Prediction of Currency Exchange Rate Based on Transformer

Lu Zhao

A project report submitted to the Auckland University of Technology

in partial fulfillment of the requirements for the degree of

Master of Computer and Information Sciences (MCIS)

2024

School of Engineering, Computer & Mathematical Sciences

Abstract

The currency exchange rate is a crucial link between the economic and trade activities of all countries. With increasing volatility, exchange rate fluctuations have become frequent under the combined effects of global economic uncertainty and political risks. Consequently, accurate exchange rate prediction is significant in managing financial risks and economic instability. However, conventional time series models cannot efficiently predict the complex and variable nature of exchange rates. In recent years, the Transformer models have achieved outstanding performance in natural language processing and computer vision, and has also attracted attention in the field of time series analysis. Transformer models such as Informer and TFT (Temporal Fusion Transformer), have also been extensively studied.

In this project, we evaluate the performance of the Transformer, Informer, and TFT models based on our four exchange rate datasets: NZD/USD, NZD/CNY, NZD/GBP, and NZD/AUD. The results indicate that the TFT model has achieved the highest accuracy in exchange rate prediction, with an R^2 value of up to 0.94 and the lowest RMSE and MAE errors. However, the Informer model offers faster training and convergence speeds than the TFT and Transformer, making it more efficient. Furthermore, our experiments on the TFT model demonstrate that integrating the VIX index can enhance the accuracy of exchange rate predictions.

Keywords: Transformer, Informer, TFT, Currency exchange rate

Table of Contents

List of Figures

List of Tables

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Signature: Lu Zhao Date: 01 June 2024

Acknowledgement

Firstly, I am deeply grateful to my supervisor Wei Qi Yan. Dr. Yan provided a wealth of professional advice and guidance in my project and played the most crucial role in this completion. Additionally, he consistently delivered weekly lectures on the latest theoretical models in research, enabling us to understand deep learning methods better and facilitating our experimental work.

Secondly, I would like to thank my family for their immense encouragement and care, allowing me to pursue my master's studies peacefully while working a full-time job.

Lastly, I sincerely appreciate AUT for providing me with sufficient academic resources and a platform to complete this research project successfully.

Lu Zhao

Auckland, New Zealand

June 2024

Chapter 1
Introduction

This chapter consists of five parts. It starts with an introduction to the relevant background and the motivation behind the project, followed by the research question and the project's contributions. Lastly, it outlines the purpose of this report and its overall structure.

1.1 Background and Motivation

The exchange rate is a fundamental economic factor, significantly impacting both domestic and international economic relations. The exchange rate acts as a bridge for financial communication between various countries (Pradeepkumar & Ravi, 2018). Its instabilities will not only affect the country's international trade and capital flows but also directly impact the international investment of enterprises, foreign trade and individual investment. Forecasting exchange rate trends is an essential basis for judging the timing of exchange rate transactions.

Accurate exchange rate forecasts can provide a reasonable reference for investors and policymakers to formulate strategies. From a personal perspective, by accurately analyzing the exchange rate market and determining the overall trend, investors can grasp the appropriate buying and selling opportunities and thereby obtain more profits in the exchange rate trading market (Patel, Patel, & Patel, 2014). For the government, accurate exchange rate forecasts provide a solid basis for relevant management departments, which is beneficial for the government to adjust resource allocation effectively, reduce economic pressure caused by violent exchange rate fluctuations, and is of outstanding significance to stabilizing the market (Alagidede & Ibrahim, 2017).

The exchange rate market is a non-linear dynamic market characterized by complexity, diversity and uncertainty. This makes exchange rate forecasting more challenging. Currently, two well-known analysis methods are employed in forecasting the exchange rate market research: fundamental analysis and technical analysis (Ranjit, Shrestha, Subedi, & Shakya, 2018). Fundamental analysis focuses on studying basic information, generally based on the status of macro factors, changes and their impact on exchange rate trends (De Grauwe & Grimaldi, 2018). It then outlines conclusions about currencies' supply and demand relationship to judge the exchange rate trend. Basic information mainly includes the economic growth level of the country, the balance of payments situation, the inflation rate and other information in the economic report, such as political events, national economic data, investor sentiment, and the intervention of various central banks. For example, the U.S. Non-Farm Payroll data is one of the economic data that the exchange rate market needs to focus on monthly. After the data is released, it may cause a turning point in the direction of the foreign exchange market and even bring violent fluctuations to the exchange market (Pearce & Solakoglu, 2007). However, technical analysis predicts exchange rate trends by studying historical exchange rate prices and trading volumes (Cavalcante, Brasileiro, Souza, Nobrega, & Oliveira, 2016). Popular techniques include Moving Averages, Relative Strength Index, RSI, Moving Average Convergence Divergence, and Exponential Smoothing.

With the introduction of artificial intelligence, research on related technologies in financial time series forecasting has also obtained more and more attention. Unlike traditional time series methods, these techniques can handle the non-linear, chaotic, noisy and complex data of exchange rate markets, allowing for more effective forecasts (Rout, Dash, Dash, & Bisoi, 2017). The dataset is crucial in exchange rate forecasting, mainly including exchange rate prices, volatility, etc. However, if the selected time series is long and has high dimensions, it is tough to achieve the expected results using existing models for prediction. Afterwards, with the rapid growth of artificial intelligence technology, the usage of deep learning models to process time series-related tasks became the recent mainstream, and a series of neural network models for time series tasks appeared. Early proposed models such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) are considered suitable for processing time series tasks (Pirani, Thakkar, Jivrani, Bohara, & Garg, 2022). Islam and Hossain (2021) proposed the GRU-LSTM model. By comparing and evaluating the results of independent LSTM and GRU models, the proposed GRU-LSTM model achieved the best results in the two currency pairs.

1.2 Research Questions

The exchange rate significantly impacts the country as a whole, as well as companies and individuals. Therefore, using deep learning technology to accurately and effectively predict exchange rates is of tremendous value. As the most popular mainstream architecture in deep learning in recent years, the Transformer models are widely adopted in typical tasks such as text classification, sentiment analysis, target detection, speech recognition, etc. However, there are few related works in the field of time series analysis, and multiple financial time series analysis research still uses traditional sequence prediction methods. Therefore, this report proposes the research questions as follows:

(1) *How does the Transformer model perform in predicting the exchange rate?*

(2) *By comparing the Transformer, Informer and Temporal Fusion Transformer, which algorithm performs best in predicting the exchange rate?*

The fundamental idea of this report is to predict the exchange rate of the New Zealand dollar (NZD) against several other currencies. By utilising three models, Transformer, Informer, and Temporal Fusion Transformer (TFT), compare, analyze, and evaluate the performance of these three models on forecasting exchange rates.

1.3 Contributions

The focus of this report is mainly on effectively achieving accurate prediction of exchange rates based on deep learning. Here are contributions of this report listed below:

- Firstly, we collected several related exchange rate datasets, NZD/USD, NZD/GBP, NZD/AUD, and NZD/CYN. Each dataset contains 4980 samples of data and seven classes. By preprocessing the data, we experimented with the Transformer model and analyzed the model's performance based on the four datasets. This also fills the research gap of the Transformer model in financial time series analysis.

- Secondly, we presented two novel models established on the Transformer framework, Informer and TFT. Through utilising the same datasets, models were trained and tested. We compared and analyzed the advantages and disadvantages of the three models.

Finally, after accomplishing the experiment, the evaluation results demonstrate the well-trained model, which would be beneficial to exchange rate prediction in the New Zealand financial market.

1.4 Objectives of This Report

Overall, this project aims to achieve exchange rate predictions based on NZD and discover the most advanced algorithms fitting for exchange rate predictions through deep learning.

Firstly, based on Transformers, we study two recent algorithms, Informer and TFT. During experiments on Google Colab, we will train the model, adjust parameters, and obtain results established on four processed datasets. Subsequently, the performance of the three algorithms is compared and analyzed to determine the optimal forecast exchange rate model. Eventually, this report will explore the pros and cons of the model, summarize the experimental results, and provide references for other related research

1.5 Structure of This Report

The main structure of this article is as follows:

In Chapter 2, we conduct a literature review on exchange rate prediction, focusing on traditional exchange rate forecasting methods and others using neural networks. Additionally, we explore the models employed in our experiments.

In Chapter 3, we elaborate on the methodologies of the three models. Then, we describe the processes of data collection and processing. Besides, the three models' experimental procedures and the model evaluation measures are also clearly explained.

In Chapter 4, we analyze the four datasets. Through experiments in the three models, we explain and analyze according to the evaluation criteria. Afterwards, we conduct ablation experiments on the TFT model to demonstrate the significance of the VIX parameter in exchange rate prediction.

In Chapter 5, we comprehensively compare the performance of the three models and analyze them in conjunction with their own characteristics. At the same time, we also discuss the limitations of this project.

In Chapter 6, we summarize the experimental project and outline future work directions.

Chapter 2
Literature Review

This chapter primarily consists of a literature review on methods for predicting exchange rates. It explores three main areas: Traditional forecasting methods and methods related to neural networks. It also investigates the three models used in this experiment: Transformer, Informer, and TFT.

2.1 Introduction

The research blossoming of exchange rate forecasting has undergone several stages, such as exchange rate determination theory, linear time series analysis, and nonlinear forecasting. Since the exchange rate is non-stationary in mean and variance, its relationship with other data series changes dynamically due to nonlinear and dynamic changes in the exchange rate over time (Xu, Han, Wan, & Yin, 2019). As international trade continues to grow at an increasing rate, it is becoming more and more common, and the factors affecting exchange rates gradually increase (Eichengreen, 2007). Predicting exchange rate changes accurately is a challenging task that cannot be achieved through a single model alone. Therefore, the combination of nonlinear and multivariate models has evolved a trend in exchange rate forecasting (Sutcu & Gulbahar, 2023).

2.2 Traditional Exchange Rate Prediction Solutions

2.2.1 ARIMA

ARIMA is one of the most universal linear methods for forecasting time series, and its research has achieved great success in academic and industrial applications. In the study of the USD/TRY exchange rate forecast, Yıldıran and Fettahoğlu (2017) generated long-term and short-term models based on the ARIMA framework. Through comparison, it was found that ARIMA is more fitting for short-term forecasts. Similarly, Yamak, Yujian, and Gadosey (2019) used a data set of Bitcoin prices and applied with ARIMA, LSTM and GRU models for prediction analysis. The results showed that ARIMA delivered the best results among these models, with MAPE and RMSE of 2.76% and 302.53, respectively.

In the framework ARIMA (*p, d, q*) model, the variable value is predicted to be a linear function of the past several observations and random errors, and it demands that the input to the ARIMA model is linear and fixed data. ARIMA performs a stationarity check on the given time series data to determine whether the mean value and autocorrelation coefficients are constant over time (Qonita, Pertiwi, & Widiyaningtyas, 2017). If not fulfilling as a fixed attribute, Arima intends to utilise the *d-th* variance method until the sequence evolves fixed attributes and the model difference order is set to *d*. Eventually, an autoregressive moving average is applied to the resulting data.

In the modelling process, given a time series, the value at time *t* is y_t, and the random error term is ϵ_t. y_t is a linear function of the past p observation values y_{t-1}, y_{t-2}, ..., y_{t-p} and q random error terms ϵ_t, ϵ_{t-1}, ..., ϵ_{t-q}.

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_p y_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \cdots - \theta_q \epsilon_{t-q} \quad (2.1)$$

where a_1, a_2, ..., a_p are the corresponding autoregressive coefficients, θ_1, θ_2, ..., θ_q are the moving average coefficients. The random error term ϵt is a distribution with zero mean and constant variance.

Similar to the differential order d, p and q are also orders of the model. When q equals zero, the model simplifies to the AR model of order p (AR (p)). If p equals zero, the model is an MA model of order q[MA (q)].

The ARIMA model accurately predicts stationary time series data but assumes that future data values depend linearly on current and past data values. However, multiple real-world time series data indicate complex nonlinear patterns, so ARIMA cannot effectively model based on them (Pahlavani & Roshan, 2015).

2.2.2 GARCH

The GARCH model is mainly employed to describe the volatility of sequence data (Lahmiri, 2017). Therefore, it is often widely used to analyze and investigate financial data. This model is able to predict volatility using a similar modelling approach to ARIMA. The model is generally considered reliable due to its interpretability and theoretical guarantees. In recent outcomes, Lin (2018) collected data based on Shanghai stocks and used GARCH and ARIMA models to compare the forecasting effects. The results revealed that the GARCH model performs better, reflecting the stock market's volatility.

GARCH model is a refinement based on the limitations of the ARCH model (Almisshal & Mustafa, 2021). It modifies the conditional variance function to obtain a GARCH model with better application outcomes. Therefore, the GARCH model is also called the generalized ARCH model. The definition of GARCH(*p, q*) is:

$$R_t = \mu + a_t$$

$$a_t = \varepsilon_t \sigma_t \tag{2.2}$$

$$\sigma_t^2 = a_0 + \sum_{i=1}^{p} a_i a_{t-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2$$

where $a_o > 0,\ a_i \geq 0, i = 1, 2, \cdots, p, \beta_j \geq 0, j = 1, 2, \cdots, q$, therefore, the conditional variance of GARCH (*p,q*) is always positive.

2.3 Typical Neural Network Methods for Predicting Exchange Rate

2.3.1 RNN

Artificial neural networks can only process inputs separately, meaning there is no relationship between the previous and next inputs. For exchange rate time series, dealing with the connection between previous and subsequent inputs is essential. Ordinary neural networks are unable to solve this problem. RNN is one of the neural networks specifically designed to handle time series problems (Hu, Zhao, & Khushi, 2021). It has the ability to extract information from a time series, allow the information to persist, and use previous knowledge to infer subsequent patterns. Traditional neural networks such as the Backpropagation Neural Network (BPNN) are also used for time series modelling, while the time series information of such models is usually less than RNN.

Differing from Fully Connected Neural Networks (FNN), RNN utilises internal memory to execute inputs and is capable of analyzing time series data in multiple natural language processing fields, such as handwriting recognition and speech recognition (Hori, Cho, & Watanabe, 2018).

RNN models consist of varying numbers of layers, each containing different types of units. The model processes data at any time according to an entire input sequence. Figure 2.1 shows the RNN model structure.

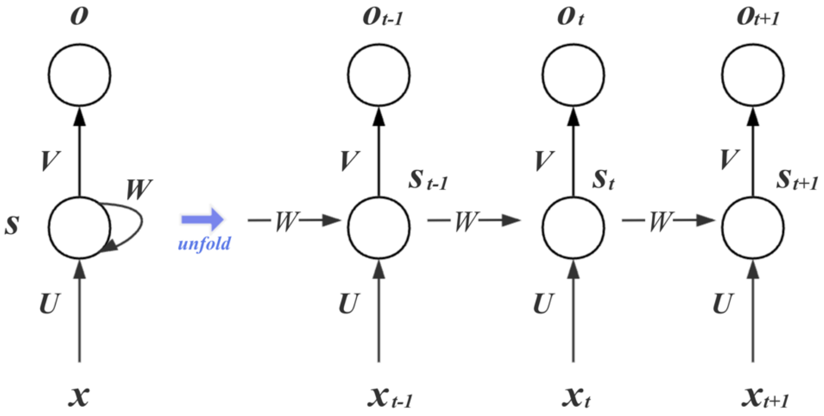


Figure 2.1 The structure of RNN

In Figure 2.1, s_t is the hidden state of node *s* at time *t*, x_t means the input at the current moment, and the output is o_t. The parameters *U*, *W*, and *V* are all shared at different times.

During the training process of RNN, the Backpropagation Through Time (BPTT) algorithm is commonly employed (Ernoult, Grollier, Querlioz, Bengio, & Scellier, 2019). The error between the model prediction result and the actual answer is reflected in the input and output weight over time *t*. Training an RNN is tricky due to its architecture, which includes backward temporal dependencies. Therefore, as the duration of the learning process extends, the complexity of RNN will progressively increase. Eq. (2.3) illustrates the learning process of RNN.

$$h_t = Wf(h_{t-1}) + W^{(hx)}x_{[t]}$$

$$y_t = W^{(S)}f(h_t) \qquad (2.3)$$

$$\frac{\partial E}{\partial W} = \sum_{t=1}^{T} \frac{\partial E_t}{\partial W}$$

where *E* denotes the loss function, *W* represents the weights in the neural network, and *t* defines the time step.

The architecture of RNN is also defined by its hyperparameters, and the choice of parameters influences its performance (Yu, Kim, & Mechefske, 2021). These hyperparameters include the number of hidden layers and units in each layer, regularization techniques, network weight initialization, activation functions, learning rates, batch size (minimum batch size) and optimization algorithm, etc.

2.3.2 LSTM

Although RNN has outstanding advantages in dealing with time series problems, as the training time rises and the number of network layers increases after the nodes of the neural network have been calculated in many stages, the features of the previous relatively long time slice have been covered, so problems such as vanishing gradient or exploding gradient are prone to occur, which leads to the incapability to learn the relationship between information, thereby losing the ability to process long-term series data (Li, Li, Cook, Zhu, & Gao, 2018).

In order to resolve the problem of RNN's difficulty in learning long-term dependencies, an improved RNN model was created, namely LSTM (Hochreiter & Schmidhuber, 1997). LSTM introduces a memory unit into each neuron module within the RNN network's hidden layer and utilises three gate control units, input gate, output gate, and forget gate, to control the state of each memory unit, respectively (Bai, 2018). Generally speaking, RNN only has the function of short-term memory, while LSTM has the function of long-term memory. LSTM is primarily used for language modelling, translation, speech recognition, sentiment analysis, predictive analysis and financial time series analysis.

