https://dergipark.org.tr/tr/pub/igdirsosbilder

Iğdır Üniversitesi Sosyal Bilimler Dergisi

E-ISSN: 2147-6152

Yıl 13, Sayı 36, Mayıs 2024

[Araştırma Makalesi] **Atıf/Citation**: Gür, Y. Y. (2024). "Consumer Price Index Forecasting in Turkey: A Comparison of Deep Learning and Machine Learning Approaches", *Iğdır Üniversitesi Sosyal Bilimler Dergisi*. (36): s. 312-344. **Gönderim Tarihi**:5.11.2023 **Kabul ve Yayın Tarihi**: 10.05.2024-31.05.2024 **DOI**: 10.54600/igdirsosbilder.1386274

Yazar/Author

Yunus Emre GÜR*

Makale Adı/Article Name

Consumer Price Index Forecasting in Turkey: A Comparison of Deep Learning and Machine Learning Approaches

Türkiye'de Tüketici Fiyat Endeksi Tahmini: Derin Öğrenme ve Makine Öğrenme Yaklaşımlarının Karşılaştırılması

ABSTRACT

This study aims to investigate the effectiveness of deep learning and machine learning algorithms on consumer price index (CPI) forecasting using monthly consumer price index data and 5 independent variables (employment rate, average dollar rate, producer price index, Brent oil prices, and consumer loan interest rate) for the period January 2005–June 2023. To this end, different deep learning and machine learning models such as Long and Short Term Memory (LSTM), Gated Recurrent Unit (GRU), Random Forest, Artificial Neural Network, and K-Nearest Neighbors are used for CPI forecasting, and their forecasting performance is evaluated with RMSE, MSE, MAE, MAPE, and R^2 error statistics. The results show that the GRU model outperforms the LSTM, Random Forest, Neural Network, and K-Nearest Neighbors models. The GRU model outperformed the other four models in RMSE, MSE, MAE, MAPE, and R^2 values.

In addition, in CPI forecasting, it was observed that the GRU deep learning model can be used effectively in the inflation domain. These results provide an effective way to forecast CPI, which is an important component of economic forecasting and inflation management. Moreover, the SHAP (Shapley Additive Explanations) analysis applied for the training and testing phases of the GRU model clarified the contribution of each feature in the GRU model's forecasts and allowed us to understand the effects of variables on the model's forecasting ability. This analysis helped to better understand the success of the GRU model and the economic dynamics underlying the forecasts. This study demonstrates the applicability of the GRU deep learning model in economics and finance and provides a valuable tool for economic and financial decision-makers.

Keywords: Inflation Forecasting, Deep Learning, Machine Learning and Gated Recurrent Unit

ÖZ

Bu çalışma, 2005 Ocak – 2023 Haziran dönemine ait aylık tüketici fiyat endeksi verileri ve 5 adet bağımsız değişken (İşsizlik Oranı, Ortalama Dolar Kuru, Üretici Fiyat Endeksi, Brent Petrol Fiyatları, İhtiyaç Kredisi Faiz Oranı) verilerini kullanarak derin öğrenme ve makine öğrenme algoritmalarının Tüketici Fiyat Endeksi (TÜFE) tahmini üzerindeki etkinliğini araştırmayı amaçlamaktadır. Bu doğrultuda, TÜFE tahmini için Uzun ve Kısa Süreli Bellek (LSTM), Geçitli Tekrarlayan Birim (GRU), Rastgele Orman, Yapay Sinir Ağı ve K-En Yakın Komşular gibi farklı derin öğrenme ve makine öğrenme modelleri kullanılmış ve tahmin performansları RMSE, MSE, MAE, MAPE ve R^2 hata istatistikleri ile değerlendirilmiştir.

Sonuçlar, GRU modelinin LSTM, Rastgele Orman, Yapay Sinir Ağı ve K-En Yakın Komşular modellerine kıyasla daha başarılı olduğunu göstermiştir. GRU modeli, diğer dört modele göre RMSE, MSE, MAE, MAPE ve R^2 değerlerinde daha iyi performans göstermiştir. Ek olarak, TÜFE tahmininde, enflasyon alanında GRU derin öğrenme modelinin etkin bir şekilde kullanılabileceği gözlemlenmiştir. Bu sonuçlar, ekonomik tahminlerin ve enflasyonun yönetilmesinin önemli bir bileşeni olan TÜFE tahmininde etkili bir yol sunmaktadır. Ayrıca, GRU modelinin eğitim ve test aşamaları için uygulanan SHAP (Shapley Additive exPlanations) analizi, GRU model tahminlerindeki her bir özelliğin katkısını açıklığa kavuşturmuş ve modelin tahmin yeteneği üzerindeki değişkenlerin etkilerini anlamaya olanak sağlamıştır. Bu analiz, GRU modelinin başarısını ve tahminlerin altında yatan ekonomik dinamikleri daha iyi anlamaya yardımcı olmuştur. Bu çalışma, GRU derin öğrenme modelinin ekonomi ve finans alanında uygulanabilirliğini göstermektedir ve ekonomik ve finansal karar vericiler için değerli bir araç sunmaktadır.

Anahtar Kelimeler: Tüketici Fiyat Endeksi Tahmini, Derin Öğrenme, Makine Öğrenimi ve Geçitli Tekrarlayan Birim

* Arş. Gör. Dr., Fırat Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, Yönetim Bilişim Sistemleri Bölümü, yegur@firat.edu.tr

Introduction

Movements in the Producer and Consumer Price Indexes are significant indicators of the inflation rate in countries. These indexes could take a similar route over time or they might go in distinct directions. Policymakers can evaluate future inflation rates, the structure of inflation, and its origins by looking at how price indices have changed over time (Erdem and Yamak, 2014:1-2).

The consumer price index is a well-known metric associated with the concept of inflation that is supposed to reflect the price of consumers' completed goods. As such, the consumer price index is the most often utilized indicator in studies examining the link between inflation and consumer confidence. In the economics literature, inflation is a contentious topic, and its most notable short-term impact is frequently seen in expenditures. But whether or not spending increases or decreases depends on a number of variables, including consumer confidence in the market (Tunalı and Özkan, 2016: 55).

For all participants in the economy, inflation rates are one of the most important macroeconomic indicators. Because of this, economic decision-makers work hard to track and predict the inflation rate. Since economic decisions are based on predicted future changes as well as present events, effective inflation forecasting is crucial for making the right choices. The monetary and fiscal policies of governments and central banks are influenced by inflation expectations, which also decide how these institutions will affect the economy. Additionally, inflation predictions have an impact on a wide range of company actions, from contracts to investment and finance choices. Future inflation estimates that are accurate assist consumers, businesses, and governments in making better decisions and raising their chances of success. Efficiency and productivity consequently rise, which benefits the economy (Bayramoğlu and Öztürk, 2017: 761).

