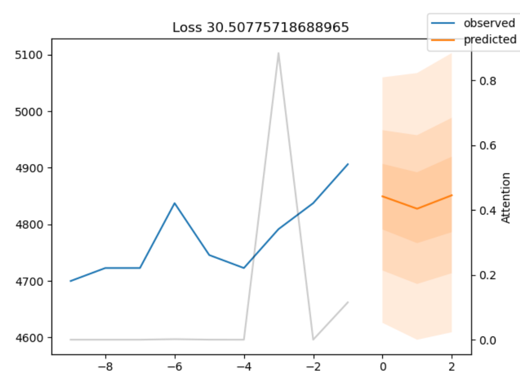

Fig. 14. ASII prediction results.

Fig. 12, 13, and 14 illustrates the prediction results within a real-world market scenario. The aim was to forecast three days ahead without having future data. The TFT model was employed to generate market predictions as of March 27, 2024. These predictions were subsequently compared with actual stock price movements observed at a later date.

The time index represents transaction dates in the dataset, numbers -6 to -1 denote historical dates given to the model, while 0, 1, and 2 are the predicted dates. Specifically, -1 corresponds to March 27, 2024 (the last date in the dataset), 0 corresponds to March 28, 2024, 1 corresponds to April 01, 2024, and 2 corresponds to April 02, 2024.

Grey lines in the plot represent the attention weights to understand the temporal patterns across past time steps. Observed line denotes the amount of attention the model pays to different points in time when making the prediction. Predicted line is an extrapolation, it refers to estimating an unknown value based on extending a known sequence of values or facts. Deviation in the prediction area is calculated using Quantile Loss, with output size=7.

$$\text{QuantileLoss(pred, outcome)} = \max\{q(\text{pred-outcome}),(q-1)(\text{pred-outcome})\} \tag{7}$$

Following a comprehensive evaluation three days later, the TFT model has demonstrated exemplary performance by accurately mirroring real-world stock movements in subsequent days. A key differentiator is the TFT model's ability to adapt to stock volatility, a capability that the Transformer and Naïve models lack.

The approach enables the TFT model to recognize the repeated patterns in closing prices at different time frames, which include trends, cycles, and seasonal variations. This aids in deciding when to make entries and exits for investments. Furthermore, the TFT model uses the opening and closing price differences to identify the patterns that may indicate reversals or continuations in markets. Other contributors to the sentiment analysis include day-of-the-week and month effects. For instance, due to weekend outlooks, the market sentiment is usually optimistic on Fridays, or cautious on the first trading day of the month since there are economic data releases.

This work has emphasized that the TFT model is very effective at capturing temporal patterns. From an application perspective, the TFT model represents one of the more advanced tools available for development in pattern recognition and predictive modeling tools, providing investors and analysts with increased and empowered analytical skills in terms of spanning market dynamics, and making fully informed and reasoned short-term decisions.

V. Conclusion

In this study, we proposed a TFT model for stock price prediction by employing multiple variables to find the influence of each variable on stock price prediction. This approach achieved an outstanding MAPE score of 0.0022. Additionally, the TFT architecture is also applied to detect sudden fluctuations in stock markets, as can be seen from the results. Nevertheless, these fluctuations may not consistently manifest at regular intervals or adhere to identical cycles on each occasion. It is imperative to acknowledge the inherent unpredictability inherent in stock market dynamics. Future research aims to investigate the integration of emerging technologies, such as reinforcement learning, with the objective of augmenting the model's robustness and efficacy in discerning intricate and dynamic patterns.

References

[1] K. Biriukova and A. Bhattacherjee, "Using transformer models for stock market anomaly detection," Creative Commons, vol. 2023, 2023.

[2] T. Muhammad et al., "Transformer-based deep learning model for stock price prediction: A case study on bangladesh stock market," International Journal of Computational Intelligence and Applications, vol. 22, pp. 1-16, 2022. arXiv:2208.08300.

[3] S. Mukherjee, B. Sadhukhan, N. Sarkar, D. Roy, and S. De, "Stock market prediction using deep learning algorithms," CAAI Transactions on Intelligence Technology, vol. 8, pp. 82-94, 2021. doi:10.1049/cit2.12059.

[4] Y. Huang, L. F. Capretz, and D. Ho, "Machine learning for stock prediction based on fundamental analysis," IEEE, 2021. doi:10.1109/SSCI50451.2021.9660134.

[5] S. Lai, M. Wang, S. Zhao, and G. R. Arce, "Predicting high-frequency stock movement with differential transformer neural network," Electronics, vol. 12, pp. 2943, 2023. doi:10.3390/electronics12132943.

[6] K. R. Dahal et al., "A comparative study on effect of news sentiment on stock price prediction with deep learning architecture," PLoS ONE, vol. 18, pp. 1-19, 2023. doi:10.1371/journal.pone.0284695.