The working approach of the LSTM model is as follows: Firstly, the input and output data from the previous moment in the hidden layer jointly influence the forgetting gate. The forgetting gate filters the above information, memorises important feature information in the time series, and discards irrelevant information. Then, the input and output data of the hidden layer at the previous moment are used and updated as input information for the input gate. Secondly, the memory unit updates its state through the input data, the output data of the hidden layer at the previous moment and the state of the memory unit at the previous moment. Finally, the input data, the output data of the hidden layer at the previous moment, and the memory unit state at the current moment are used concurrently on the output gate to output the hidden layer information at the present moment. The structure diagram of LSTM is as follows:

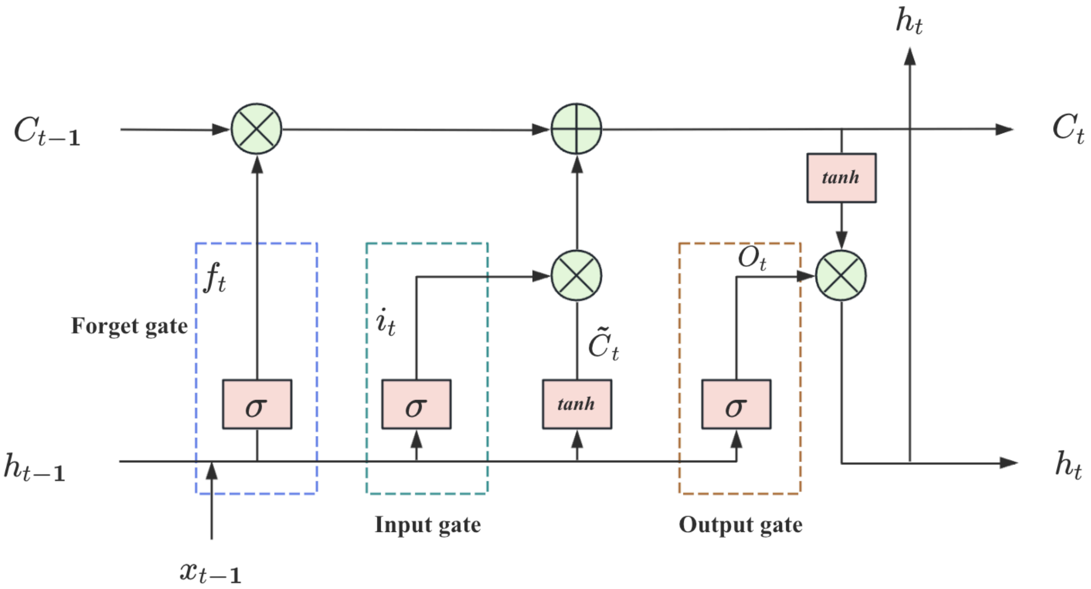


Figure 2.2 The structure of LSTM

The information transmission process of neurons in the LSTM model is shown in the following way.

$$
\begin{gathered}
f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \\
i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i) \\
\tilde{C}_t = \tanh\,(W_c \cdot [h_{t-1}, X_t]\ + b_c) \\
C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \\
O_t = \sigma(W_o \cdot [h_{t-1}, X_t] + b_o) \\
h_t = O_t * \tanh\,(C_t)
\end{gathered}
\tag{2.4}
$$

where f_t is the output of the forget gate at time *t*, σ is the sigmoid activation function, W_f is the weight of the forget gate, h_{t-1} is the outcome of the hidden layer at time *t* -1, X_t is the input of the input gate at time *t*, b_f is the deviation of the forget gate, i_t is the output of the input gate at time *t*, C_t is the candidate cell state at time *t*, *tanh* is the tanh activation function, O_t is the output of the output gate at time *t*, h_t is the output of the hidden layer at time *t*.

2.3.3 GRU

Compared with LSTM, the Gated Recurrent Unit Neural Network (GRU) has a more straightforward network structure (Munir, Ren, Mustafa, Siddique, & Qayyum, 2021). At the same time, GRU is also a neural network that introduces a "gating" mechanism. However, the difference between GRU and LSTM is that it only has one leading line responsible for memory and does not require the assistance of other memory units (Munir et al., 2021). In terms of function, GRU is similar to LSTM, both of which are designed to solve the inherent gradient problem of RNN (Gharehbaghi, Ghasemlounia, Ahmadi, & Albaji, 2022). The following is the GRU structure diagram.

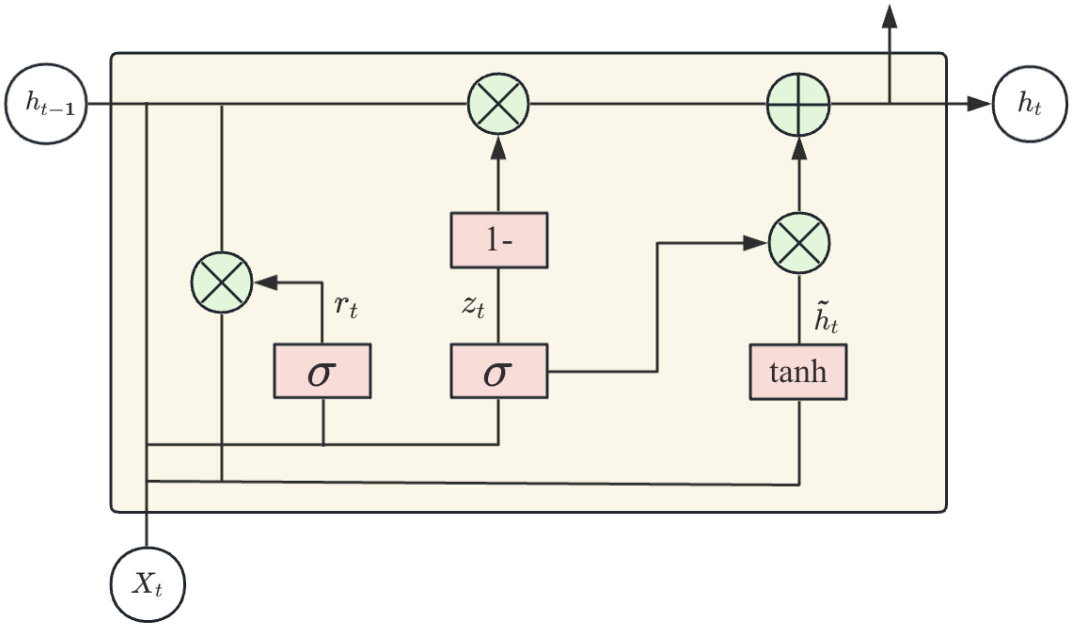


Figure 2.3 The structure of GRU

The update gate and reset gate in GRU instantly replace the previous input gate and forget gate. The calculation of the update gate is as follows:

$$h_t = z_t * h_{t-1} + (1 - z_t) * g(x_t, h_{t-1}; \theta)$$
$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{2.5}$$

where $z_t \in [0,1]^D$ represents the update gate. The calculation method of the candidate state h_t at the current moment:

$$\tilde{h}_t = \tanh\left(W_h x_t + U_h(r_t * h_{t-1}) + b_h\right) \tag{2.6}$$

where $r_t \in [0,1]^D$ represents the reset gate, which can control the dependence of the current candidate state $\tilde{h}_t$ on the input information of the previous state.

GRU updates are calculated as

$$h_t = z_t * h_{t-1} + (1 - z_t) * \tilde{h}_t \tag{2.7}$$

GRU can also be considered an LSTM with a more compact structure with fewer internal parameters. Based on this characteristic, GRUs are more accessible to fit during training. With less experimental data, GRU will have better prediction results than LSTM. When there is a large amount of experimental data, the LSTM model will execute better than GRU. In the study of predicting traffic flow, Hussain, Afzal, Ahmad, and Mostafa (2021) utilised the GRU model to train one month's traffic data. They applied the optimizer ADAM to optimize the hyperparameters and adjust the window step size during the experiment. By comparing the test results of the LSTM and ARIMA models, GRU takes relatively less training time, reduces errors, and improves overall performance by 4.5% compared to other models.

2.4 Attention Mechanism

The attention mechanism is one of the most widespread techniques in the current field of deep learning, employed to enhance the expressiveness of the model when processing sequence data (Song, Luktarhan, Shi, & Wu, 2023). Traditional models such as RNN and LSTM face challenges in capturing long-term dependencies within extended sequential data. Nevertheless, the attention mechanism enhances the model's capacity to train on lengthy sequences by learning to assign different weights to input information at distinct positions (Niu, Zhong, & Yu, 2021).

Specifically, the attention mechanism assigns a weight or score to each input position during calculation. These weights or scores represent which input positions the model should focus on when calculating the output. This allows the model to process inputs more flexibly, thereby improving the model robustness and performance.

General attention mechanisms include Luong attention and Bahdanau attention, which are frequently employed in machine translation, speech recognition, and natural language processing (T. Wang et al., 2020). Figure 2.4 illustrates the attention mechanism as a process that maps a query along with a collection of key-value pairs to output, with the query, keys, values, and output all represented as vectors. The output is a weighted sum of values. The weight of each value is calculated by the query's compatibility function with the corresponding key.

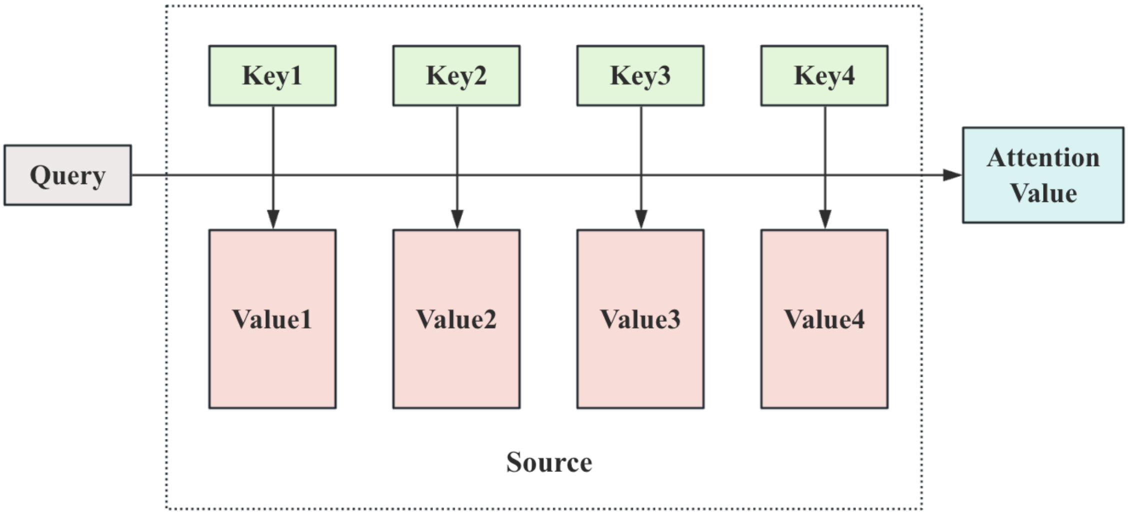


Figure 2.4 The overview of the attention mechanism

In the computation process, the first step involves determining the similarity between the query and each key to calculate the respective weight, as shown in eq.(2.8). Function options for calculating similarity include the dot product, splicing, and perceptron, among others, where *Q* represents the query, and *K* represents the key.

$$f(Q, K_i) = \begin{cases} Q^T K_i & dot \\ Q^T W_\alpha K_i & general \\ W_\alpha [Q; K_i] & concat \\ v^\tau \tanh\,(W_\alpha Q + U_\alpha K_i) & perceptron \end{cases} \tag{2.8}$$

Then, the *softmax* function to normalize the above weights as

$$a_i = soft\text{max}\big(f(Q, K_i)\big) = \frac{\exp\left(f(Q, K_i)\right)}{\sum_j \exp\left(f\big(Q, K_j\big)\right)} \tag{2.9}$$

Eventually, the normalized weights and related values are weighted and summed to obtain the final attention calculation result.

$$Attention(Q, K, V) = \sum\nolimits_i a_i V_i \tag{2.10}$$

By incorporating an attention mechanism, the model can prioritize critical information while disregarding unimportant details. Based on Hong Kong stock price prediction (Chen and Ge, 2019), the attention mechanism was applied and achieved favourable results. Their experiments indicate that due to the incorporation of the attention mechanism, the LSTM's memory cell structure decreases the long-term dependency of sequences, making the LSTM with attention mechanism superior to conventional LSTM models.

In sequence-to-sequence situations, the attention mechanism mainly relies on the encoder-decoder architecture. The encoder encodes the input into a fixed-length context vector, and the decoder gradually obtains the complete target output based on the context vector and the currently decoded output. The encoder compresses all the input information into a fixed-length latent vector, which may cause the model's performance to drop sharply when the input sentence length is lengthy, especially when it is more extended than all the sentences in the training set. During the encoding process, each observation point in the sequence is given the same weight, leading to model limitations in some tasks.

2.5 Transformer

Transformer was initially explored by Vaswani et al. (2017), no longer stuck to the framework of RNN and CNN, and attention is applied to the seq-to-seq structure to form a transformer model and applied it to process natural language tasks. Since then, Transformer model has achieved outstanding results in fields such as computer vision (Han et al., 2022).

Transformer model is extensively applied to the field of natural language processing. Compared with LSTM, its main innovation is realized through self-attention. It adds a parallel computing mechanism compared to feedback neural networks (FNN) such as RNN and LSTM (Yang, Cen, Liu, Xiong, & Chen, 2022). After that, the improvement of the self-attention part includes two parts: Firstly, lowering the attention framework's computational complexity so that the input data's length can be increased and more extended learning for time dependence. The second is to improve the model's calculation efficiency and the effect of corresponding prediction tasks through structural advancements.

Sukhbaatar, Grave, Bojanowski, and Joulin (2019) proposed the application of adaptive attention length attention to Transformer. This also extends the input length of the Transformer model, learning longer context dependencies while maintaining the same memory space and computing speed as the original model. Guo et al. (2019) presented the Star-Transformer model, a star architecture consisting of a relay node and n satellite nodes. Related to the standard Transformer model, Star-Transformer is suitable for data sets of various sizes, reducing computational complexity, faster task processing, and more satisfactory performance.

Besides, the research work on Transformer in time series has also aroused great interest among scholars. Through experimental research on 12 public datasets with time series (Lara-Benítez, Gallego-Ledesma, Carranza-García, and Luna-Romera, 2021), it was found that Transformer can capture long-term dependencies and obtain the best accurate prediction results in five of the dataset training. However, its calculation is more complex than CNN, so the training process is relatively slow. Shiyang Li et al. (2019) discovered in their research that the self-attention of the traditional transformer model is insensitive to the local context, which would cause the model to have exceptions in prediction accuracy, and lengthy unnecessary information will limit the computational memory of the model, making the computational complexity of Transformer increases with sequence length. To address this problem, they proposed LogSparse Transformer, which assigns numbers to input data and selects log(N) data points according to the principle of log(N), thus forming a sparse attention mechanism. From the result of the experiment, it indicates that the newly introduced model offers superior benefits compared to the conventional Transformer.

Although numerous scholars have conducted in-depth research outcomes on the Transformer, it is evident from the literature that most studies primarily focus on reducing the computational requirements of the Transformer model (Tay, Dehghani, Bahri, & Metzler, 2022). However, they overlook the importance of capturing the dependencies among neighbouring elements, addressing the heterogeneity between the values of time series data, the temporal information corresponding to the time series, and the positional information of each dimension within the time series.

2.6 Informer

In order to solve the heterogeneity of time information, position information and numbers, A model based on Transformer architecture and attention mechanism was offered (Zhou et al., 2021). For the first time, time coding, position coding and scalar were introduced in the embedding layer to crack the long sequence input problem. ProbSparse self-attention captures long-distance dependencies and lessens the time complexity in the calculation process. Using the distillation mechanism can effectively reduce the time dimension of the feature map and lower memory consumption. Although Informer outperforms LSTM in time series forecasting tasks, its inability to capture dependencies among neighbouring elements with the multi-head attention mechanism leads to insufficient capture of the time-series local information. This results in lower prediction accuracy and higher memory consumption, which could be more conducive to large-scale deployment.

Gong et al. (2022) introduced a relative coding algorithm based on the Informer framework to predict the heating load. The experimental results showed that the improved Informer model is more robust. Wu, Xu, Wang, and Long (2021) conducted research based on Informer and proposed Autoformer, a new decomposition architecture designed with an autocorrelation mechanism. The model breaks the preprocessing convention of sequence decomposition and updates it into the basic internal blocks of the deep model. This design gives Autoformer the ability to decompose complex time series progressively. In addition, inspired by the random process theory, Autoformer designed an autocorrelation mechanism based on sequence periodicity, replacing the Self-Attention module in Transformer with autocorrelation mode. In long-term forecasting, Autoformer achieves outstanding accuracy.

2.7 TFT

Transformer model has achieved excellent results in natural language processing and computer vision tasks (Bi, Zhu, & Meng, 2021). Applying this model to capture long-term dependencies and data interaction in time series has become the focus of many scholars. The general method for processing time series data is to treat data in all dimensions with equal weight. This may cause the model to ignore some critical input information or be interfered with by noise, which is also a shortcoming of traditional processing methods.