The Consumer Price Index (CPI) is affected by a number of economic factors. These factors include the unemployment rate, the average dollar exchange rate, the producer price index, Brent oil prices, and interest rates on personal loans. In particular, it has been noted that changes in Brent oil prices have a direct impact on the CPI. Studies have shown that fluctuations in oil prices can potentially have negative consequences for CPI (Jang and Beruvides, 2020). Moreover, when the pass-through effects of changes in global oil prices on the producer price index (PPI) and CPI are analysed, it is determined that demand shocks in the crude oil market make significant contributions to fluctuations in inflation (Sun et al., 2022). The relationship between oil prices and inflation is not limited to oil-importing countries. In the case of Asian countries, oil price shocks have significant effects on economic activity and price indices, especially in the short run and in local currencies (Cuñado and Gracia, 2005). In developing countries, the impact of oil price shocks on CPI differs depending on the source of these changes (Sakashita and Yoshizaki, 2016). However, the effect of the unemployment rate on CPI is an important research topic. The existing literature reveals that fluctuations in the unemployment rate have a significant effect on the CPI. In particular, a decrease in the unemployment rate generally leads to a downward trend in the CPI, while an increase in the unemployment rate is not expected to increase the CPI in a directly proportional manner (Saqib et al., 2023). This clearly shows the impact and importance of employment levels on consumer prices. In the context of the housing market, the interaction between the unemployment rate and the CPI is also noteworthy. It has been determined that changes in unemployment rates can affect housing markets and, consequently, have indirect effects on household consumption patterns (Bhat et al., 2021). It has been observed that fluctuations in the labour market can negatively affect consumer behaviour and thus the CPI, an effect that points to broader economic fluctuations. In addition, the relationship between economic growth and unemployment and inflation has been analysed in depth. Research shows that economic growth may lead to a decrease in unemployment rates, which may have an impact on inflation rates (Soylu et al., 2018). These findings emphasise the complex relationships between macroeconomic variables and their aggregate effects on key economic indicators such as CPI.

Exchange rate changes, especially the US dollar, have a determining effect on the CPI. Volatility in the dollar exchange rate can have a multifaceted impact on the CPI. Existing literature has documented that exchange rate fluctuations have a direct impact on consumer prices by changing the costs of imported products. A study based on the case of Thailand revealed that various factors such as the exchange rate, fiscal and monetary policies, crude oil prices, and price levels in the US can have an impact on consumer prices (Hsing, 2020). Moreover, recent studies analysing the pass-through effect of the exchange rate on consumer prices have shown a decrease in the pass-through rates to domestic markets (Lin and Thompson, 2020). The relationship between the exchange rate and inflation has also been examined in different geographies. In an Egyptian study, a structural VAR model was used to determine the pass-through of the exchange rate to inflation, drawing attention to the effect of the exchange rate on import prices (Helmy et al., 2018). In the Turkish context, the relationship between exchange rate fluctuations, especially the US dollar, and the CPI and Producer Price Index (PPI) has been analysed in detail (Usupbeyli and Uçak, 2020). In short, the exchange rate, especially the dollar exchange rate, is a critical factor in the formation of consumer prices. Since exchange rate changes affect the costs of imported goods, they have a direct impact on CPI. Understanding the link between the dollar exchange rate and the CPI is vital for economists and policymakers to assess the course of inflation and make informed decisions about monetary and fiscal policies.

On the other hand, the Producer Price Index (PPI) has a critical impact on CPI. Changes in the PPI can affect the CPI through a number of mechanisms. Studies have indicated that fluctuations in the PPI may lead to adjustments in consumer prices in subsequent periods. In a European context, a study showed that if producers adjust their prices in response to changes in import prices, these changes can lead to significant changes in both producer and consumer prices (Schröder and Hüfner, 2002). This implies that there is a direct correlation between PPI changes and CPI changes. The pass-through effect of PPI on CPI has been analysed in different studies. These studies have documented that the PPI makes a significant contribution to fluctuations in the CPI and that these two indices are strongly interlinked (Gritli, 2021). Moreover, the pass-through of oil prices and its impact on domestic price inflation have been noted to have a significant impact on producer prices, especially in countries that are heavily dependent on oil imports (Sek, 2019). The long-term effects of the relationship between PPI and CPI have also been investigated in detail. These studies have revealed that the relationship between PPI and CPI is deeply significant and that PPI can significantly affect consumer prices in certain economic situations (Altunöz, 2022). These findings emphasise the importance of understanding how changes in producer prices are reflected in consumer prices and their impact on inflation. As a result, the impact of the Producer Price Index on the Consumer Price Index is of great importance. Changes in PPI can have both direct and indirect effects on consumer prices, underscoring the complex relationship between producer and consumer price dynamics in the economy.

Personal loan interest rates can have a significant impact on the CPI. Research has shown that fluctuations in retail loan interest rates can affect consumer prices in different ways. It has been shown that changes in interest rates can have an impact on the CPI by affecting consumers' spending behaviour and habits (Gharaibeh and Farooq, 2022). At the same time, the analysis of the relationship between interest rates and loan default rates has highlighted how changes in interest rates can have behavioural effects on loan repayments and thus affect the overall economic structure, including the CPI (Serrano-Cinca et al., 2015). The impact of interest rate fluctuations on consumer credit decisions has also been analysed. The findings revealed that consumers are sensitive to changes in interest rates and how these changes affect the credit decisions of individuals with different income levels (Azam et al., 2012). Moreover, investigating the impact of interest rates on loan asset quality has highlighted the importance of interest rate movements in shaping the performance of loan portfolios (Kosztowniak, 2022). In addition, studies on the relationship between interest rates and loan restructuring during periods of economic difficulty have shown that interest rates, together with factors such as loan maturity extensions and payment delays, play a critical role in the success of loan restructuring efforts. These efforts can have significant effects on CPI and overall economic stability (Firdauza and Rahadian, 2022). Thus, the impact of retail loan interest rates on the CPI suggests that changes in interest rates can have broad economic consequences through their direct and indirect effects on consumer behaviour, loan repayment patterns, credit decisions, and loan asset quality.