[7] M. Paivarinta and L. A. Esteban, "Transformer-based deep learning model for stock return forecasting: Empirical evidence from US markets in 2012–2021," Turun Yliopisto, 2022.

[8] C. Li and G. Qian, "Stock price prediction using a frequency decomposition based GRU transformer neural network," Applied Sciences, vol. 13, pp. 1-18, 2023. doi:10.3390/app13010222.

[9] B. Lim and S. Zohren, "Time series forecasting with deep learning: A survey," Royal Society, vol. 379, 2020. arXiv:2004.13408.

[10] J. Zou et al., "Stock market prediction via deep learning techniques: A survey," Association for Computing Machinery, pp. 1-35, 2023. arXiv:2212.12717.

[11] J. Sen et al., "Automated stock trading framework using reinforcement learning," 2023. doi:10.13140/RG.2.2.16321.12640/1.

[12] S. Elsayed, D. Thyssens, A. Rashed, H. S. Jomaa, and L. Schmidt-Thieme, "Do we really need deep learning models for time series forecasting?," 2021. arXiv:2101.02118.

[13] B. D. Ripley, "Naive time series forecasting methods," R News, vol. 2, pp. 7-10, 2002.

[14] N. Malibari, I. Katib, and R. Mehmood, "Predicting stock closing prices in emerging markets with transformer neural networks: The Saudi stock exchange case," International Journal of Advanced Computer Science and Applications, vol. 12, pp. 876-886, 2021.

[15] D. Othan, Z. H. Kilimci, and M. Uysal, "Financial sentiment analysis for predicting direction of stocks using bidirectional encoder representations from transformers (BERT) and deep learning models," International Conference on Innovative & Intelligent Technologies, vol. 19, pp. 30-34, 2019. doi:10.17758/URUAE8.UL12191013.

[16] K. Alkhatib, H. Khazaleh, H. A. Alkhazaleh, A. R. Alsoud, and L. Abualigah, "A new stock price forecasting method using active deep learning approach," Elsevier, vol. 8, pp. 96, 2023. doi:10.3390/joitmc8020096.

[17] R. Zhang, "LSTM-based stock prediction modeling and analysis," Atlantis Press, vol. 211, pp. 2537-2542, 2022.

[18] Z. Lin, "Comparative study of LSTM and transformer for a-share stock price prediction," in Proceedings of the 2023 2nd International Conference on Artificial Intelligence, Internet and Digital Economy (ICAID 2023), pp. 72-82, 2023. doi:10.2991/978-94-6463-222-4_7.

[19] T. S. Mian, "Evaluation of stock closing prices using transformer learning," Engineering, Technology & Applied Science Research, vol. 13, pp. 11635-11642, 2023. doi:10.48084/etasr.6017.

[20] Q. Wang and Y. Yuan, "Stock price forecast: Comparison of LSTM, HMM, and transformer," in Proceedings of the 2nd International Academic Conference on Blockchain, Information Technology and Smart Finance (ICBIS 2023), pp. 126-136, 2023. doi:10.2991/978-94-6463-198-2_15.

[21] B. Lim, S. O. Arik, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," 2020. arXiv:1912.09363.

[22] H. Kaeley, Y. Qiao, and N. Bagherzadeh, "Support for stock trend prediction using transformers and sentiment analysis," IISES, 2023. arXiv:2305.14368.

[23] Yahoo!YF, https://finance.yahoo.com (accessed Mar. 28, 2024).

[24] D. Soydaner, "Attention mechanism in neural networks: Where it comes and where it goes," Neural Computing and Applications, vol. 34, pp. 13371-13385, 2022. arXiv:2204.13154.

[25] A. Vaswani et al., "Attention is all you need," in NIPS, 2017. arXiv:1706.03762.

[26] S. Kim and H. Kim, "A new metric of absolute percentage error for intermittent deman forecasts," Elsevier, vol. 32, pp. 669-679, 2016. doi:10.1016/j.ijforecast.2015.12.003.

[27] T. H. H. Aldhyani and A. Alzahrani, "Framework for predicting and modeling stock market prices based on deep learning algorithms," Electronics, vol. 11, pp. 3149, 2022. doi:10.3390/electronics11193149.

[28] A. Lopez-Lira and Y. Tang, "Can chatGPT forecast stock price movements? Return predictability and large language models," SSRN, pp. 1-69, 2024. doi:10.2139/ssrn.4412788.

[29] Y. Li, S. Lv, X. Liu, and Q. Zhang, "Incorporating transformers and attention networks for stock movement prediction," Wiley, vol. 2022, pp. 1-10, 2022. doi:10.1155/2022/7739087.