Temporal Fusion Transformer (TFT) is a Transformer model for multi-step prediction tasks, which is developed to effectively process different types of input information (i.e., static, known or observed inputs) and construct feature representations to achieve high predictive performance (Lim, Arık, Loeff, & Pfister, 2021). Zhang, Zou, Yang, and Yang (2022) utilised the TFT model to predict short-term highway speed. They collected Minnesota traffic data and applied it to the training and testing of the model. Compared with traditional models, the TFT model performs best when the prediction range exceeds 30 minutes.

In the electricity load forecasting study, Huy, Minh, Tien, and Anh (2022) compared the TFT with traditional statistical prediction models. The results demonstrated that deep learning methods exceed statistical models, and the forecasting results of TFT were significantly better than those of traditional methods. In addition, Hu (2021) proposed that in the study of predicting stock prices based on the TFT model, it was discovered that compared with the SVM and LSTM models, TFT had the lowest error.

Chapter 3
Methods

This chapter illustrated the methodologies of Transformer, Informer, and TFT models. Subsequently, we describe the datasets and the preprocessing methods. The experimental processes of the three models are presented, and the measures for model evaluation are clarified.

3.1 Transformer

In Transformers, the self-attention mechanism has received more recognition compared to other neural network models that utilize the attention mechanism. The attention mechanism of Transformers is better at capturing the internal correlation of data and features, and more effectively solves the problem of long-distance dependence (Wang, Pi, Zhang, Liu, & Guo, 2022). In the Transformer, query is $Q \in \mathbb{R}^{N \times D_k}$, key is $K \in \mathbb{R}^{M \times D_v}$, value is $V \in \mathbb{R}^{M \times D_v}$, the scaled dot-product attention is defined as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3.1}$$

where *N* and *M* define the length of the query and key (or value), D_k and D_v indicate the dimensions of the key (or query) and value. Normalization is achieved using the scaling factor $1/\sqrt{d_k}$ to separate the vector dimension from the softmax distribution, ensuring stable gradients during training and preventing gradient vanishing issues. Consistency in the dimensions of the query and key is necessary, and the lengths of the key and value should match. This process somewhat represents the same sequence in different spaces.

Contrary to other models that only take use of a single attention module, the Transformer employs multi-head attention modules to operate in parallel (Sridhar & Sanagavarapu, 2021). In this step, the original queries, keys, and values of dimension D_m are each projected into spaces of dimensions D_k , D_m , and D_v using *H* different learned vectors. The model computes each of these projected queries, keys, and values according to formulas 3.2, outputting attention weights for each. Then, it concatenates all these outputs and projects them back into an D_m dimensional representation.

$$\begin{gathered} MultiHeadAttn(Q, K, V) = Concat(head_1, \cdots, head_H)W^O, \\ where \quad head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \end{gathered} \tag{3.2}$$

As shown in Figure 3.1, a Transformer model for sequence-to-sequence tasks typically includes an encoder and a decoder, each comprising multiple layers. Initially, the model transforms each input sequence element into a vector by combining its embedding with its positional embedding. This creates a matrix of representation vectors for the input sequence. The encoder processes this matrix through six layers, resulting in an encoded matrix of the exact dimensions, which captures the contextual information of each element in the sequence. This encoded matrix feeds into the decoder. During generating the sequence, the decoder forecasts the i+1th observation point based on the first i observation points, with the i+1th and following observation points being masked.

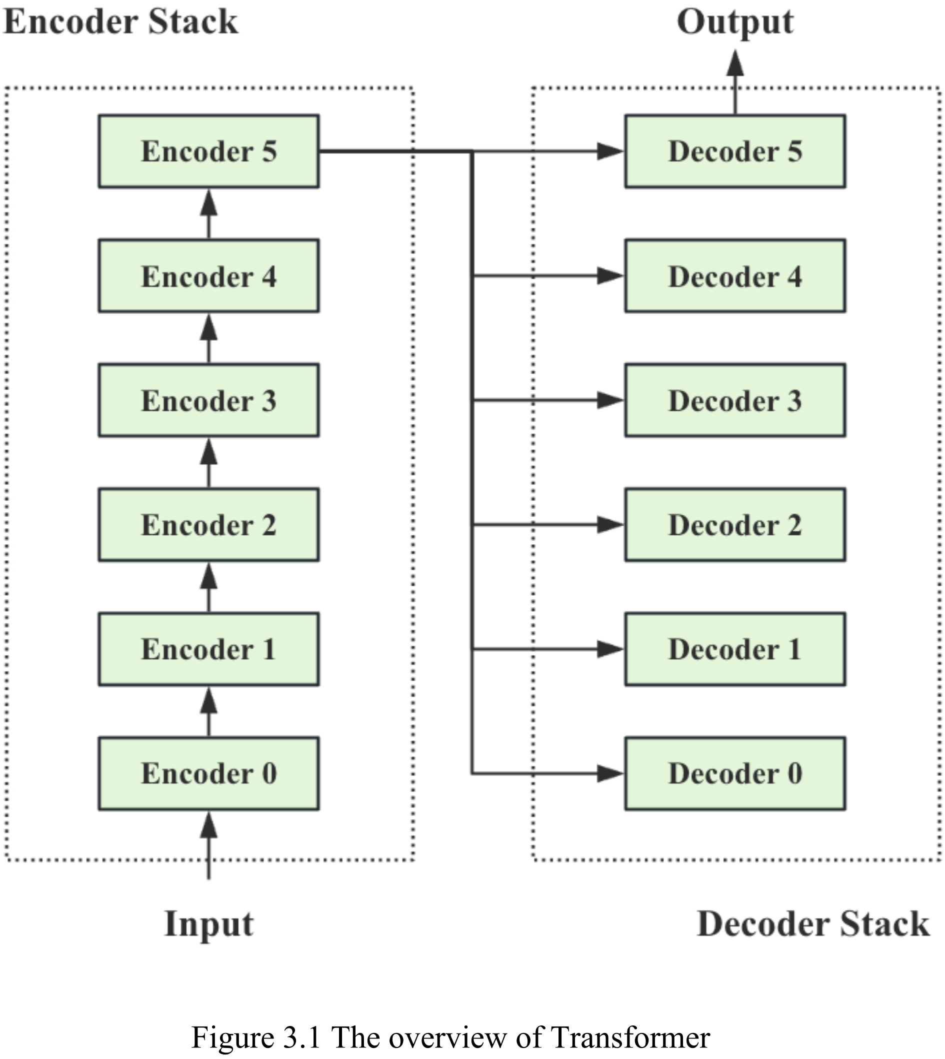


Figure 3.1 The overview of Transformer

In the encoder-decoder structure, the input X^t passes through the encoder and is converted into a hidden state H^t, and the output Y^t is obtained from $H^t = \{h_1^t, \cdots, h_L^t\}$ during the decoding process. This inference involves a sequential process known as ‘dynamic decoding’, where the decoder updates to a new hidden state h_{k+1}^{t} using the prior state h_{k}^{t} and additional outputs from the k_{th} step, subsequently predicting the $(k+1)_{th}$ sequence y_{k+1}^{t}.

In Figure 3.2, the Transformer is composed of an interconnected encoder (left) and a decoder (right) including repeatedly stacked blocks. Here is a simplified illustration of the encoder and decoder. For simplification, only one block in both the encoder and the decoder is shown here. Each encoder module primarily comprises a multi-head self-attention module, a Position-wised Feedforward Network (PFFN), and residual connection and layer normalization modules. The multi-head self-attention module consists of multiple self-attention modules and is used to obtain the relationship between the current generated sequence and the previously generated context.

In constructing the stacked model, a residual connection surrounds each module, followed by a layer normalization module, represented as *Add&Norm* in Figure 3.2. The residual connection is commonly utilised to address the training issues of deep networks, allowing the network to focus only on the current difference and preventing network degradation. Layer normalization is utilised to normalize the activation values of the layer, ensuring that the mean and variance of the inputs to each neuron are consistent across layers to accelerate convergence. In eq. (3.3), *X* represents the input of multiple-head attention or FNN, $MultiHeadAttn(X)$ and $PFFN(H')$ defines the output, where the output and input dimensions remain equal.

$$\begin{gathered} H' = LayerNorm(MultiHeadAttn(X) + X) \\ H = LayerNorm\big(H' + PFFN(H')\big) \end{gathered} \tag{3.3}$$

PFNN processes a three-dimensional tensor with dimensions (batch_size, seq_length, feature_size) and features two fully connected layers. The activation function for the first layer is ReLU, while the second layer lacks an activation function and operates along the last dimension. Given that each position in the sequence is updated independently, this operation is analogous to a 1×1 convolution.

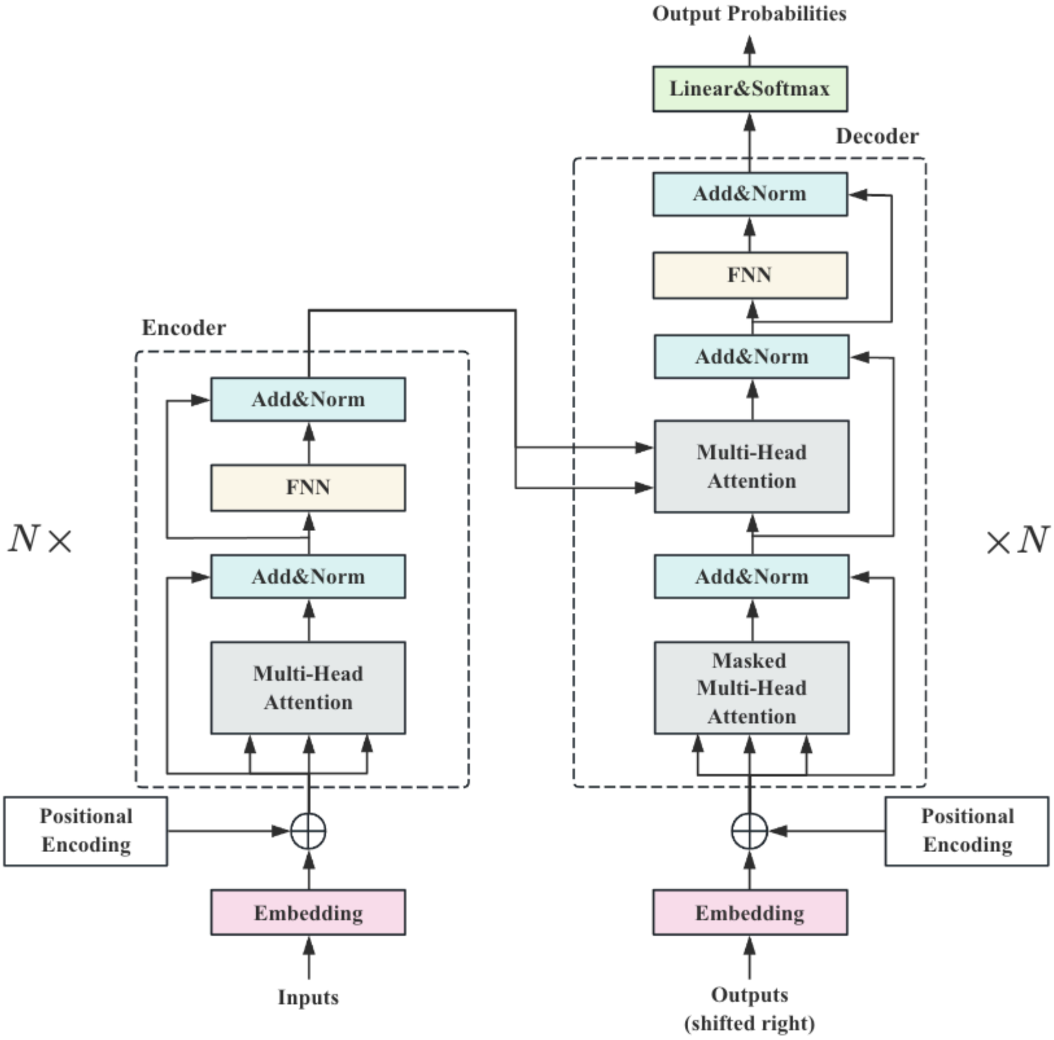


Figure 3.2 The network structure of Transformer

The architecture of decoders resembles the encoder's, though there are distinctions. Firstly, the decoder includes two multi-head attention layers: The first layer masks the input, while the query and value matrices of the second layer are derived from the output of encoders encoding information matrix. The query matrix for this second layer is generated from the output of the prior decoder module. Second, before producing its output, the encoder applies a Softmax layer to compute the probability of the following predicted observation. Nevertheless, a number of characteristics of the initial Transformer result in poor performance on long sequence forecasting tasks, including:

1) The self-attention operation of the model makes its computation relatively slow.

2) The multi-layer stacking design of its encoder/decoder causes the space complexity to increase quadratically in relation to the sequence length, leading to

excessive memory usage when processing long sequences, making direct modelling of long-time sequences impractical.

3) In the generation process of long sequences, the dynamic encoding mechanism of this proposed model results in slow generation speeds, similar to RNN models.

3.2 Informer

The Informer model has been proposed to address the long-sequence forecasting issues in the Transformer. This model provides an improved self-attention module to reduce time complexity (Sun, Hou, Lv, & Peng, 2022). During the convolution process, the Informer halves the feature maps before concatenating them. Additionally, the Informer model takes only a tiny portion of the input sequence from the end as the starting tokens and uniformly generates the forecast sequence during output. Figure 3.3 demonstrates the structural diagram of the Informer model.

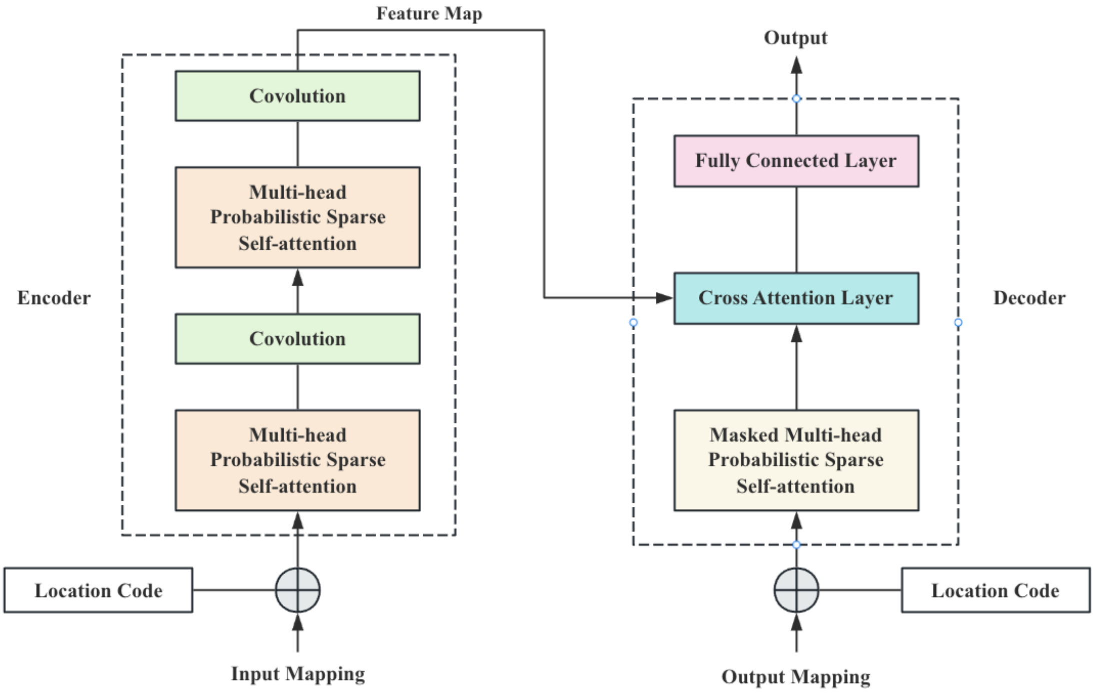


Figure 3.3 The structure of Informer model

The conventional self-attention mechanism in the Transformer network requires storing a large amount of computation and quadratic equations for dot product calculations (Al-Ali et al., 2023). In the Informer network, probabilistic sparse self-attention replaces traditional self-attention. Each input vector is utilised to calculate query, key, and value vectors in the self-attention mechanism.

Then, attention weights are calculated by computing the dot product of query vectors and key vectors. The attention weights represent the similarity between each and all input vectors. Probabilistic research has demonstrated that self-attention is sparsely allocated and adheres to a long-tail distribution. In the probabilistic sparse self-attention mechanism, query vectors compute the similarity with each key vector, generating an attention distribution.

The query vectors may be more active in this process, i.e., they have higher similarity scores with more key vectors. In contrast, others may be more idle, i.e., they have higher similarity scores with fewer key vectors. For this situation, only the dot products of active queries are calculated, that is, only the dot products between query vectors with higher similarity scores and their corresponding key vectors are calculated. Their dot products are replaced with an average value for idle queries, thus avoiding unneeded calculations for unnecessary query vectors and improving computational efficiency. The probabilistic sparse self-attention calculation is shown in the eq. (3.4).

$$ProbAttn\left(\hat{Q}^l, K^i, V^i\right) = softmax\left(\frac{\hat{Q}^l {K^i}^T}{\sqrt{d}}\right) V^i \tag{3.4}$$

where $\hat{Q}^l$ represents the distance computed by the KL divergence among the attention distribution and the uniform distribution to determine the value of each query point, thus specifying which queries should be allocated computational resources, it selects the active query with the most significant distance from Top-u. In accordance with the sampling factor c, define $u = c \cdot \ln L_Q$. Where L_Q means the length of the query vector. When computing dot products in probabilistic sparse self-attention, the method of dot product sparsification diminishes the memory to $O(L^2)$, where L is the sequence length. Nevertheless, the time complexity still corresponds to $O(L^2)$. Calculations based on KL divergence are approximate values. Under a long-tail distribution, query vectors, key vectors, and value vectors of equal length should be selected for dot product calculations while setting other dot products to 0 for distance comparison. Typically, $L_Q = L_K = L$, making the time complexity evolve $O(L\ \ln L)$.