In this paper, 222 months of Consumer Price Index (CPI) data covering the period between January 2005 and June 2023 are analysed using deep learning and machine learning algorithms. The main purpose of the analysis is to determine which algorithm can most accurately predict the CPI for the specified period. Firstly, a literature review is conducted to determine the factors affecting the CPI. As a result of the literature review, it was found that inflation has a strong correlation with variables such as the exchange rate, interest rates, unemployment rate, oil prices, and producer price index. In the study, the consumer price index is taken as the dependent variable. As independent variables, producer price index, average Brent oil prices, unemployment rate, average dollar exchange rate (in Turkish Lira), and consumer loan interest rates were selected. These variables cover data collected over a period of 18 years and six months. In this phase of the research, we compare the effectiveness of various machine learning (Random Forest, Artificial Neural Networks, and K-Nearest Neighbors) and RNN (Long and Short Term Memory, Recurrent Unit with Gate) models in CPI forecasting. The paper starts with a summary of the related literature, includes the methodology, data set used, and model comparisons, and finally concludes with the findings and their economic interpretation.

1. Literature Review

This is a review of several studies that classify and predict data using the Random Forest (RF), Artificial Neural Networks (ANN), K-Nearest Neighbors (KNN), Gated Recurrent Unit (GRU), and Long and Short Term Memory (LSTM) models.

Rodriguez-Vargas (2020) assessed the machine learning techniques' predictive capacity for Costa Rican inflation forecasting. The predictions were built using LSTM networks, random forests, and severe GBoost, which are the two k-nearest neighbor variations. These projections are assessed using standards from the literature on optimum forecasting and contrasted with the effectiveness of the Central Bank of Costa Rica's existing univariate inflation predictions. The best-performing predictions were found to be LSTM, univariate KNN, and partly random forests. Furthermore, a mixture of projections produced better outcomes than the average of the univariate forecasts, outperforming each individual forecast they included.

Işığıçok et al. examined the capabilities of two distinct forecasting approaches for projecting future inflation using CPI data in their 2020 research. ANN and Box-Jenkins (ARIMA) models are used here to estimate 9-month inflation for the April–December 2019 period. The predictive abilities of the two methods are compared. For this, 207 data points from January 2002 to March 2019 are included in the CPI. The outcomes produced by the two methods are shown to be quite similar to one another.

Savitri et al.'s study from 2021 focused on inflation, particularly in developing nations like Indonesia. In this work, inflation rates were predicted and simulated using the LSTM forecasting approach. This method is an improvement on the Recurrent Neural Network (RNN) model that came before it. The outcomes showed that the best model contained five hidden layer nodes, used the Adam optimizer, and had a learning rate of 0.1. The forecasts indicated that Indonesia's inflation rate will only slightly increase by September 2022, still falling under the category of moderate inflation.

In order to anticipate inflation, Yang and Guo (2021) trained and examined variables from the consumer price index using the GRU model. Experimental findings using past data show that the GRU-RNN model can accurately predict China's inflation rate. This innovative strategy greatly outperformed certain well-known models in comparison, proving its efficacy.

Hatipoğlu et al. (2021) used k-nearest neighbor regression (kNNR) to estimate unemployment rates in Turkey. A fresh dataset of potential variables that could have an impact on the unemployment rate was produced specifically to this aim. The performance of the kNNR method was investigated using two different machine learning techniques: ridge regression and linear regression. kNNR approach (coefficient of determination (R^2) = 0.7498) was shown to outperform the comparative methods based on the data gathered from the experimental findings. According to the findings, the kNNR approach may be used in this industry.

Jamil (2022) created a hybrid model that incorporates ARIMA and LSTM methodologies for forecasting inflation, as opposed to separate models like ARIMA, LSTM, and PROPHET. The research evaluates the predictive ability of dependent and independent variables in predicting inflation in both established and developing nations using a range of measures, such as R^2, RMSE, MAE, and MAPE. To evaluate the consequences of consumer-level price inflation, monthly consumer price index data are gathered. When utilizing 90% of the available data for training and 10% for testing, the proposed hybrid model consistently outperforms existing models in predicting inflation in both developed and developing nations.

Basher and Sadorsky (2022) addressed a number of unanswered issues. They examined how several business cycle indicators, including market volatility, inflation, and interest rates, impact estimates of Bitcoin prices and if their relative importance changes over time. They also investigated the possibility that the macroeconomic factors that are crucial for forecasting Bitcoin values also have an impact on gold price forecasts. To answer these problems, they combined machine learning classifiers based on trees with traditional logit econometric models. The research produced important results. First, as compared to logit models, random forests scored better in terms of properly predicting the direction in which Bitcoin and gold prices will move. Bagging and random forests produced forecasting accuracies of between 75% and 80% for a five-day prediction, and over 85% for projections spanning 10 to 20 days. Second, the most important factors for predicting the future directions of the prices of gold and bitcoin appeared as technical indications, which point to some degree of market inefficiency. The research also shown that using Bitcoin as an alternative to gold may assist to diversify oil price volatility, which is essential for estimating the values of gold and bitcoin. This was one of the main takeaways from the study. Gold may be used as a method for diversification or as a hedge against inflation due to the fact that the value of gold has a higher influence on overall inflation than the value of bitcoin does.

Alkaff et al. (2022) proposed using RNN with gated GRU design to anticipate crime rates in Banjarmasin, accounting for the country's per capita income and inflation rate as factors. The research included statistics on inflation, the price of staple items in Banjarmasin City markets, and data from crimes reported to the Banjarmasin District Court. For evaluating the model, both RMSE and R-Square measurements were used. The study's findings, which included an RMSE value of 2.21 and a R^2 value of 0.84, demonstrated how effectively the GRU-RNN model worked.

In a research, Nakorji and Amuni (2022) underlined how crucial inflation forecasting is to attaining the global aim of price stability. The goal of this project is to anticipate Nigeria's inflation rate using machine learning techniques as an alternative to conventional approaches. Monthly data were acquired from the Central Bank Statistical Bulletin (2021). The research discovered that ridge regression and artificial neural networks had the greatest results in predicting inflation in Nigeria when compared to LASSO, elastic networks, and PLS. It also showed that the prime lending rate, maximum lending rate, food inflation, core inflation, and interbank rate are the main causes of inflation in Nigeria. In order to predict Nigeria's inflation rate, the research suggests using ridge regression and artificial neural network machine learning approaches.

In their research from 2023, Ganzagh et al. evaluated the ARDL, NARX, LSTM, and ARDL-D-LSTM models to see how well they could forecast Iran's monthly inflation rate over both the short and long terms. As part of the study, monthly inflation rates in Iran were calculated from 2005 to 2018, and the model's performance was evaluated using data from 2018 to 2020. Based on the results, for both short- and long-term forecasting horizons, the NARX model performs better than both the ARDL-D-LSTM hybrid model and the RMSE criterion.