In general, in probabilistic sparse self-attention calculation, attention is only given to some far-active queries. In contrast, the dot products for other queries are substituted with the mean of the value vectors, thus reducing the computational task.

On the encoder side, the Informer network's self-attention produces outcomes equivalent to redundant value vector aggregates. Inspired by dilated convolutions, distillation techniques are used to compress the temporal aspect of the input, facilitating the processing of lengthy sequences. Priority is given to sequences with prominent features, and at the subsequent level, self-attention feature maps are created. Equation (3.5) illustrates the progression from layer *i* to layer *i +1*.

$$F_{i+1} = Maxpool(ELU(Conv1d([F_i]PSA))) \tag{3.5}$$

where $[F_i]PSA$ represents the fundamental operation of performing multi-head probabilistic sparse self-attention among them. F_{i+1} is the feature representation of the i_{th} layer, and the input sequence is diminished by half in each layer through convolution and maximum pooling, thereby progressively reducing the feature layer to achieve the shorten of the input sequence. Memory usage drops to $o((2-\varepsilon)L\ \log L)$. The encoder stack consists of multiple attention and convolutional layers with sequential sub-layer connections. The central stack structure diagram is displayed in Figure 3.4.

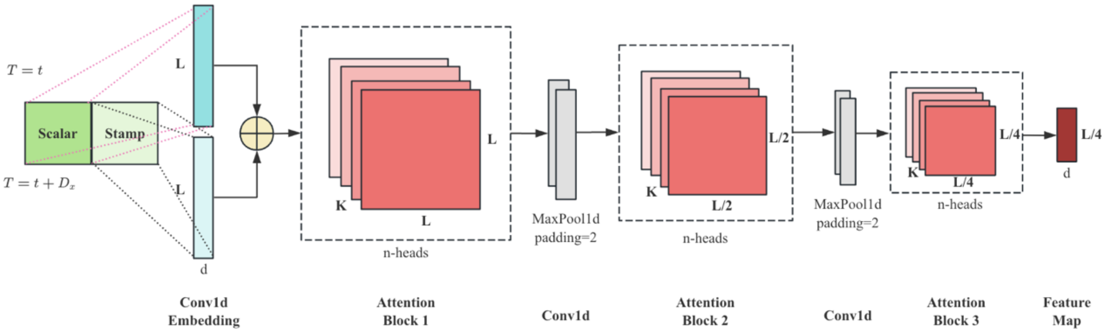


Figure 3.4 The structure of Informer encoder

The secondary stack is half the size of the primary stack, with both the encoding and convolutional layers decreased by one to strengthen the distillation operation's robustness. Moreover, each stack maintains identical output dimensions. Eventually, the outputs from all stacks are combined to produce the final hidden output of the encoder.

The decoder part performs attention operations on the intermediate results output by the encoder, reshapes the output data using a fully connected layer, and ultimately delivers the prediction results. The typical decoder comprises two multi-head self-attention mechanisms: the first employs probabilistic sparse self-attention, while the second utilizes conventional self-attention. Employing generative inference prediction, the input vector of the decoder is as shown in eq.(3.6).

$$X_{fdec} = concat(X_{token}, X_{phol}) \in R^{(L_{token}+L_y)\times d_{model}} \tag{3.6}$$

where X_{fdec} represents the input sequence of the decoder, X_{token} is the starting mark of the sequence, and X_{phol} symbolises the placeholder of the target sequence. Zeros are padded to timestamps to keep the dimensions consistent as the prediction sequence is entered. Masked multi-head self-attention is utilized, which obscures future information to guarantee that each position concentrates on present data and prevents autoregressive behaviours.

3.3 TFT

Temporal Fusion Transformer (TFT) is a time series prediction model based on the Transformer architecture proposed by Lim et al. (2021), aiming to solve the limitations of traditional time series prediction models. TFT introduces a novel to capture features and nonlinear relationships across multiple time scales (Fayer et al., 2023). TFT employs recurrent layers for localized processing and interpretable attention layers to manage long-term dependencies. The algorithm also leverages specialized components for feature selection and a series of gating layers to suppress unnecessary components, thereby maintaining the model's high performance in various scenarios. The framework of TFT specifically includes gated mechanisms, variable selection networks, static covariate encoders, an encoder-decoder-based LSTM model, an interpretable multi-head attention mechanism for integrating information, and a temporal fusion decoder. The model architecture of TFT is displayed in Figure 3.5:

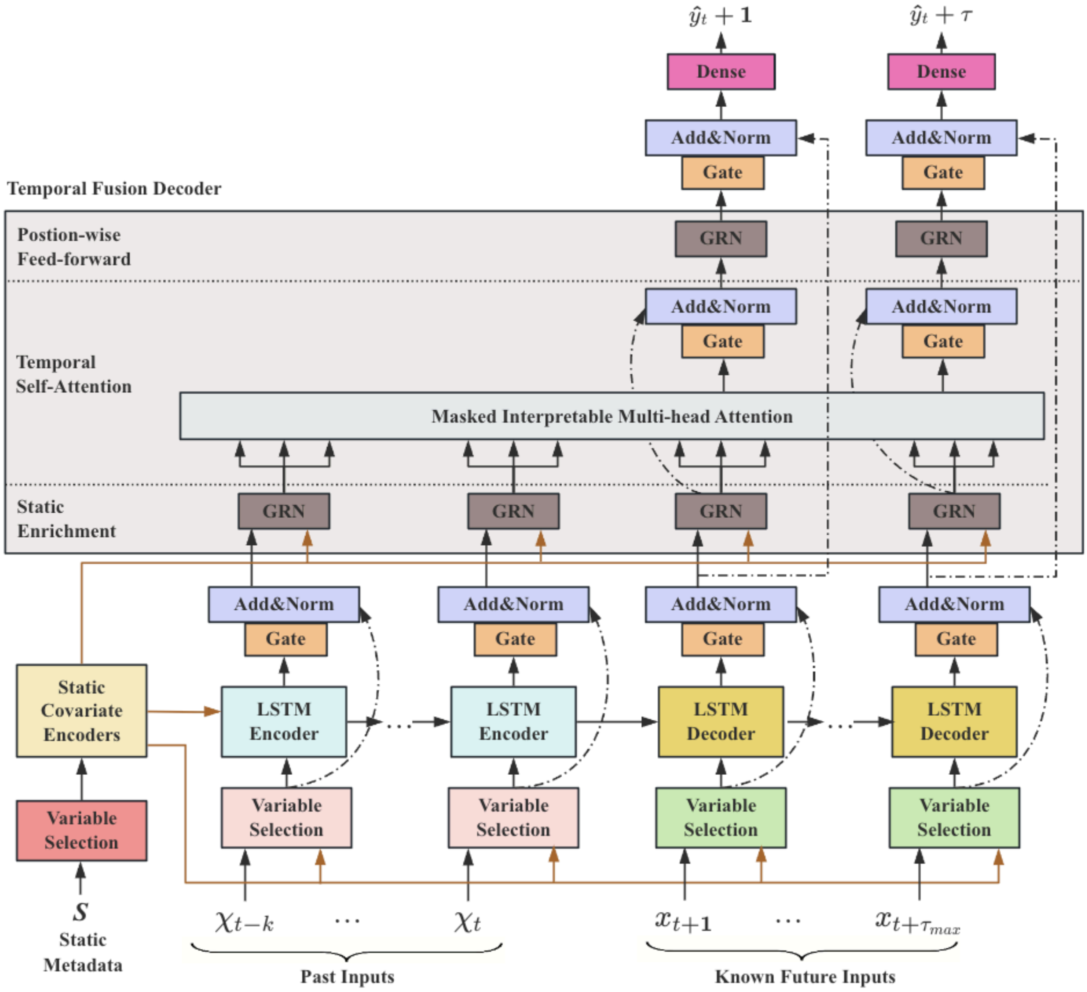


Figure 3.5 The architecture of the TFT model

The main components of this TFT model are:

(1) Gating mechanism and variable selection network

The input data features first pass through the GRN module to split and filter variables, assigning corresponding weights to the split vectors to form new feature vectors. This vector integrates the selected information of all input feature vectors, representing a high-level abstraction of a series of input features. GRN utilises skip-connection and GLU functions to control the contribution of feature information of linear and nonlinear features, especially by adding static covariates to train model learning. The variable selection network cooperates with GRN and Softmax for feature selection.

(2) Static covariate encoder

After filtering and sorting variables, the classic LSTM model that processes sequence data is used as the encoder/decoder. The forgetting, memory, and selective output mechanisms of the LSTM can better capture the long-term information of the sequence. In the static covariate encoder, one part of the module transfers the past features to the LSTM encoder, the other part passes the future features to the LSTM decoder, and finally, the LSTM encoder and decoder are combined to form a unified temporal feature and assigned to the next module.

(3) Temporal fusion decoder. This structure is conducted to capture both the long-term and short-term temporal relationships that exist in the dataset and consists of three sub-layers:

- Static Enrichment Layer (SEL). The GRN module introduces static covariates to enhance the timing features.
- Temporal Self-Attention Layer (TSL). This module can learn the long-term dependencies of time series data and provide model interpretability by fulfilling the self-attention mode in the attention mechanism.
- Position-wise Feed-forward Layer (PFL). Perform nonlinear processing on the information output by TSL.

The overall operation process of the TFT can be summarized as follows: Firstly, skip-connection and GLU processing are implemented through GRN, and then feature selection is performed through the variable selection network with the GRN and SoftMax functions. Ultimately, the final output of the model prediction is derived from the multi-head self-attention module within the temporal fusion decoder, which offers interpretability and the capacity to grasp long-term dependencies.

3.4 Dataset Collection

Due to the changes in the exchange rate between NZD and various currencies being impacted by multiple aspects, they display diverse characteristics of change. Therefore, we select four representative currencies, USD, GBP, CNY, and AUD, as training and test samples. The datasets of NZD against these four currencies are all from Yahoo Finance and Investing website (www.investing.com). To enhance the learning ability of our proposed model for unexpected fluctuations, each sample includes daily data from January 3, 2005, to February 2, 2024, totalling 4,980 entries. This period captures exchange rate variations during significant global market changes, such as the 2008 financial crisis, the COVID-19 pandemic in 2020, and the early 2022 Russia-Ukraine conflict. The primary variables of the dataset include closing, opening, highest, lowest, and floating prices of the day. We select the closing price as the experimental object.

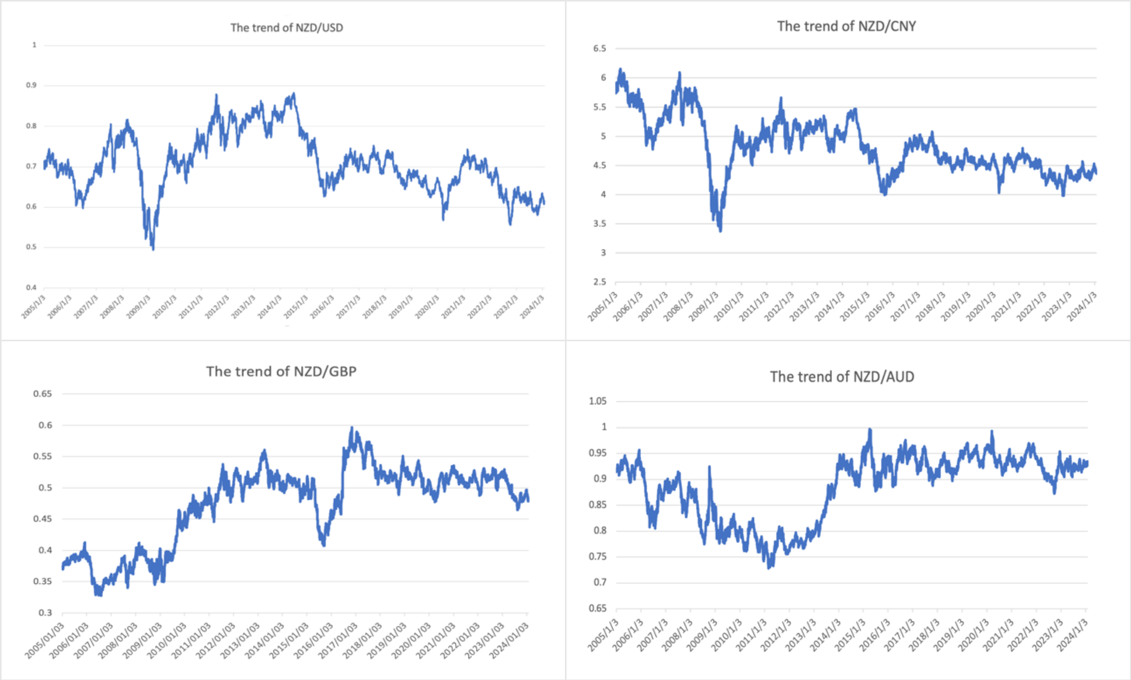


Figure 3.6 The trend charts of NZD against the four currencies

In addition to fundamental data, this study also chose the widely used VIX index in financial research applied for the NZD/USD dataset, a volatility index derived from the weighted average of the implied volatility of index options proposed by the Chicago Board Options Exchange, with the S&P 500 index as its reference index. In an era of increasing economic globalization and global stock market linkage, it is also widely used to indicate investor sentiment fluctuations in the international financial market. In the research work (Xu et al. , 2023), the VIX index was also included as a sentiment indicator of the exchange rate market to improve prediction precision and accurately fit the trend of exchange rate fluctuations under sudden events. Figure 3.6 shows the trend charts of NZD against the four currencies.

3.5 Dataset Preprocessing

3.5.1 Handle Missing Value

While collecting these four datasets, the timing of the exchange rate data and the VIX index data are not aligned due to the differences in holidays and market closure days across regions. To ensure uniformity in the time selection, we gather the VIX index data based on the timing of the exchange rate data, which results in missing values in the VIX index data. To facilitate the smooth progress of subsequent experiments and improve the accuracy of the experiments, it is necessary first to address these missing values.

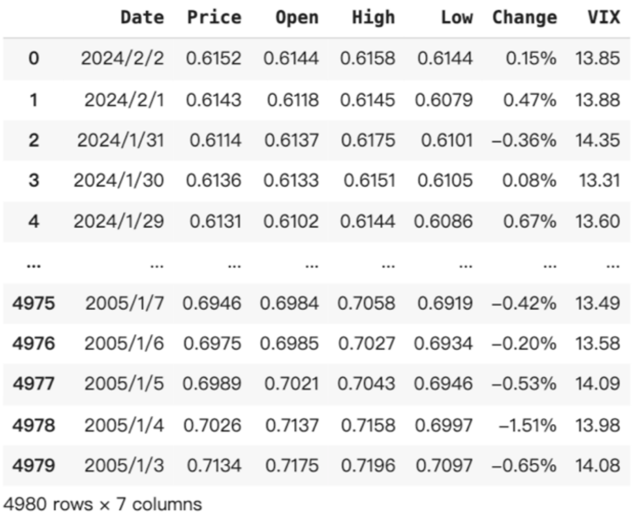
Two primary approaches to managing missing values are deletion and imputation. Deletion requires eliminating entries that have missing attribute values to obtain a complete dataset. Although this method is straightforward, opting for completeness of information by reducing historical data may disrupt the patterns in subsequent data (Pratama, Permanasari, Ardiyanto, & Indrayani, 2016). On the other hand, imputation applies by filling in missing values with typical values to achieve the information. Typically grounded in statistical principles, this approach fills in missing values based on the distribution of values from other entries in the decision table, for example, by supplementing with the average value of the remaining attributes, among other methods. We adopt eq. (3.7) for imputing missing values.

$$X_i = \frac{X_{i-1} + X_{i+1}}{2} \tag{3.7}$$

where X_i defines the data to be imputed, X_{i-1} represents the data from the day before the missing data, and X_{i+1} illustrates the data from the day after the missing data.

3.5.2 Data Normalization

In our datasets, it is required to use close price, open price, high price, low price, change, and VIX index as input data. As shown in Figure 3.5, there is a significant numerical difference between the input data, leading to inconsistent data scales and affecting the learning efficiency of the model. Therefore, data normalization is critical to be performed before training the model.



	Date	Price	Open	High	Low	Change	VIX
0	2024/2/2	0.6152	0.6144	0.6158	0.6144	0.15%	13.85
1	2024/2/1	0.6143	0.6118	0.6145	0.6079	0.47%	13.88
2	2024/1/31	0.6114	0.6137	0.6175	0.6101	−0.36%	14.35
3	2024/1/30	0.6136	0.6133	0.6151	0.6105	0.08%	13.31
4	2024/1/29	0.6131	0.6102	0.6144	0.6086	0.67%	13.60
...	...	...	...	...	...	...	...
4975	2005/1/7	0.6946	0.6984	0.7058	0.6919	−0.42%	13.49
4976	2005/1/6	0.6975	0.6985	0.7027	0.6934	−0.20%	13.58
4977	2005/1/5	0.6989	0.7021	0.7043	0.6946	−0.53%	14.09
4978	2005/1/4	0.7026	0.7137	0.7158	0.6997	−1.51%	13.98
4979	2005/1/3	0.7134	0.7175	0.7196	0.7097	−0.65%	14.08

4980 rows × 7 columns

Figure 3.7 The original dataset of NZD/USD

Data normalization refers to scaling data to a small decimal between (0, 1) and (-1, 1). The purpose is to simplify data comparison and analysis, and also reduce the potential impact of outliers or extreme values on the analysis (Zyprych-Walczak et al., 2015). Additionally, normalized data can be more efficiently plotted, as it eliminates scale differences that could make it difficult to see trends or patterns. Normalized data can accelerate the convergence speed of models, improve data consistency, and convert multi-dimensional values to dimensionless values, making comparison and calculation much more manageable. In this report, we adopt the most common method, min-max standardization, which is scaling the data to the range [0,1] to simplify calculations. The calculation process is as

$$X^{*} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{3.8}$$

Among them, X^*represents the dimensionless data after normalization, X means the observation value, X_{min} denotes the minimum value, and X_{max} tells the maximum value. Denormalization is restoring normalized data to facilitate subsequent data analysis and other operations. The calculation process of denormalization is as

$$y = X^* \cdot (X_{max} - X_{min}) + X_{min} \tag{3.9}$$

Among them, y signifies the data after denormalization, X^* symbolises the dimensionless data after normalization, X means the observation value, X_{min} is the minimum value, and X_{max} illustrates the maximum value.

3.6 Experiment Implementation

We chose Google Colab as our experimental platform, where its powerful GPU makes model training simpler and more effective. In the experiments, three models, Transformer, Informer, and TFT, were utilised for training on four datasets, respectively. The following Table 3.1 presents the relevant parameters and configurations of the experimental environment.

Table 3.1 The configurations of the experiment

Operating System	Ubuntu 22.04.3 LTS
Python	3.10.12
CUDA	12.2
PyTorch	2.2.1
RAM	15G
Hard Disk Drive	225G
GPU	Tesla P100-PCIE-16GB

3.6.1 The Experimental Implementation of Transformer

In the training process of the Transformer model, it is vital to set essential parameters, which are continuously adjusted and optimized. The related parameters are ultimately determined, as shown in Table 3.2. Due to the complexity of the Transformer, we employ a lower learning rate parameter of 0.0005. Although this means that the model learns more slowly, it can help the model adapt more finely to the training data, leading to better stable and accurate predictions. The value of input_window is set to 7, which allows for more suitable capturing of weekly patterns or trends in the data for time series data like exchange rates, a typical setting in financial sequences. We experimented with the multiple training epochs, setting them at 50, 100, 150, and 200, and finally found that 150 is the best, avoiding the risk of overfitting.

Table 3.2 The parameter setting of Transformer

Parameters	**Settings**
input_window	7
batch_size	100
learning_rate	0.0005
epochs	150

3.6.2 The Experimental Implementation of Informer

Unlike the parameter settings of the Transformer, through multiple attempts, we have set the number of epochs to 60. Since the Informer optimizes computational complexity, reducing unnecessary computations and parameter usage, it achieves better results in a shorter training time. The table below details the model parameters of the Informer.

Table 3.3 The parameter setting of Informer

Parameters	**Settings**
sequence_length	64
predict_length	5
batch_size	128
learning_rate	5e-5
epochs	60

3.6.3 The Experimental Implementation of TFT

The model training of TFT is conducted within a PyTorch-lightning framework. In this environment, it is possible to adjust the model's hyperparameters promptly during the data training process. This setup integrates with the Early-Stopping mechanism to obtain an outstanding combination of parameters. For the TFT model, a learning rate of 0.001 is a moderate value that supports balanced training speed and convergence quality. Setting the hidden layer's size to 32 means the TFT model is relatively simple and computationally efficient. Since no overly complex recognition tasks exist, we set the number of attention heads to 1. After multiple debugging and screening rounds, the initial parameters of TFT model are finally determined, as demonstrated in Table 3.4.

Table 3.4 The parameter setting of TFT

Parameters	**Settings**
learning_rate	0.001
hidden_size	32
attention_head_size	1
output_size	8
batch_size	128
epochs	150

3.7 Evaluation Methods

In the experiment of exchange rate prediction, to reflect the reliability of the predictive performance accurately and objectively, we utilise four different evaluation metrics, including root mean square error (RMSE), mean absolute error (MAE), coefficient of determination (R^2), mean absolute percentage error (MAPE). The smaller the RMSE and MAE, the closer the predictions are to the actual values. A larger R^2 indicates a better fit of the model. MAPE provides a comprehensive indication of the model's overall predictive effectiveness.

(1) RMSE

The RMSE is calculated in Equation 3.10, where $\hat{y}_i$ represents the predicted values, and y_i means the corresponding actual values. The closer the predicted values are to the exact values, the closer the RMSE is to 0, indicating better prediction accuracy. Conversely, a higher RMSE signifies a poorer model performance, reflecting larger deviations between predictions and actual observations. RMSE measures the discrepancy between predicted and actual values and is more intuitive on a magnitude scale than MSE.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{3.10}$$

(2) MAE

The MAE is computed as indicated in Equation 3.11, where $\hat{y}_i$ is the predicted value, and y_i is the corresponding actual value, with the range being $[0, +\infty)$. MAE is employed to describe the error between the predicted and actual values. The closer the MAE is to 0, the closer the predicted values are to the exact values. Contrarily, the greater the prediction error, the larger the MAE.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.11}$$

(3) R^2

The coefficient of determination, R^2, is calculated as shown in the formula 3.12, where $\hat{y}_i$ represents the predicted values, y_i means the actual values, and $\bar{y}_i$ is the mean value of y_i. Usually, R^2 values range between [0, 1]. The closer R^2 is to 1, the better the model fits the data, indicating a more substantial explanatory power of the model.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \tag{3.12}$$

(4) MAPE

The calculation process for MAPE is exhibited in the formula, where y_i is the actual value, $\hat{y}_i$ is the predicted value, and n represents the total number of observations. MAPE measures the mean absolute percentage error of the predictions, indicating the degree of deviation between the model's predicted values and the actual values. The smaller the metric, the higher the precision of the forecast and the less deviation from the actual values (Saigal & Mehrotra, 2012).

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \qquad (3.13)$$

Chapter 4
Results

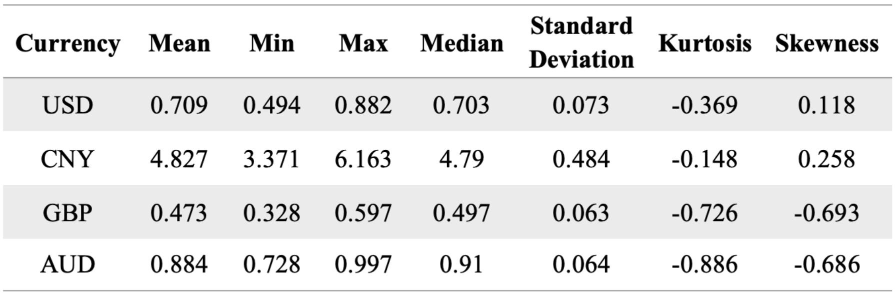
In this chapter, we present descriptive statistics for the four datasets and clarify the experimental results of these datasets across the three models. Additionally, we demonstrate the ablation experiment conducted on the TFT model.

4.1 Data Description

4.1.1 Descriptive analysis

After preprocessing the four datasets, the total number of samples for each is 4,980. To better understand the data's characteristics and distribution features and utilize the relevant data for modelling, it is essential to conduct a descriptive statistical analysis before modelling. Table 4.1 provides the descriptive statistics for the four datasets.

Table 4.1 The descriptive statistics of NZD against four currency exchange rates



Currency	Mean	Min	Max	Median	Standard Deviation	Kurtosis	Skewness
USD	0.709	0.494	0.882	0.703	0.073	-0.369	0.118
CNY	4.827	3.371	6.163	4.79	0.484	-0.148	0.258
GBP	0.473	0.328	0.597	0.497	0.063	-0.726	-0.693
AUD	0.884	0.728	0.997	0.91	0.064	-0.886	-0.686

From Table 4.1, we notice that the standard deviation for NZD/USD is 0.073, indicating that the exchange rate fluctuates within a narrow range. A kurtosis value of -0.369 and a skewness value of 0.118 suggest that the distribution of NZD/USD deviates slightly from a normal distribution, showing slight flatness and right skewness. Still, overall, it is close to symmetry. Compared to NZD/CNY, there is a significant difference between its minimum and maximum values, which are 3.371 and 6.163, respectively. The median of 4.79 is slightly lower than the average, implying a skewed distribution to the right. The standard deviation is 0.484, indicating the volatility is higher than the other three currency pairs. The kurtosis and skewness are -0.148 and 0.258, respectively, indicating a relatively flat and slightly right-skewed distribution.

The statistical results for NZD/GBP show that the average exchange rate for NZD/GBP is 0.473, with a minimum of 0.328 and a maximum of 0.597, revealing a smaller fluctuation range and, hence, a relatively stable exchange rate. The median of 0.497 is very close to the mean, reflecting the central tendency of the data. Its standard deviation of 0.063 is the smallest among the four currency pairs, showing the lowest volatility. The average exchange rate for NZD/AUD is 0.884, with a fluctuation range from 0.728 to 0.997, which is relatively moderate. The median of 0.91 is higher than the average, exhibiting more data points in the higher value range. A standard deviation of 0.064 indicates lower volatility. The kurtosis of -0.886 and skewness of -0.686 present a skewed and peaked distribution, suggesting a frequent occurrence of lower values.

Throughout this detailed analysis, we summarize that these four datasets demonstrate diverse levels of volatility and distribution characteristics. NZD/GBP and NZD/AUD show relatively lower volatility, while NZD/USD and NZD/CNY exhibit higher volatility.

4.1.2 Correlation analysis

Although exchange rates are a type of nonlinear time-series data, analyzing the strength of the relationship between target values and variables is critical before model training. It assists us in better understanding the importance of specific features, thereby enabling feature selection to simplify the model and avoid overfitting. Table 4.2 shows the correlation between the four datasets' opening price, highest price, lowest price, change, and target value closing price.

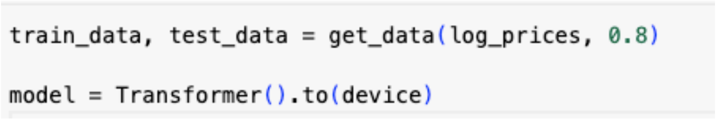
Table 4.2 The correlation between closing price and other four variables in datasets

Currency	Open	High	Low	Change
USD	0.99726	0.99873	0.99877	0.03811
CNY	0.99732	0.99986	0.99986	0.03154
GBP	0.99878	0.99943	0.99942	0.01795
AUD	0.99806	0.99907	0.99903	0.03121

Table 4.2 displays that in the four datasets, the respective “Close” prices have an extremely high correlation with the “Open”, “High”, and “Low” prices, with correlation coefficients above 0.997. This indicates that their relationships are very close during the trading day, almost moving in sync. However, compared to “Change”, the correlation coefficients between the “Closing” prices of the four currencies and “Change” are pretty low, ranging from 0.01795 to 0.03811, but they are still positively correlated. This means

that although “Change” can reflect the trend of the closing price to a certain extent, the direct relationship between “Change” and “Close” price is not as apparent as with other price indicators.

After completing the analysis, we divided the dataset into two parts for the training process of the three models: 80% for training and 20% for testing. Figure 4.1 is the dataset division for the transformer code.



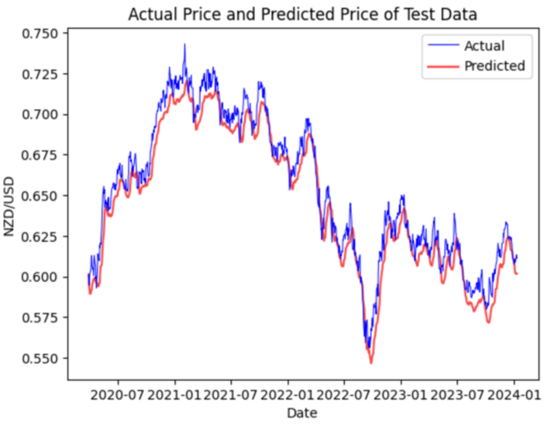
```
train_data, test_data = get_data(log_prices, 0.8)

model = Transformer().to(device)
```

Figure 4.1 The code of Transformer dataset division

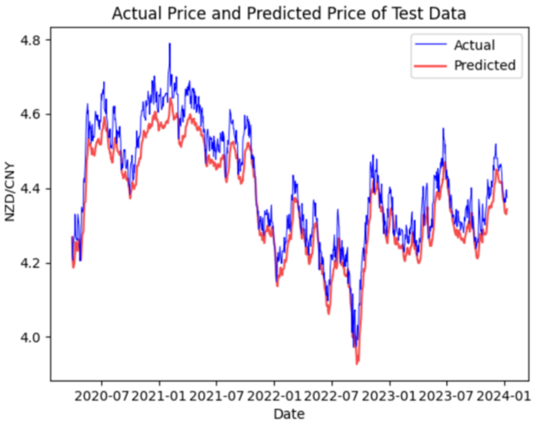
4.2 Experimental Results of Transformer

In this experiment, the initial model we trained on Google Colab for the four different exchange rate datasets was the Transformer model. Considering both the training on the training set and the predictions on the test set, the Transformer has achieved satisfactory results. Figure 4.2 displays the actual and predictive results on the test set of four datasets.

NZD/USD



NZD/CNY



NZD/AUD

NZD/GBP

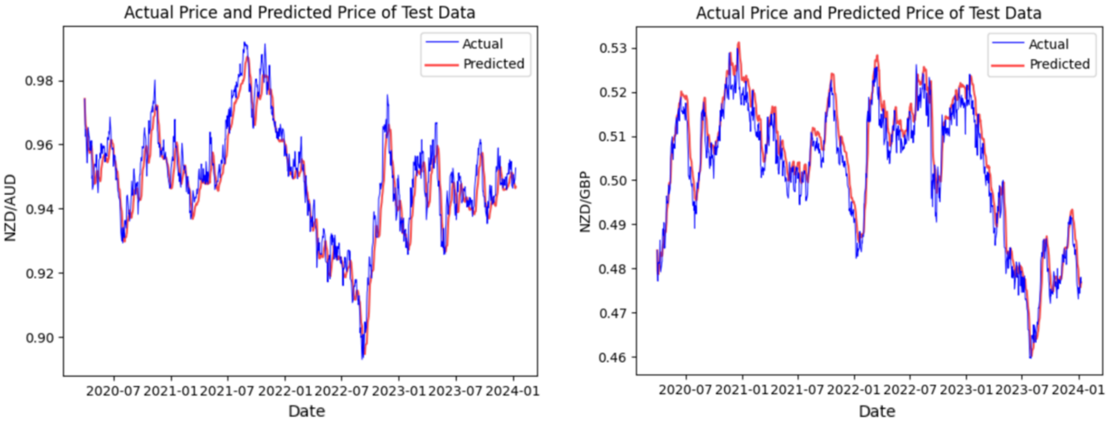


Figure 4.2 The predictive result of each dataset performed on Transformer

From the prediction results related to the four test sets, the trend of NZD/USD is very close to the actual result, reaching highs and lows at almost the same time, and the high degree of overlap between the two lines indicates that the Transformer can effectively capture the trends and seasonal changes of the exchange rate. However, from the NZD/CNY prediction graph, we discover some deviations during periods of high volatility, and the Transformer model has yet to capture the peaks and troughs of the exchange rate perfectly. Despite this, the overall prediction trend still tracks the real exchange rate well. Similar to NZD/AUD, though the figure shows a strong correlation between prediction and reality, the Transformer still underestimates or overestimates the peaks in some intervals. Regarding the NZD/GBP trend, the prediction accuracy is high for most of the timeline, showing that the Transformer is robust. Table 4.3 shows more details of the experimental evaluation results of Transformer.

Table 4.3 The Transformer experiment results with four datasets

Currency	**RMSE**	**MAPE**	**MAE**	**R^2**
NZD/USD	0.011	0.0141	0.0092	0.9369
NZD/CNY	0.0585	0.0113	0.0502	0.8589
NZD/AUD	0.0571	0.0048	0.0046	0.8891
NZD/GBP	0.0051	0.0084	0.0042	0.8841

By evaluating the model with four different indicators, we notice that in the Transformer model's training and prediction for four datasets, NZD/USD exhibits remarkably high precision and reliability. The very low RMSE and MAE values show that the forecast values are extraordinarily close to the actual values. Furthermore, the low MAPE value of 0.0141 verifies that the error percentage is minor, representing an ideal outcome in currency prediction. In comparison, an R^2 value close to 0.94 indicates that the model has strong predictive power and a high degree of explanatory capability regarding the fluctuating exchange rate trends.

Although the MAPE values are relatively low in the NZD/CNY and NZD/AUD predictions, at 0.0113 and 0.0048, respectively, the increased RMSE and MAE indicate that the model faces more significant challenges in forecasting these currency pairs. Possible reasons may include higher market volatility, differences in trading volume, or the datasets' characteristics. Nevertheless, the R^2 values for both currency pairs exceed 0.85, reflecting the Transformer's powerful capability to capture essential information and trends.