In order to anticipate stock prices, Haryono et al. (2023) used the Transformer Encoder GRU (TEGRU) architecture. TEGRU includes a transformer encoder that transports time series data from multi-headed attention learning to the GRU layer for stock price computation. The prediction models were assessed using the accuracy mean absolute percentage error (AcMAPE), which is susceptible to the misclassification of changes in stock price. The results of the experiment demonstrated how sentiment indicators might affect stock issuers by enhancing the accuracy of the stock price forecast model. In five different feature scenarios, the TEGRU design fared better than competing transformer designs. It was also found that TEGRU had elements that decreased each stock issuer's financial risk.

With an emphasis on thorough forecast data gathering, Boaretto and Medeiros (2023) performed study to examine the efficacy of various forecasting methodologies in foretelling inflation. Using Brazil as a case study, they used conventional time series techniques, linear ML models, and non-linear ML models, taking into account various degrees of inflation granularity. The outcomes highlight the advantages of utilizing models in situations when there is a wealth of inflation data, especially in light of the COVID-19 epidemic. The use of the random forest model, which made use of both aggregate and specific inflation data, led to impressive medium- and long-term forecasting outcomes.

ARFIMA, GARCH, and GJR-GARCH models are only a few examples of the statistical modeling that has been the main focus of recent study on South African inflation, according to Kubheka (2023). He applies this idea to deep learning in this work and assesses model performance with measures including MSE, RMSE, RSMPE, MAE, and MAPE. He finds the model with the best degree of prediction accuracy using the Diebold-Mariano test. The study's conclusions indicate that cluster sampling-based LSTM models outperform the previously used ARFIMA-GARCH and ARFIMA-GJR-GARCH models.

2. Data and methodology

In this study, a total of 1332 data points, including the monthly average consumer price index (CPI), unemployment rate, dollar/Turkish lira exchange rate, producer price index (PPI), oil price, and consumer loan interest rates, covering 222 months from January 2005 to June 2023, are used. The data was acquired through the EVDS system of the Central Bank. Multiple variables, such as the unemployment rate, the USD to Turkish Lira exchange rate, the Producer Price Index (PPI), oil prices, and consumer loan interest rates, were employed to estimate the CPI. Table 1 displays a subset of the dataset that was used in this study.

Table 1. A Part of The Dataset Used In The Study

Date	**Average General CPI**	**Unemployment Rate**	**Averag e USD (TL)**	**Average PPI**	**Average Brent Oil Prices**	**Personal Loan Interest Rate**
2005-1	114.49	11.80	1.35	114.83	44.51	29.69
2005-2	114.51	11.90	1.31	114.81	45.48	26.22
2005-3	114.81	11.20	1.30	117.25	53.10	26.21
2005-4	115.63	10.40	1.35	119.62	51.88	25.13
2005-5	116.69	9.60	1.37	119.23	48.65	24.98
2005-6	116.81	9.60	1.35	119.64	54.35	24.90
2005-7	116.14	9.60	1.33	119.33	57.52	25.07
…..	….	….	….	….	….	….
…..	….	….	….	….	….	….
2022-12	1128.45	10.40	18.64	2021.19	81.00	30.23
2023-1	1203.48	10.30	18.76	2105.17	82.52	29.79
2023-2	1241.33	10.60	18.82	2138.04	82.59	29.88
2023-3	1269.75	10.30	18.97	2147.44	78.43	29.21
2023-4	1300.04	10.00	19.30	2164.94	84.74	33.04
2023-5	1300.6	8.80	19.68	2179.02	75.47	36.93
2023-6	1351.59	9.00	23.06	2320.72	74.84	41.46

Resource: evds2.tcmb.gov.tr

In addition, the correlation matrix showing the correlation of each variable in the dataset used in the study with other variables is shown in Table 2. When two variables have a positive correlation, there is a positive connection between them; when there is a negative correlation, there is a negative link. There is a perfect link between the variables when the correlation value is 1 or -1.

Table 2. The Correlation Matrix

	Average CPI	Unemployment Rate	Average USD (TL)	Average PPI	Average Brent Oil Prices	Personal Loan Interest Rate
Average CPI	1.000000	0.026788	0.991686	0.984358	0.038475	0.551966
Unemployment Rate	0.026788	1.000000	0.042192	-0.041148	-0.494708	0.032317
Average USD (TL)	0.991686	0.042192	1.000000	0.982769	0.022717	0.603090
Average PPI	0.984358	-0.041148	0.982769	1.000000	0.103561	0.591260
Average Brent Oil Prices	0.038475	-0.494708	0.022717	0.103561	1.000000	-0.067729
Personal Loan Interest Rate	0.551966	0.032317	0.603090	0.591260	-0.067729	1.000000

The findings of the correlation matrix show that there is a 0.026788 link between the "unemployment rate" and the CPI. The CPI and "unemployment rate" have a very weak positive association because of this value's lowness. There is a 0.991686 association between CPI and "USD". This demonstrates the very high positive correlation between "USD" and CPI. Conversely, there is a 0.984358 association between "PPI" and CPI. This number indicates that the CPI and "PPI" have a very strong positive association. The correlation between "Brent oil prices" and the CPI is 0.038475. This indicates that the CPI and "Brent oil prices" have a very slender positive association. Furthermore, there is a 0.551966 link between the "Personal Loan Interest Rate" and the CPI. This suggests that the "Personal Loan Interest Rate" and CPI have a somewhat favorable association. In conclusion, your data frame's correlation matrix displays the connections between its variables. A favorable association between two variables is shown by strong positive correlations, and a negative relationship is indicated by negative correlations. A link between two variables is extremely weak or nonexistent when the correlation is close to zero. An essential tool for comprehending the connection between variables is the correlation matrix.

In this study, Python software is used for consumer price index prediction. Accordingly, the first step is to import the necessary Python libraries. These libraries allow data processing, model building, and visualization of results. The libraries used are NumPy, Pandas, Matplotlib, Scikit-learn, and TensorFlow. The data set was loaded from an Excel file using Pandas. Also, "data.dropna()" was used to remove NaN values from the data set. Then, independent variables (X) and dependent variables (Y) are separated from the data frame. The data are normalized using min-max scaling. This converts each feature to a value between 0 and 1 and helps the model work better.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$

Following data normalization, training and test sets accounted for 80% and 20% of the total data set, respectively. This made it possible to assess the model's training performance using a different set of data. To divide the data set, a particular random seed (random_state) was used. The reason for using "random_state" is to ensure that the random splitting of the data set occurs in the same way each time. This ensures repeatability. That is, by using the same "random_state" value, the same data split is obtained each time. This helps to maintain comparability of results when evaluating the performance of the model or testing hyperparameter settings. It is vital to make sure that the findings are similar by utilizing the same data split since the purpose of this research is to compare the performance of various models. This will help to more fairly evaluate which model performs better. Therefore, the "random state" value was set to 42 and applied to all models.

The training loss of all models used in the study was monitored throughout the training period, and graphs showing the results were shown. In this way, it was possible to observe how all models learned and how the loss decreased.

Predictions were then made on the test data of all trained models. The predictions were returned to the original scales before Min-Max scaling. Five different methodologies were used in the investigation: LSTM, GRU, RF, ANN, and KNN models. The performance of the models was assessed using estimated statistical indicators, such as coefficient of determination (R^2), mean absolute percentage error (MAPE), mean square error (MSE), and mean absolute error (MAE). Equations 2, 3, 4, 5, and 6 may be utilized to calculate these statistical data.

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

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{3}$$

$$MAE = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \tag{4}$$

$$MAPE = \frac{\sum_{t=1}^{n}\frac{u_t}{Y_t}}{n} * 100 \tag{5}$$

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \mu)^2} \tag{6}$$

The findings were reported after the computation of the various error metrics developed before to quantify the discrepancy between the predicted and actual values. The projected and actual numbers were then shown on a graph. This made it possible to comprehend the models' performance better. Data preparation, model construction, training, prediction, and outcome assessment are all included in this process. The hyperparameters used in each of these stages are also detailed in the Results section. These hyperparameters include settings and optimizing algorithms that affect the performance of the models. In addition, the following sub-headings provide basic information about the models used in the study.

2.1. Long Short-Term Memory (LSTM)

It has been shown that the LSTM networks—which Hochreiter and Schmidhuber introduced in 1997—are more accurate than traditional neural network models. These networks, classified as recurrent neural networks, differ from conventional neural networks by incorporating a feedback loop between previous decisions and current actions. Due to the architectural design's ability to address the vanishing gradient problem in the update rule, they can handle long-term dependencies effectively (Rodrguez-Vargas, 2020). The structural diagram of the LSTM model is depicted below;

The LSTM method's forget gate is in charge of precisely throwing away the cell state from the preceding sequence. The time series' current input is represented by the symbol x_t, and its previous hidden state is represented by the symbol h_{t-1}.The activation function σ_g processes both of these values. This processing results in the output vector f_t, which is connected to the forget gate. This relationship is possible to expressed using Equation (7). In Equation 7; the bias coefficient represented as b_f, forget gates represented as W_f and U_f and activation function represent as σ_g.

$$f_t = \sigma_g \left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{7}$$

In Equation 8 and 9; the current point in the time series input x_t and the hidden state h_{t-1} from the prior time frame are what generate the values of the coefficients i_t and C'_t within this gate. The computation of these coefficients makes use of the activation function. The weight coefficients are denoted by symbols such as W_i, U_i, W_c, and U_c, meanwhile the activation function is denoted by the initials σ_g and σ_c.

$$i_t = \sigma_g \left(W_i x_t + U_i h_{t-1} + b_i\right) \tag{8}$$

$$C'_t = \sigma_c \left(W_c x_t + U_c h_{t-1} + b_c\right) \tag{9}$$

In Equation 10, the cell state, denoted as C_t, goes through an update process where it is derived by adding the product of the output of the input gate, i_t , and the cell candidate data, C'_t , and multiplying the output of the forget gate, f_t, by the previous cell state, C_{t-1}. The calculation gives a description of the modified cellular state, C_t.

$$C_t = f_t \, x \, C_{t-1} + i_t \, x \, C'_t \tag{10}$$

Equation (11) shows how the output vector o_t is created by applying the activation function σ_g to the input vectors h_{t-1} and x_t. The bias coefficient, b_0, and the weight coefficients of the cell state, W_o and U_o, are connected to the input gate.

$$o_t = \sigma_g \left(W_o x_t + U_o h_{t-1} + b_o\right) \tag{11}$$

$$h_t = o_t \text{ x } tan\, h \left(C_t\right) \tag{12}$$

After being created, the output value o_t is multiplied by the current sequential cell state C_t. Equation (12) shows how the output of the hidden layer is produced by the activation function tanh.

2.2. Gated Recurrent Unit (GRU)

The Recurrent Neural Network (RNN) variation known as the GRU was first developed by Cho et al. (2014).Recurrent neural networks (RNNs) struggle to efficiently capture and process long-range input; this problem is solved by the introduction of a gating component. The update gate (z_t) and reset gate (r_t) are the only gates included in GRU, in contrast to LSTM, which has a more complex structure. The update gate, also known as the input gate in the context of a Gated Recurrent Unit (GRU), is essential in deciding the degree to which the input (x_t) and the previous output (h_{t-1}) should be transferred to the following cell. On the other hand, the reset gate serves the purpose of determining the degree to which the prior information should be disregarded. The present memory content facilitates the transmission of only pertinent information to the subsequent iteration, as dictated by the weight W (Gao et al., 2021). The primary operations of the Gated Recurrent Unit (GRU) are determined by the Equation 13 and 14.

Update Gate:

$$z_t = \sigma(W_z * [h_{t-1}, x_t)] \quad (13)$$

Reset Gate:

$$r_t = \sigma(W_r * [h_{t-1}, x_t)] \quad (14)$$

The Gated Recurrent Unit (GRU) unit's candidate status value is indicated as $\tilde{h}_t$ computed as stated in Equation 15, and the final output status value is displayed as $\tilde{h}_t$calculated as given in Equation 16. These values are obtained after the gate has been reset and updated.

$$\tilde{h}_t = \tanh(W_{\tilde{h}_t} * ([r_t * h_{t-1}, x_t]) \quad (15)$$

$$h_t = (1\text{-}z_t)* h_{t-1} + z_t*\tilde{h}_t \quad (16)$$

2.3. Random Forest (RF)

In applications including both regression and classification problems, Breiman first unveiled the RF approach in 2001, showcasing its extraordinary efficacy and versatility. This approach, which combines many randomized decision trees and averages their predictions, has shown to be highly effective when there is a significant disparity between the number of variables and observations. Based on statistical learning theory, this method creates a large number of sample sets from the original training datasets by using Bootstrap randomized resampling. For each sample set, a decision tree model is constructed, and a voting mechanism is then employed to aggregate the decision tree outputs and predict classification outcomes (Xu et al., 2017). Figure 1 depicts the random forest algorithm's organizational structure.