Compared to other results, NZD/GBP has the lowest RMSE and MAE, implying that the model can generate highly accurate predictions with minimal error for this currency pair. An R^2 value of 0.8841 demonstrates a satisfactory model fit, and although slightly lower than NZD/USD, it is still an excellent result, given the complexity of the currency market. Figure 4.3 visualizes the experiment result of the Transformer.

The strong performance of Transformer model partly derives from its self-attention mechanism, which allows it to fully consider the influence of other points in time when predicting the exchange rate at any given moment.

In conclusion, the Transformer performs outstandingly across all four datasets, especially in NZD/USD predictions, where it achieves a very high level of accuracy.

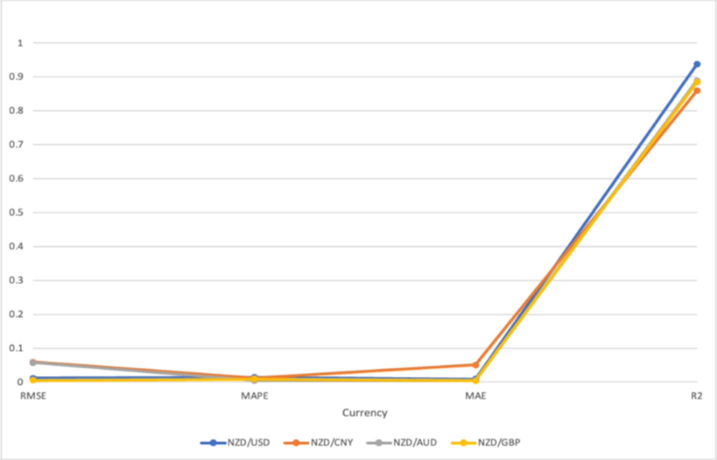
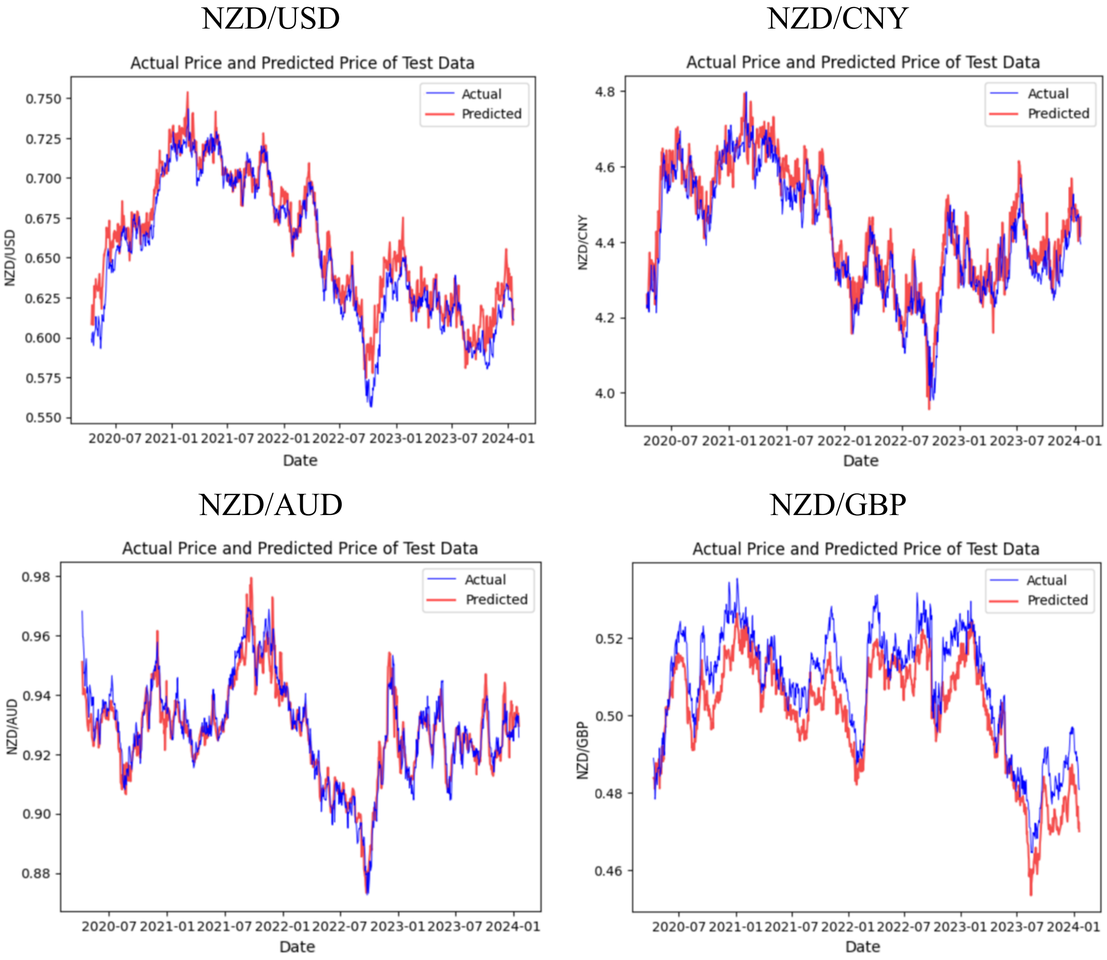


Figure 4.3 Visualisation of the experiment result on Transformer

4.3 Experimental Results of Informer

The second model we conducted in this experiment was the Informer, an advancement based on the Transformer framework. The prediction trend charts are as follows through the training and prediction of four datasets.


NZD/USD

NZD/CNY

NZD/AUD

NZD/GBP

Figure 4.4 The predictive result of each dataset performed on Informer

From the prediction trend charts, the actual and predicted values of NZD/USD are roughly similar, especially at the peaks and troughs. However, between July 2022 and January 2023, there is a relatively large gap between the predicted and actual lines. As for NZD/CNY, the overall prediction for this currency pair also maintained synchronicity, but the deviation at the beginning of 2023 was more extensive than that of NZD/USD. Similar to NZD/USD is NZD/AUD, where the prediction curve of this currency pair closely matches the actual price curve most of the time. Likewise, in a number of intervals, the prediction failed to capture the rapid changes in the exact exchange rate. Slightly different from the trend results of the other three, the trend of NZD/GBP did not match as well as the others, but it also captured the trend of the exchange rate to a certain extent. The below Table 4.4 illustrates the results based on the four evaluation metrics.

Table 4.4 The Informer experiment results with four datasets

Currency	**RMSE**	**MAPE**	**MAE**	**R^2**
NZD/USD	0.012	0.0144	0.0092	0.925
NZD/CNY	0.0579	0.0105	0.0465	0.8631
NZD/AUD	0.0051	0.0042	0.0039	0.9104
NZD/GBP	0.0056	0.0088	0.0045	0.8587

In the NZD/USD results, the Informer model achieved a high level of precision, specifically reflected in the low RMSE value of 0.012. At the same time, the low MAPE and MAE values of 0.0144 and 0.0092, respectively, also demonstrate that the forecast errors are relatively minor. An R^2 value of 0.925 further indicates that the Informer can broadly explain fluctuations in the exchange rate.

In the NZD/CNY results, despite the low MAPE value of 0.0105, which explains a certain degree of accuracy, the higher RMSE and MAE values reveal the challenges faced by the Informer in predicting this currency pair. Nevertheless, an R^2 value exceeding 0.85 means that the Informer can still fit the data reasonably well despite the difficulties.

For NZD/AUD and NZD/GBP, the Informer performed exceptionally well, especially in NZD/AUD, where the very low RMSE and MAE values reflect the Informer's superior performance in terms of prediction accuracy. The MAPE value is nearly zero, almost achieving a perfect prediction effect. This shows that the Informer can accurately predict the exchange rate movements of these two currency pairs, even in the face of fluctuations in exchange rates.

Overall, the Informer model demonstrates adaptability and accuracy under different market conditions in handling the exchange rate predictions of these four datasets. Particularly in predicting NZD/AUD and NZD/USD, it illustrates the advantages of being an improved model based on the Transformer. Although there are challenges in the NZD/CNY predictions, the model can still effectively capture and predict the dynamics of exchange rate changes. The below Figure 4.5 displays the visualization of evaluation results.

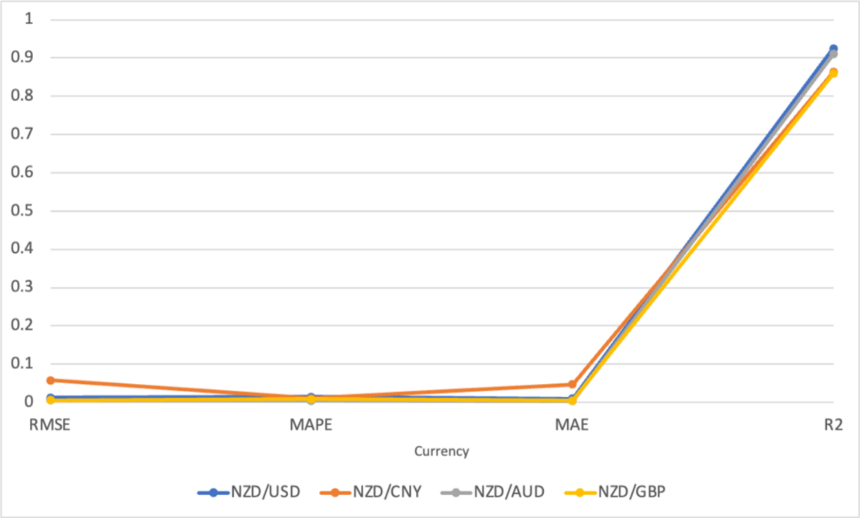
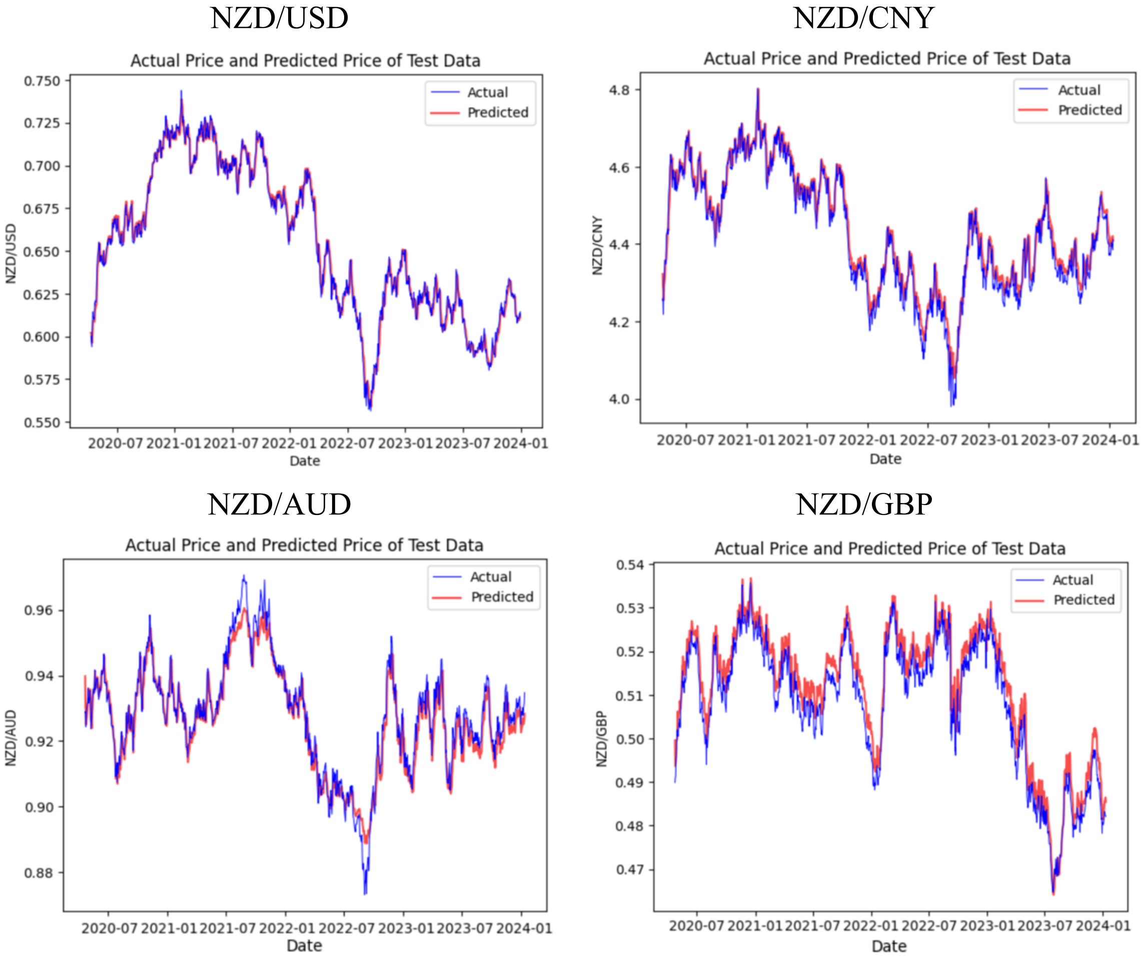


Figure 4.5 Visualisation of the experiment result on Informer

4.4 Experimental Results of TFT

Our third trained model is the TFT, an enhancement of the Transformer model that specializes in processing time-series data. Figure 4.6 exhibits the TFT's prediction trends for the four test sets.


NZD/USD

NZD/CNY

NZD/AUD

NZD/GBP

Figure 4.6 The predictive result of each dataset performed on TFT

From the trend chart of NZD/USD, the prediction curve closely follows the actual price curve most of the time, demonstrating the solid predictive capability of the TFT model for this currency pair, particularly in adapting quickly during significant trend changes. As for the trend charts of the other three currency pairs, we can observe some lag or deviation at critical turning points. Nevertheless, the TFT model generally follows the actual trends well. Table 4.5 below shows the result by using evaluation metrics.

Table 4.5 The TFT experiment results with four datasets

Currency	**RMSE**	**MAPE**	**MAE**	**R^2**
NZD/USD	0.0045	0.0055	0.0035	0.9892
NZD/CNY	0.0312	0.0056	0.0041	0.96
NZD/AUD	0.0041	0.0035	0.0032	0.9381
NZD/GBP	0.0044	0.0075	0.0062	0.9122

The assessment results above show that the predictions for NZD/USD are the best, with shallow RMSE, MAPE, and MAE values of 0.0045, 0.0055, and 0.0035, respectively, revealing that the predicted values are very close to the actual values. The high R^2 value additionally confirms the nearness between the predicted and actual trends of exchange rate fluctuations for this currency pair.

As for the results of NZD/CNY, though the MAPE remains at a low level of 0.0056, a relatively higher RMSE suggests significant deviations between the predicted and actual values at specific time points. However, the high R^2 value of 0.96 indicates that the TFT model can still capture most exchange rate changes.

In the predictions for NZD/AUD, even though the R^2 value is slightly lower than that of NZD/USD, the remarkably low errors indicate the high accuracy of the TFT on this test set.

Although NZD/GBP has the highest MAE among all the currency pairs at 0.0075, this does not mean that the model's overall performance is poor. An R^2 value of 0.9122 points that the model successfully captures most of the dynamics of the pound's exchange rate changes, with a slight decrease in predictive accuracy, possibly due to the complexity of market fluctuations during specific periods.

The TFT model performs reasonably well across all four test sets, especially in predicting NZD/USD and NZD/AUD, showing high accuracy and reliability. Despite the drop in predictive precision for NZD/CNY and NZD/GBP, the R^2 values still demonstrate that the model's predictions are pretty reliable for these currency pairs. Here, Figure 4.7 visualizes the evaluation result of TFT.

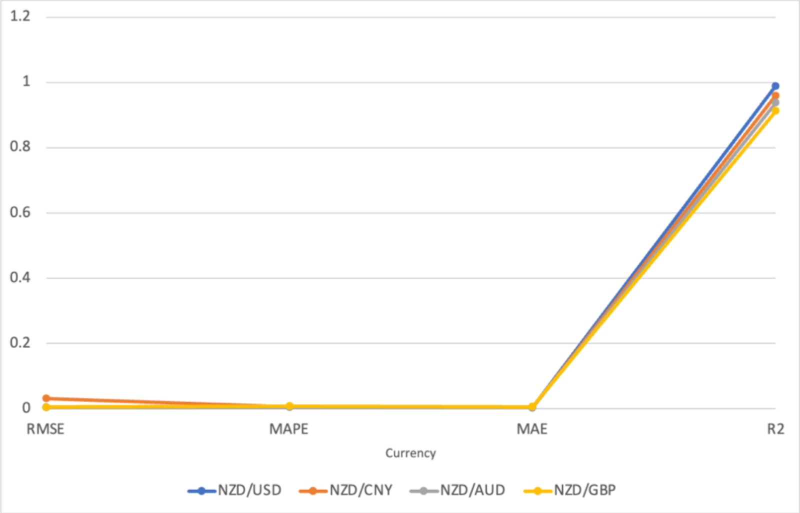
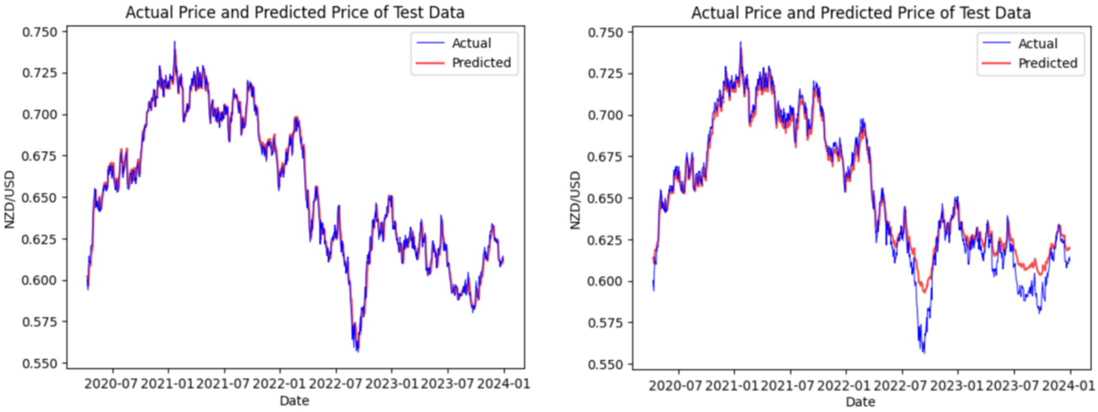


Figure 4.7 Visualisation of the experiment result on TFT

4.5 Ablation Experimental Result Performed on TFT

We conducted an experiment on the NZD/USD dataset by using the TFT model to validate the rationality and effectiveness of the VIX index proposed in section 3.4 for improving prediction accuracy. We assess the results by comparing the prediction trend charts and suggested evaluation metrics. Figure 4.8 compares the predictive trend, including or excluding the VIX index.