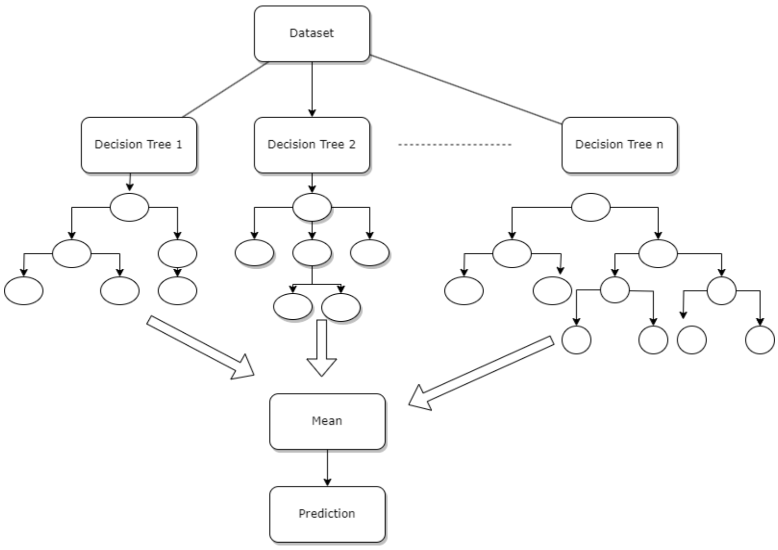

Figure 1: Random Forest Algorithm Structure

The Random Forest approach starts with the generation of training data, which is a process similar to that used in supervised learning. The remaining test data is retrieved using this model, which is built using the RF training data. As suggested by its name, the learning process entails a machine acquiring knowledge and abilities. This refers to the process of aiding the system's knowledge acquisition. The dataset that is given to the algorithm throughout this process and contains the variables required for the algorithm to build the model is referred to as the "training data." Data that are different from the "training data" and are not utilized to train the model are referred to as "test data." The algorithm's performance is evaluated using the test data. The RF algorithm model uses this data, which is different from the training data. The results generated by each distinct tree are then examined (Sevgen & Aliefendioglu, 2020).

2.4. Artificial Neural Networks (ANN)

ANNs are extensions of mathematical abstractions that drew inspiration from neural networks in biology. With the introduction of reduced computing units by McCulloch and Pitts in 1943, neural networks—also referred to as connectionist models or parallel distributed processing—saw a surge in early attention. The fundamental building blocks of computers are neural network nodes, sometimes referred to as synthetic neurons. A transfer function is used to describe the nonlinear behavior of neurons in a simplified mathematical model of a neuron, whereas synaptic effects are represented by connection weights that modify the impact of matching input signals. The weighted aggregate of the input signals modified by the transfer function indicates when the neuron will fire. By modifying the weights in accordance with the chosen learning method, a synthetic neuron may learn (Abraham, 2005).

Figure 2 shows the modeling of a multilayered neural network and a typical artificial neuron. With reference to Figure 2, the output signal flow (O) of a neuron and the signal flow from inputs $x_1, \ldots, x_n$ are both shown by arrows and are thought to be unidirectional. The following connection yields the neuron output signal O:

$$O = f(net) = f(\sum_{j=1}^{n} w_j x_j \tag{17}$$

If the function $f(net)$ is called an activation (transfer) function and w_j is the weight vector. A scalar product of the weight and input vectors defines the variable net.

$$net = w^T x = w_1 x_1 + \cdots + w_n x_n \tag{18}$$

When T is a matrix's transpose and, in the most basic scenario, the output value O is calculated as;

$$O = f(net) = \begin{cases} 1 & if\ w^T x \geq \theta \\ 0 & otherwise \end{cases} \tag{19}$$

where the node type is referred to as a linear threshold unit and θ is known as the threshold level.

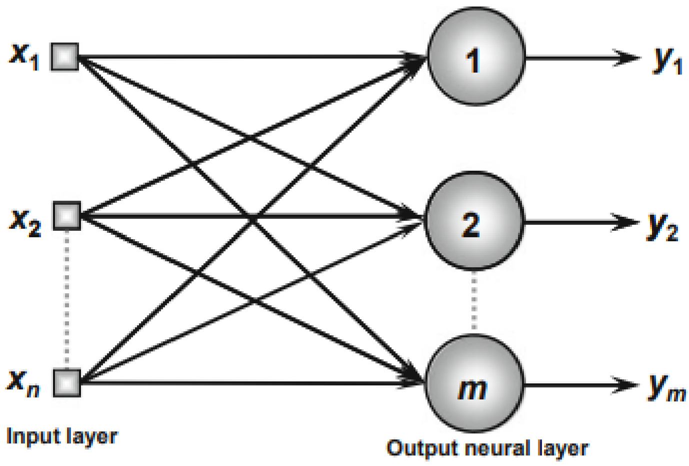

Figure 2: An illustration of a feedforward single-layer network

Resource: Da Silva et al. (2017)

Only one neural layer—which doubles as the output layer—and one input layer make up this neural networkFigure 2 depicts a basic layered feed-forward network including n inputs and m outputs. The flow of information in a neural network is characterized by a unidirectional movement, whereby data is sent from the input layer to the output layer. Figure 2 illustrates how the number of outputs in networks with this layout is always equal to the number of neurons. These networks are often used to linear filtering and pattern classification issues. As will be covered in the next sections, two of the primary network types that make up the feed-forward architecture are Perceptron and ADALINE. Both of these networks use learning algorithms throughout their training processes. The learning algorithm employed by the first network is based on the Hebb rule, while the second network utilizes the Delta rule.

2.5. K-Nearest neighbors (kNN)

The training datasets are represented by tree structures in the structure-based k-NN approach. Berchtold proposed a Voronoi cell-based method specifically for nearest neighbor searches. Branch-and-bound strategies serve as the foundation for index structures used in range queries. A seminal method, created by Roussopoulos, finds the k nearest neighbors in which an R-tree indexes the points using depth-first tree traversal. Throughout the traversal, a range of methods are used to arrange and trim elements in the tree's nodes (Dhanabal & Chandramathi, 2011).

Cheung and Fu (1998) simplified this approach without compromising its efficacy. Many modifications are made to branch-and-bound algorithms to better match nearest neighbor problems, especially when high-dimensional data is involved. There are several incremental approaches for similarity ranking that can rapidly find the (k +1)-st nearest neighbor once the k closest neighbors are found. For the objects that need to be visited, they take use of the global priority queue of an R-tree. Hjaltason and Samet (1999) specifically introduced an incremental nearest neighbor method that makes use of a priority queue of the R+-tree elements that need to be visited. They show that such a best-first traversal is optimal for a given R-tree. Henrich (1994) introduced a method that is somewhat similar and utilizes two priority queues. For high-dimensional data, multi-step nearest neighbor query processing methods are often used.

Nearest neighbor search is one of the core problems in computational geometry, non-parametric statistics, and machine learning. An exhaustive search requires quadratic effort for every N points; however, several fast methods may reduce the complexity in both exact and approximation searches. The common kernel (kNN kernel) in all these techniques accurately solves a lot of small-size issues via exhaustive search (Yu et al., 2015).

Applications ranging from cross-validation studies in supervised learning to nearest-neighbor graph creation for manifold learning and hierarchical clustering are just a few of the broad range of uses for it. Since the all-nearest-neighbor issue needs $O(N^2)$ distance evaluations, it becomes prohibitively costly for extremely big N. Regular spatial decompositions such as quadtrees, octrees, or KD-trees may solve the kNN issue in low dimensions ($say\ d\ < 10$) by employing $O(N)$ distance assessments. However, it is known that treebased algorithms ultimately have quadratic complexity in larger dimensions. We have to give up on accurate searches and settle for approximate searches in order to get around this issue. The most advanced techniques for solving the kNN issue in large dimensions use randomization techniques like hashing or tree-based techniques. These approximation search strategies all boil down to the same thing. X of size N is divided into N/m groups of size $m \times n$ (containing m query points and n reference points—both from X) and N/m exact search kNN problems are solved for the m query points in each group, where $n = O(m)$ is dependent on the specifics of the approximation search strategy. The kNN kernel exactly looks for the k nearest points among n reference points when m query points are selected from X. The kNN kernel is used by both the approximation search algorithms and the lower-dimensional exact search methods.

2.6. Shapley Additive Explanations (SHAP) Method

The SHAP (Shapley Additive Explanations) method is a technique used in machine learning to explain the predictions of complex models. It provides a way to attribute the prediction result to different features in the input data. SHAP values represent the contribution of each feature value to the prediction of a particular instance (Khan, 2024). This method is particularly useful for understanding the importance of different features in predictions, regardless of the complexity of the underlying machine learning model (Rodríguez-Pérez and Bajorath, 2020). In the context of multivariate time series forecasting, SHAP can be applied to provide insight into how each variable in the time series contributes to the overall forecast. By using SHAP in multivariate time series forecasting, researchers can better understand the impact of each variable on the predicted outcomes (Li et al., 2021). This allows for both local and global interpretation of the model, enabling analysts to not only explain individual predictions but also understand the overall behaviour of the model with respect to all input variables (Lee et al., 2022).

SHAP is an additive method that can be applied after model building and is a versatile tool for interpreting the results of various machine learning models (Kitani and Iwata, 2023). By helping to identify and prioritise features that significantly influence the predictions made by the model, it can help to identify the key factors driving the predicted outcomes (Rodríguez-Pérez and Bajorath, 2020). Furthermore, SHAP values can be used to represent the relative importance of different factors, providing a clear understanding of the impact of each variable on the predicted outcomes. In summary, the SHAP method is a valuable tool in machine learning interpretability and allows complex model predictions to be explained by attributing them to specific input characteristics. In the context of multivariate time series forecasting, SHAP can be used to understand the contribution of each variable to the predicted results, providing both local and global insights into model behaviour. In this study, the SHAP method is applied to interpret the results of the GRU model, which shows the best CPI forecast performance.

Findings of the Study

In this study, LSTM, GRU, RF, ANN, and kNN models are used to predict the Consumer Price Index (CPI). Many independent variables, such as unemployment rate, dollar/Turkish lira exchange rate, producer price index (PPI), oil price, and consumer loan interest rates, are used in the models.

The MinMaxScaler approach was used in the research to standardize the dependent variable and independent variables on the dataset using Python software. The data set was then split into test (20%) and training (80%) groups. Then, each model was guaranteed to utilize the same data split by using the "random state" command. This made it possible for the outcomes to be comparable and reproducible. In this sense, the models' predictions on the test data set also perform comparably. 42 was chosen as the random state rate.

The LSTM model was developed first, after the data set was prepared to assess the models' prediction capabilities. The LSTM model contains 50 neurons and is composed of a single layer. There is just one unit in the output layer. The model used the Adam optimization process and was built using mean squared error (MSE). On the other hand, the activation function used was "ReLU". A dropout rate of 0.2 was applied, and the learning rate was set at 0.001. A hyperparameter called learning rate is used when deep learning and machine learning models are being trained. The amount of weight update that occurs during a model's training is determined by its learning rate. More precisely, the learning rate—which is established at every learning step—determines how much the weights are altered. Dropout makes it possible to train the model more effectively. Because of the improved generalization and less overfitting, it may need less data in this approach. The purpose of these settings was to avoid overfitting.

Following its creation, the LSTM model underwent 100 epochs of training on the training set. Every epoch's training loss was noted. The training loss at the conclusion of each epoch was monitored using the variable "history". On a graph, the training loss values that were noted throughout the training process were shown. Figure 3 displays the graph of the LSTM model's training loss values.

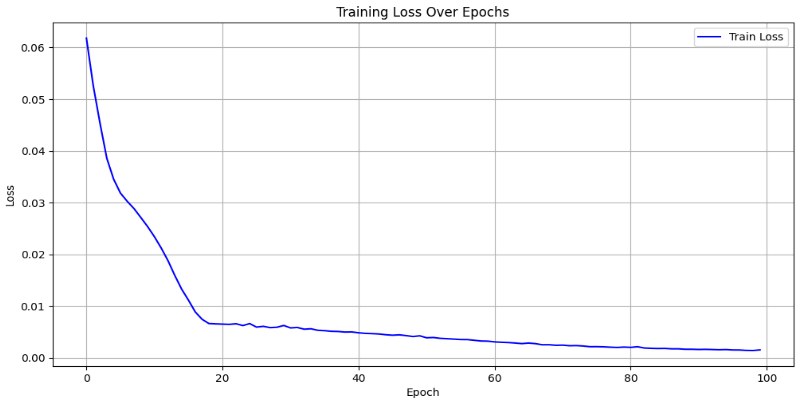

Figure 3. Loss Values of The LSTM Model During Training

Upon analysis of Figure 3, the loss value from the first epoch onwards is 0.0618. The training loss began to decrease with each new epoch. This demonstrates that the model has a superior understanding of the data and that its predictions are more accurate than the actual values. The loss value of 0.0016 in the final epoch (100/100) shows that the model is well-suited to the training set. These outcomes are used to track the model's training performance. A reduced training loss suggests that the model provides a better fit to the training set. This does not imply, however, that the model operates effectively on data from the actual world. As such, an independent assessment of its performance using test data is necessary. These findings demonstrate that the model is effectively trained and progressively lowers its loss on training data. However, in order to assess the model's capacity for generalization, test data performance should also be taken into account.