Include VIX index

Exclude VIX index

Figure 4.8 The predictive result includes or excludes the VIX index

We discover that although the predicted values are quite consistent with the actual value curves in the trend chart without the VIX index, there are apparent deviations at some extreme points and turning points. The peaks and troughs are not nicely captured, especially in the significant price fluctuation intervals. However, in the chart with the VIX index included, the prediction curve follows the actual price trend more closely, particularly at some extreme points where deviations previously occurred. Table 4.6 below shows the results of the related evaluation criteria.

Table 4.6 The evaluation result of including VIX and excluding VIX

NZD/USD	RMSE	MAPE	MAE	R^2
Include VIX	0.0045	0.0055	0.0035	0.9892
Exclude VIX	0.0312	0.0056	0.0041	0.96

Comparing the results, we find that after integrating the VIX index, the RMSE, MAPE, and MAE values are all very low, at 0.0045, 0.0055, and 0.0035, respectively, indicating high accuracy of the predictions. Furthermore, an R^2 value of 0.9892 signifies an extremely high correlation between the predicted and actual values. In contrast, the RMSE value of 0.0312 for the excluded VIX index is high, suggesting a decrease in precision, and the MAE of 0.0041, slightly higher than when the VIX is included, indicates an increase in prediction error.

The comprehensive assessment results conclude that the TFT model, when including the VIX index, performs better in forecasting the NZD/USD exchange rate, especially regarding accuracy and data fit. The significant reduction in RMSE demonstrates that introducing the VIX index can notably enhance the model's capability to capture market volatility and reduce large prediction deviations. The high R^2 value further validates the effectiveness of the VIX index in improving predictive performance.

Chapter 5
Analysis and Discussions

In this chapter, based on the experimental results, we conduct a detailed comparison and analysis of the performance of the Transformer, Informer, and TFT models, while also highlighting the limitations of this project.

5.1 Analysis and Discussion

To compare the performance of the three models more thoroughly—Transformer, Informer, and TFT, we summarized the evaluation results of each model for the four test sets, taking the average for each evaluation criterion. The results are presented in the following Table 5.1.

Table 5.1 The evaluation result of each model based on the test set

Model	**RMSE**	**MAPE**	**MAE**	$\mathbf{R^2}$
Transformer	0.0329	0.0097	0.0171	0.8922
Informer	0.0201	0.0095	0.016	0.8893
TFT	0.0111	0.0055	0.0043	0.9499

From the table, it is evident that the Transformer model has relatively high RMSE and MAE values. Nevertheless, an R^2 value of 0.8922 indicates a reasonable correlation between the predictions and actual values. Its explanatory power is slightly weaker compared to the other two models. As an improved version of the Transformer, the Informer has lower RMSE and MAE values, at 0.0201 and 0.016, respectively, while its MAPE is 0.0095 and R^2 is 0.8893. This suggests that the Informer performs better than the original Transformer in some respects, especially when dealing with time series data with high volatility. Lastly, the TFT exhibits the best performance among all the models, with an RMSE of only 0.0111, a MAPE of 0.0055, an MAE of 0.0043, and the highest R^2 value of 0.9499. The TFT model integrates various techniques for time series forecasting, including time attention mechanisms and interpretable features, enabling it to excel across all evaluation metrics. Figure 5.1 visualizes the result of three model performances.

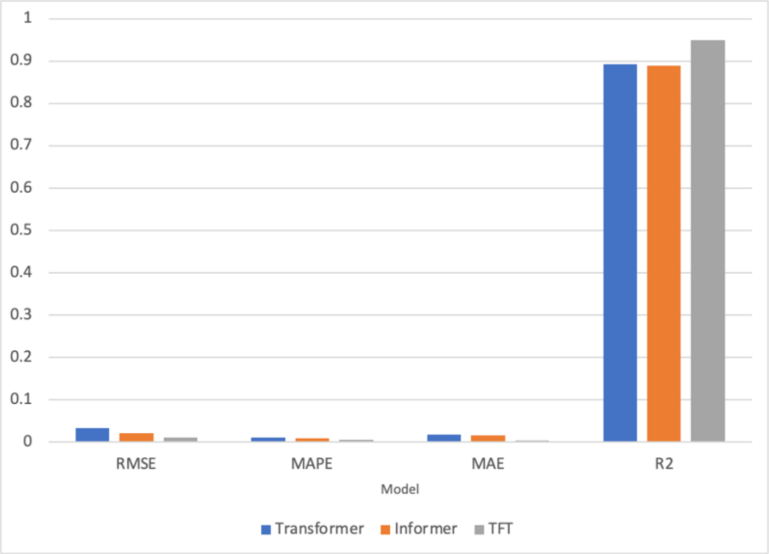


Figure 5.1 The visualization result of the three models' performance

These results imply that the TFT performs best in handling these currency exchange rate prediction tasks, possibly because it is designed to capture complex patterns in time series. The Informer and Transformer also perform well but cannot achieve outstanding results like TFT for this specific task. The differences may derive from the TFT's special treatment of the time dimension in its model architecture and its ability to capture and integrate various factors affecting the predictive variables.

Additionally, from the perspective of training time and model convergence speed, the Informer can reach stable and accurate predictions within a relatively few 60 epochs, which might make it more efficient than the traditional Transformer and the TFT. However, considering the performance after model training, the TFT has demonstrated the highest R^2 value in currency exchange rate predictions. Although the TFT might require a more complex training process, its return on investment in model performance is optimal.

5.2 Limitations of This Project

The limitations of this project mainly fall into the following aspects:

Firstly, the variables selected for this project are limited, only including primary exchange rate data. However, exchange rate trends are influenced by other complex factors, such as national policies, inflation rates, and investor psychological expectations. Thus, the lack of comprehensive feature selection will inevitably lead to unavoidable errors in prediction. Based on this, it is possible to consider incorporating more factors that affect exchange rates in the data selection process.

Secondly, we integrated the VIX index in the NZD/USD analysis and only experimented and verified it within the TFT model. However, more is needed to fully account for the explanatory power of the VIX index on exchange rate predictions. A way to overcome this limitation is to incorporate volatility indexes from different countries and conduct experiments and evaluations on multiple models, comparing and analyzing the significance of these indexes for exchange rate prediction.

Moreover, due to limited time, each model's selection of parameters and functions was primarily based on relevant literature and materials, which may introduce subjectivity and randomness. Therefore, further research and experimentation are needed to select the optimal parameters.

Finally, the experiments in this project are all based on the Transformer model framework. It is vital to conduct experimental comparisons with other cutting-edge models to comprehensively analyze and determine the best model for predicting exchange rates.

Chapter 6
Conclusion and Future Work

The conclusion based on the experiment's results is made in this chapter, and future work is proposed from a variety of aspects.

6.1 Conclusion

This project aims to analyze and discuss the accuracy and performance of models through exchange rate predictions. This report employed three models: Transformer and its advanced versions, Informer and TFT. We collected four exchange rate datasets, NZD/USD, NZD/CNY, NZD/GBP, and NZD/AUD, and applied them to the three models for training and validation. Experiments were conducted on the Google Colab platform, and four evaluation criteria were utilised to analyze and compare the performance of the three models.

All three models achieved satisfactory prediction effects on the four datasets from the results. However, comparisons indicated that the TFT model offered the best performance in exchange rate prediction, especially regarding accuracy and capturing trends in data changes. The Informer balanced efficiency and accuracy, demonstrating excellent predictive capabilities in fewer epochs. This is because of its sparse attention mechanism, which reduces computational complexity. Among the three models, the Transformer performed the least ideally, with relatively higher RMSE and MAE values and the lowest R^2 value.

Moreover, after introducing the VIX index into the TFT model for NZD/USD, we concluded from the comparison that the VIX index is significant for exchange rate prediction. As a measure of market volatility, using the VIX index in the model provided additional information, helping to improve the accuracy of exchange rate predictions.

6.2 Future Work

Although this study has made some progress in exchange rate prediction research, a number of shortages and issues still require further investigation. Therefore, our future research will be conducted in three aspects. Firstly, to enhance the accuracy of our models in predicting exchange rates, it is necessary to include more economic indicators and other relevant factors influencing exchange rates in the data collection process. Secondly, to verify the significance of the VIX index in exchange rate prediction, more models need to be experimented with and tested. Lastly, further research and experimental validation are required to optimise model parameters.

References

Al-Ali, E. M., Hajji, Y., Said, Y., Hleili, M., Alanzi, A. M., Laatar, A. H., & Atri, M. (2023). Solar energy production forecasting based on a hybrid CNN-LSTM-transformer model. *Mathematics, 11*(3), 676.

Alagidede, P., & Ibrahim, M. (2017). On the causes and effects of exchange rate volatility on economic growth: Evidence from Ghana. Journal of African Business, 18(2), 169-193.

Almisshal, B., & Mustafa, E. (2021). Modelling exchange rate volatility using GARCH models. Gazi İktisat ve İşletme Dergisi, 7(1), 1-16.

An, N., Yan, W. (2021) Multitarget tracking using Siamese neural networks. ACM Transactions on Multimedia Computing, Communications and Applications.

An, N. (2020) Anomalies Detection and Tracking Using Siamese Neural Networks. Master's Thesis. Auckland University of Technology, New Zealand.

Bai, X. (2018). Text classification based on LSTM and attention. International Conference on Digital Information Management (ICDIM).

Bi, J., Zhu, Z., & Meng, Q. (2021). Transformer in computer vision. IEEE International Conference on Computer Science, Electronic Information Engineering and Intelligent Control Technology (CEI).

Cavalcante, R. C., Brasileiro, R. C., Souza, V. L., Nobrega, J. P., & Oliveira, A. L. (2016). Computational intelligence and financial markets: A survey and future directions. Expert Systems with Applications, 55, 194-211.

Cao, X. (2022) Pose Estimation of Swimmers from Digital Images Using Deep Learning. Master's Thesis, Auckland University of Technology.

Cao, X. and Yan, W. (2022) Pose estimation for swimmers in video surveillance.

Multimedia Tools and Applications, Springer.

Chambers, J., Yan, W., Garhwal, A., Kankanhalli, M. (2014) Currency security and forensics: A survey. Multimedia Tools and Applications, 74(11), 4013-4043.

Chen, Z. (2023) Real-Time Pose Recognition for Billiard Players Using Deep Learning. Research Report, Auckland University of Technology, New Zealand.

Chen, Z., Yan, W. (2024) Real-time pose recognition for billiard player using deep learning. Deep Learning, Reinforcement Learning and the Rise of Intelligent Systems, pp.188-208, IGI Global.

Chen, S., & Ge, L. (2019). Exploring the attention mechanism in LSTM-based Hong Kong stock price movement prediction. Quantitative Finance, 19(9), 1507-1515.

Du, H., Zhou, S., Yan, W., Wang, S. (2023) Study on DNA storage encoding based IAOA under innovation constraints. Current Issues in Molecular Biology, 45 (4), 3573-3590

De Grauwe, P., & Grimaldi, M. (2018). The exchange rate in a behavioral finance framework.

Eichengreen, B. (2007). China’s exchange rate regime: the long and short of it. In China's Financial Transition at a Crossroads (pp. 314-349). Columbia University Press.

Ernoult, M., Grollier, J., Querlioz, D., Bengio, Y., & Scellier, B. (2019). Updates of equilibrium prop match gradients of backprop through time in an RNN with static input. Advances in Neural Information Processing Systems, 32.

Fayer, G., Lima, L., Miranda, F., Santos, J., Campos, R., Bignoto, V., . . . Capriles, P. (2023). A temporal fusion transformer deep learning model for long-term streamflow forecasting: a case study in the funil reservoir, Southeast Brazil. Knowledge-Based Engineering and Sciences, 4(2), 73-88.

Fu, Y., Nguyen, M., Yan, W. (2022) Grading methods for fruit freshness based on deep learning. Springer Nature Computer Science.

Fu, Y. (2020) Fruit Freshness Grading Using Deep Learning. Master's Thesis. Auckland University of Technology, New Zealand.

Gao, X., Nguyen, M., Yan, W. (2021) Face image inpainting based on generative adversarial network. International Conference on Image and Vision Computing New Zealand.

Gao, X., Nguyen, M., Yan, W. (2022) A face image inpainting method based on autoencoder and adversarial generative networks. Pacific-Rim Symposium on Image and Video Technology.

Gao, X. (2022) A Method for Face Image Inpainting Based on Generative Adversarial Networks. Master's Thesis, Auckland University of Technology, New Zealand.

Gao, X., Nguyen, M., Yan, W. (2023) Human face mask detection using YOLOv7+CBAM in deep learning. Handbook of Research on AI and ML for Intelligent Machines and Systems

Gao, X., Nguyen, M., Yan, W. (2023) Enhancement of human face mask detection performance by using ensemble learning models. PSIVT.

Gao, X., Nguyen, M., Yan, W. (2023) A high-accuracy deformable model for human face mask detection. PSIVT

Gharehbaghi, A., Ghasemlounia, R., Ahmadi, F., & Albaji, M. (2022). Groundwater level prediction with meteorologically sensitive Gated Recurrent Unit (GRU) neural networks. Journal of Hydrology, 612, 128262.

Gong, M., Zhao, Y., Sun, J., Han, C., Sun, G., & Yan, B. (2022). Load forecasting of district heating system based on Informer. Energy, 253, 124179.

Gu, D., Nguyen, M., Yan, W. (2016) Cross models for twin recognition. International Journal of Digital Crime and Forensics 8 (4), 26-36.

Gu, Q., Yang, J., Kong, L., Yan, W., Klette, R. (2017) Embedded and real-time vehicle detection system for challenging on-road scenes. Optical Engineering, 56 (6), 063102.

Gu, Q., Yang, J., Yan, W., Klette, R. (2017) Integrated multi-scale event verification in an augmented foreground motion space. Pacific-Rim Symposium on Image and Video Technology (pp.488-500)

Gu, Q., Yang, J., Yan, W., Li, Y., Klette, R. (2017) Local Fast R-CNN flow for object-centric event recognition in complex traffic scenes. Pacific-Rim Symposium on Image and Video Technology (pp.439-452)

Guo, Q., Qiu, X., Liu, P., Shao, Y., Xue, X., & Zhang, Z. (2019). Star-Transformer. NAACL-HLT (pp. 1315-1325).

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., . . . Xu, Y. (2022). A survey on vision transformer. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(1), 87-110.

Herrera, A., Beck, A., Bell, D., Miller, P., Wu, Q., Yan, W. (2008) Behavior analysis and prediction in image sequences using rough sets. International Machine Vision and Image Processing Conference (pp.71-76)

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.

Hori, T., Cho, J., & Watanabe, S. (2018). End-to-end speech recognition with word-based RNN language models. IEEE Spoken Language Technology Workshop (SLT).

Hu, X. (2021). Stock price prediction based on temporal fusion transformer. International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI).

Hu, Z., Zhao, Y., & Khushi, M. (2021). A survey of forex and stock price prediction using deep learning. Applied System Innovation, 4(1), 9.

Hussain, B., Afzal, M. K., Ahmad, S., & Mostafa, A. M. (2021). Intelligent traffic flow prediction using optimized GRU model. IEEE Access, 9, 100736-100746.

Huy, P. C., Minh, N. Q., Tien, N. D., & Anh, T. T. Q. (2022). Short-term electricity load forecasting based on temporal fusion transformer model. IEEE Access, 10, 106296-106304.

Islam, M. S., & Hossain, E. (2021). Foreign exchange currency rate prediction using a GRU-LSTM hybrid network. Soft Computing Letters, 3, 100009.

Ji, H., Liu, Z., Yan, W., Klette, R. (2019) Early diagnosis of Alzheimer's disease based on selective kernel network with spatial attention. Asian Conference on Pattern Recognition.

Ji, H., Yan, W., Klette, R. (2019) Early diagnosis of Alzheimer's disease using deep learning. ACM ICCCV.

Jiao, Y., Weir, J., Yan, W. (2011) Flame detection in surveillance. Journal of Multimedia 6 (1).

Kieran, D., Yan, W. (2010) A framework for an event-driven video surveillance system. Advanced Video and Signal Based Surveillance (AVSS).

Lahmiri, S. (2017). Modeling and predicting historical volatility in exchange rate markets. Physica A: Statistical Mechanics and Its Applications, 471, 387-395.

Lara-Benítez, P., Gallego-Ledesma, L., Carranza-García, M., & Luna-Romera, J. M. (2021). Evaluation of the transformer architecture for univariate time series forecasting. 19th Conference of the Spanish Association for Artificial Intelligence, CAEPIA 2020/2021, Málaga, Spain.

Li, C., Yan, W. (2021) Braille recognition using deep learning. International Conference on Control and Computer Vision.

Li. C. (2022) Special Character Recognition Using Deep Learning. Master's Thesis

Auckland University of Technology, New Zealand.

Li, R., Nguyen, M., Yan, W. (2017) Morse codes enter using finger gesture recognition. International Conference on Digital Image Computing: Techniques and Applications.

Li, Y., Ming, Y., Zhang, Z., Yan, W., Wang, K. (2021) An adaptive ant colony algorithm for autonomous vehicles global path planning. International Conference on Computer Supported Cooperative Work in Design.

Li, S., Jin, X., Xuan, Y., Zhou, X., Chen, W., Wang, Y.-X., & Yan, X. (2019). Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. Advances in Neural Information Processing Systems, 32.

Li, S., Li, W., Cook, C., Zhu, C., & Gao, Y. (2018). Independently recurrent neural network (indrnn): Building a longer and deeper RNN. IEEE Conference on Computer Vision and Pattern Recognition.

Liang, C., Lu, J., Yan, W. (2022) Human action recognition from digital videos based on deep learning. ACM ICCCV.

Liang, C., Yan, W. (2024) Human action recognition based on YOLOv7. Deep Learning, Reinforcement Learning, and the Rise of Intelligent Systems, pp.126-145, Chapter 6, IGI Global.

Liang, S. (2021) Multi-language Datasets for Speech Recognition Based on the End-to-End Framework. Master’s Thesis. Auckland University of Technology, New Zealand.

Liang, S., Yan, W. (2022) A hybrid CTC+Attention model based on end-to-end framework for multilingual speech recognition. Springer Multimedia Tools and Applications.

Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 37(4), 1748-1764.

Lin, Z. (2018). Modelling and forecasting the stock market volatility of SSE Composite Index using GARCH models. Future Generation Computer Systems, 79, 960-972.

Liu, C., Yan, W. (2020) Gait recognition using deep learning. Handbook of Research on Multimedia Cyber Security (pp.214-226)

Liu, J., Yan, W. (2022) Crime prediction from surveillance videos using deep learning. Aiding Forensic Investigation Through Deep Learning and Machine Learning Frameworks. IGI Global.

Liu, W., Miller, P., Ma, J., Yan, W. (2009) Challenges of distributed intelligent surveillance system with heterogenous information. Procs. of QRASA (pp.69-74)

Liu, W., Li, Y., Tomasetto, F., Yan, W., et al. (2022) Non-destructive measurements of Toona sinensis chlorophyll and nitrogen content under drought stress using near infrared spectroscopy. Frontiers in Plant Science.

Liu, X., Nguyen, M., Yan, W. (2019) Vehicle-related scene understanding using deep learning. Asian Conference on Pattern Recognition.

Liu, X. (2019) Vehicle-Related Scene Understanding Using Deep Learning. Master’s Thesis, Auckland University of Technology, New Zealand.

Liu, X., Yan, W. (2020) Vehicle-related scene segmentation using CapsNets. International Conference on Image and Vision Computing New Zealand.

Liu, X., Yan, W. (2022) Depth estimation of traffic scenes from image sequence using deep learning. Pacific-Rim Symposium on Image and Video Technology.

Liu, X., Yan, W. (2022) Vehicle-related distance estimation using customized YOLOv7. International Conference on Image and Vision Computing New Zealand (IVCNZ)

Liu, X., Yan, W. Kasabov, N. (2023) Moving vehicle tracking and scene understanding: A hybrid approach. Multimedia Tools and Applications.

Liu, X., Yan, W. (2024) Vehicle detection and distance estimation using improved YOLOv7 model. Deep Learning, Reinforcement Learning and the Rise of Intelligent Systems, pp. 173-187, Chapter 9, IGI Global.

Liu, Y., Nand, P., Hossain, A., Nguyen, M., Yan, W. (2023) Sign language recognition from digital videos using feature pyramid network with detection Transformer. Multimedia Tools and Applications.

Liu, X. (2024) Vehicle-Related Scene Understanding Using Deep Learning. PhD Thesis, Auckland University of Technology, New Zealand.

Lu, J. (2016) Empirical Approaches for Human Behavior Analytics. Master's Thesis. Auckland University of Technology, New Zealand.

Lu, J., Shen, J., Yan, W., Boris, B. (2017) An empirical study for human behaviors analysis. International Journal of Digital Crime and Forensics 9 (3), 11-17.

Lu, J., Nguyen, M., Yan, W. (2018) Pedestrian detection using deep learning. IEEE AVSS.

Lu, J., Nguyen, M., Yan, W. (2020) Human behavior recognition using deep learning. International Conference on Image and Vision Computing New Zealand.

Lu, J., Nguyen, M., Yan, W. (2020) Comparative evaluations of human behavior recognition using deep learning. Handbook of Research on Multimedia Cyber Security, 176-189.

Lu, J., Nguyen, M., Yan, W. (2021) Sign language recognition from digital videos using deep learning methods. International Symposium on Geometry and Vision.

Lu, J. (2021) Deep Learning Methods for Human Behavior Recognition. PhD Thesis. Auckland University of Technology, New Zealand.

Ma, J., Liu, W., Miller, P., Yan, W. (2009) Event composition with imperfect information for bus surveillance. IEEE International Conference on Advanced Video and Signal Based Surveillance.

Mehtab, S., Yan, W. (2021) FlexiNet: Fast and accurate vehicle detection for autonomous vehicles-2D vehicle detection using deep neural network. International Conference on Control and Computer Vision.

Mehtab, S., Yan, W. (2022) Flexible neural network for fast and accurate road scene perception. Multimedia Tools and Applications.

Mehtab, S. Yan, W., Narayanan, A. (2022) 3D vehicle detection using cheap LiDAR and camera sensors. International Conference on Image and Vision Computing New Zealand.

Mehtab, S. (2022) Deep Neural Networks for Road Scene Perception in Autonomous Vehicles Using LiDARs and Vision Sensors. PhD Thesis, Auckland University of Technology, New Zealand.

Munir, H. S., Ren, S., Mustafa, M., Siddique, C. N., & Qayyum, S. (2021). Attention Based GRU-LSTM for software defect prediction. PLOS One, 16(3), e0247444.

Niitsuma, M., Tomita, Y., Yan, W., Bell, D. (2018) Towards musicologist-driven mining of handwritten scores. IEEE Intelligent Systems.

Niitsuma, M., Tomita, Y., Yan, W., Bell, D. (2011) Classifying Bach's handwritten C-Clefs. International Society for Music Information Retrieval Conference (ISMIR 2011).

Niu, Z., Zhong, G., & Yu, H. (2021). A review on the attention mechanism of deep learning. Neurocomputing, 452, 48-62.

Pahlavani, M., & Roshan, R. (2015). The comparison among ARIMA and hybrid ARIMA-GARCH models in forecasting the exchange rate of Iran. International Journal of Business and Development Studies, 7(1), 31-50.

Patel, P. J., Patel, N. J., & Patel, A. R. (2014). Factors affecting currency exchange rate, economical formulas and prediction models. International Journal of Application or Innovation in Engineering & Management, 3(3), 53-56.

Pearce, D. K., & Solakoglu, M. N. (2007). Macroeconomic news and exchange rates. Journal of International Financial Markets, Institutions and Money, 17(4), 307-325.

Pirani, M., Thakkar, P., Jivrani, P., Bohara, M. H., & Garg, D. (2022). A comparative analysis of ARIMA, GRU, LSTM and BiLSTM on financial time series forecasting. IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE).

Pradeepkumar, D., & Ravi, V. (2018). Soft computing hybrids for FOREX rate prediction: A comprehensive review. Computers & Operations Research, 99, 262-284.

Pratama, I., Permanasari, A. E., Ardiyanto, I., & Indrayani, R. (2016). A review of missing values handling methods on time-series data. International Conference on Information Technology Systems and Innovation (ICITSI).

Qonita, A., Pertiwi, A. G., & Widiyaningtyas, T. (2017). Prediction of rupiah against us dollar by using arima. International Conference on Electrical Engineering, Computer Science and Informatics (EECSI).

Ranjit, S., Shrestha, S., Subedi, S., & Shakya, S. (2018). Foreign rate exchange prediction using neural network and sentiment analysis. International Conference on Advances in Computing, Communication Control and Networking (ICACCCN).

Rout, A. K., Dash, P. K., Dash, R., & Bisoi, R. (2017). Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach. Journal of King Saud University-Computer and Information Sciences, 29(4), 536-552.

Saigal, S., & Mehrotra, D. (2012). Performance comparison of time series data using predictive data mining techniques. Advances in Information Mining, 4(1), 57-66.

Shen, D., Xin, C., Nguyen, M., Yan, W. (2018) Flame detection using deep learning. International Conference on Control, Automation and Robotics.

Shen, Y., Yan, W. (2019) Blind spot monitoring using deep learning. International Conference on Image and Vision Computing New Zealand.

Song, C., He, L., Yan, W., Nand, P. (2019) An improved selective facial extraction model for age estimation. International Conference on Image and Vision Computing New Zealand.

Song, Y., Luktarhan, N., Shi, Z., & Wu, H. (2023). TGA: A novel network intrusion detection method based on TCN, BiGRU and attention mechanism. Electronics, 12(13), 2849.

Sridhar, S., & Sanagavarapu, S. (2021). Multi-head self-attention transformer for dogecoin price prediction. International Conference on Human System Interaction (HSI).

Sukhbaatar, S., Grave, É., Bojanowski, P., & Joulin, A. (2019, July). Adaptive Attention Span in Transformers. Association for Computational Linguistics (pp. 331-335).

Sun, Y., Hou, L., Lv, Z., & Peng, D. (2022). Informer-based intrusion detection method for network attack of integrated energy system. IEEE Journal of Radio Frequency Identification, 6, 748-752.

Sutcu, M., & Gulbahar, I. T. (2023). Long term currency forecast with multiple trend corrected exponential smoothing with shifting lags. International Journal of Industrial Optimization, 47-57.

Tay, Y., Dehghani, M., Bahri, D., & Metzler, D. (2022). Efficient transformers: A survey. ACM Computing Surveys, 55(6), 1-28.

Tong, D., Yan, W. (2022) Visual watermark identification from the transparent window of currency by using deep learning. Applications of Encryption and Watermarking for Information Security, pp.59-77, Chapter 3. IGI Global.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.

Wang, G., Wu, X., Yan, W. (2017) The state-of-the-art technology of currency identification: A comparative study. International Journal of Digital Crime and Forensics 9 (3), 58-72.

Wang, H., Yan, W. (2022) Face detection and recognition from distance based on deep learning. Aiding Forensic Investigation Through Deep Learning and Machine Learning Framework, pp.144-160, IGI Global.

Wang, J., Yan, W. (2016) BP-neural network for plate number recognition. International Journal of Digital Crime and Forensics (IJDCF) 8 (3), 34-45.

Wang, J. (2016) Event-driven Traffic Ticketing System. Master's Thesis, Auckland University of Technology, New Zealand.

Wang, J., Ngueyn, M., Yan, W. (2017) A framework of event-driven traffic ticketing system. International Journal of Digital Crime and Forensics (IJDCF) 9 (1), 39-50.

Wang, T., Zhu, Y., Jin, L., Luo, C., Chen, X., Wu, Y., . . . Cai, M. (2020). Decoupled attention network for text recognition. AAAI Conference on Artificial Intelligence.

Wang, X., Pi, D., Zhang, X., Liu, H., & Guo, C. (2022). Variational transformer-based anomaly detection approach for multivariate time series. Measurement, 191, 110791.

Wu, H., Xu, J., Wang, J., & Long, M. (2021). AutoFormer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in Neural

Information Processing Systems, 34, 22419-22430.

Xiang, Y., Yan, W. (2021) Fast-moving coin recognition using deep learning. Springer Multimedia Tools and Applications.

Xiao, B., Nguyen, M., Yan, W. (2023) Apple ripeness identification from digital images using transformers. Multimedia Tools and Applications, Springer Science and Business Media LLC.

Xiao, B., Nguyen, M., Yan, W. (2023) Fruit ripeness identification using transformers. Applied Intelligence, Springer Science and Business Media LLC.

Xiao, B., Nguyen, M., Yan, W. (2023) A mixture model for fruit ripeness identification in deep learning. Handbook of Research on AI and ML for Intelligent Machines and Systems, pp.1-16, Chapter 16, IGI Global.

Xiao, B. (2024) Fruit Ripeness Identification from Digital Images Using Deep Learning. PhD Thesis, Auckland University of Technology, New Zealand.

Xin, C. (2020) Detection and Recognition for Multiple Flames Using Deep Learning. Master's Auckland University of Technology, New Zealand.

Xin, C., Nguyen, M., Yan, W. (2020) Multiple flames recognition using deep learning. Handbook of Research on Multimedia Cyber Security, 296-307.

Xu, H., Xu, C., Sun, Y., Peng, J., Tian, W., & He, Y. (2023). Exchange rate forecasting based on integration of Gated Recurrent Unit (GRU) and CBOE Volatility Index (VIX). Computational Economics. doi:10.1007/s10614-023-10484-2

Xu, Y., Han, L., Wan, L., & Yin, L. (2019). Dynamic link between oil prices and exchange rates: A non-linear approach. Energy Economics, 84, 104488.

Yamak, P. T., Yujian, L., & Gadosey, P. K. (2019). A comparison between arima, lstm, and GRU for time series forecasting. International Conference on Algorithms, Computing and Artificial Intelligence.

Yan, W. (2019) *Introduction to Intelligent Surveillance: Surveillance Data Capture, Transmission, and Analytics*. Springer Nature.

Yan, W. (2023) Computational Methods for Deep Learning: Theory, Algorithms, and Implementations. Springer Nature.

Yan, W., Nguyen, M., Stommel, M. (2023) International Conference on Image and Vision Computing (IVCNZ 2022), Springer Nature LNCS 13836

Yan, W., Nguyen, M., Stommel, M. (2024) Pacific Conference on Image and Video Technology (PSIVT 2023), Springer Nature LNCS 14403

Yang, Z., Cen, J., Liu, X., Xiong, J., & Chen, H. (2022). Research on bearing fault diagnosis method based on transformer neural network. Measurement Science and Technology, 33(8), 085111.

Yıldıran, C. U., & Fettahoğlu, A. (2017). Forecasting USDTRY rate by ARIMA method. Cogent Economics & Finance, 5(1), 1335968.

Yu, W., Kim, I. Y., & Mechefske, C. (2021). Analysis of different RNN autoencoder variants for time series classification and machine prognostics. Mechanical Systems and Signal Processing, 149, 107322.

Yu, Z. (2021) Deep Learning Methods for Human Action Recognition. Master's Thesis, Auckland University of Technology, New Zealand.

Yu, Z., Yan, W. (2020) Human action recognition using deep learning methods. International Conference on Image and Vision Computing New Zealand.

Zhang, H., Zou, Y., Yang, X., & Yang, H. (2022). A temporal fusion transformer for short-term freeway traffic speed multistep prediction. Neurocomputing, 500, 329-340.

Zhang, Q. (2018) Currency Recognition Using Deep Learning. Master's Thesis, Auckland University of Technology, New Zealand.

Zhang, Q., Yan, W. (2018) Currency detection and recognition based on deep learning. IEEE International Conference on Advanced Video and Signal Based Surveillance.

Zhang, Q., Yan, W., Kankanhalli, M. (2019) Overview of currency recognition using deep learning. Journal of Banking and Financial Technology, 3 (1), 59–69.

Zhang, Y., Yan, W., Narayanan, A. (2017) A virtual keyboard implementation using finger recognition. International Conference on Image and Vision Computing New Zealand.

Zhang, Y. (2016) A Virtual Keyboard Implementation Based on Finger Recognition. Master's Thesis, Auckland University of Technology, New Zealand.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. AAAI Conference on Artificial Intelligence.

Zhou, H., Nguyen, M., Yan, W. (2023) Computational analysis of table tennis matches from real-time videos using deep learning. PSIVT 2023.

Zhu, W., Peng, B., Yan, W. (2024) Dual knowledge distillation on multiview pseudo labels for unsupervised person re-identification. IEEE Transactions on Multimedia.

Zhu, Y., Peng, B., Yan, W. (2022) Ski fall detection from digital images using deep learning. ACM ICCCV.

Zhu, Y., Yan, W. (2022) Image-based storytelling using deep learning. ACM ICCCV.

Zyprych-Walczak, J., Szabelska, A., Handschuh, L., Górczak, K., Klamecka, K., Figlerowicz, M., & Siatkowski, I. (2015). The impact of normalization methods on RNA-Seq data analysis. BioMed Research International, 2015.