After evaluating the training performance of the LSTM model using the trained model, predictions were made based on the training and test data. The scales for these predictions were then set to their initial values. Table 3 shows the results of calculations using RMSE, MSE, MAE, MAPE, and R^2 to determine the errors between predicted and actual values on the training and test data.

Table 3. Error Calculations of LSTM Model Predictions on Training and Test Data

LSTM Model	Training	Test
RMSE	29.3146	26.6693

MSE	859.3471	711.2542
MAE	19.1467	18.8214
MAPE	0.0677	0.0688
$\boldsymbol{R^2}$	0.9872	0.9882

With regard to predicting the consumer price index, these findings demonstrate how well the LSTM model performs. Since R^2 is high and error measurements like RMSE and MAE are low, the model's predictions are not too far off from the real data. On the test data, a graph displays the actual and anticipated CPI values of the model. Figure 4 represents this graph.

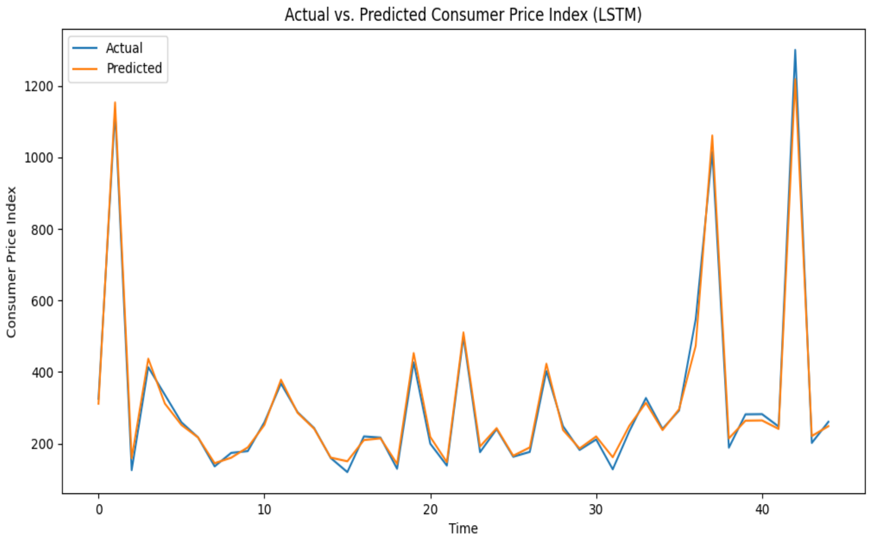

Figure 4. The Actual and Predicted CPI Values of The LSTM model on The Test Data

However, after the prediction process with the LSTM model, the prediction process was performed with the GRU model, which is the second deep learning method. While creating the GRU model, the Python programming language and "tensorflow" were used. The model consists of a single layer of 50 units and is defined by the ReLU activation function. Next, a thick coating was put on top. This was used to provide the model's results. The Adam optimization technique was used to train the GRU model on the training set and update the weight values. The model was trained for 100 epochs with a batch size of 32 during the training phase. Furthermore, the model's training loss processes were tracked and its performance was assessed using test and training datasets. A 0.2 dropout rate and a 0.001 learning rate were used. The same hyperparameters, together with the identical settings for the epoch and batch size, were utilized to compare the performance of the LSTM and GRU models. This would allow for a comparison of the models' performances.

Following its creation, the GRU model underwent 100 epochs of training on the training set. Every epoch's training loss was noted. The "History" variable was used to monitor the training loss at the conclusion of each epoch. On a graph, the training loss values that were noted throughout the training process were shown. Figure 5 displays the training loss values of the GRU model on a graph.

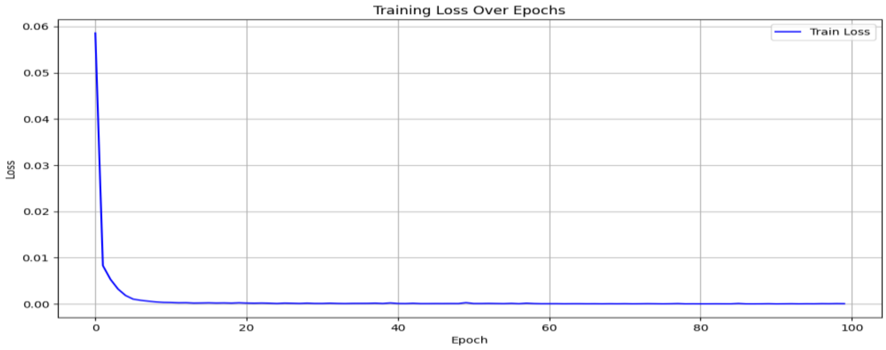

Figure 5. Loss Values of The GRU Model During Training

The loss value beginning in the first period is 0.0586 when Figure 5 is examined. The training loss began to decrease with each new epoch. This demonstrates that the model has a superior understanding of the data and that its predictions are more accurate than the actual values. In the most recent era (100/100), the loss value was 0.0001. This demonstrates that, using the training data, the model generates very accurate predictions. Furthermore, the developers see that at the conclusion of every epoch, the training loss consistently declines, suggesting that the model has a high capacity for learning. But by the time the training phase is over, the model has adjusted to the data and achieved a very low error rate, suggesting that it might make accurate predictions in the future.

After evaluating the training performance of the GRU model, predictions were generated using the trained model based on training and test data. Then, the scales for these predictions were set to their initial values. Table 4 shows the results of calculations using RMSE, MSE, MAE, MAPE, and R^2 to determine the errors between predicted and actual values on the training and test data.

Table 4. Error Calculations of GRU Model Predictions on Test Data

GRU Model	Training	Test
RMSE	20.5306	13.1784
MSE	221.5074	173.6715
MAE	11.7539	6.1201
MAPE	0.0404	0.0157
$\boldsymbol{R^2}$	0.9937	0.9971

Upon examination of Table 4, the RMSE value of 13.1784 suggests that, on average, the model's predictions on the test data differ by 13.18 units from the actual values. This suggests that overall, the model operates well. With an average squared error of 173.67 units on the test data, the model has an MSE value of 173.6715. Since the MSE increases the size of the mistakes, it offers information that is comparable to that of the RMSE. On the test data, however, the model's average absolute error is 6.12 units, according to the MAE value of 6.1201. The model's predictions are often within a reasonable range of their real values when the MAE value is low. The model's predictions often differ by 1.57% from the real values, as shown by the MAPE value of 0.0157. The model's predictions often have a small percentage error margin when the MAPE value is low. A statistical indicator of how well the model fits and explains the data is the R^2 value. With a score of 0.9971, the model accounts for 99.71% of the variation in the data, indicating a good fit. A high R^2 score suggests that the model provides a good explanation for the data and that its predictions may be trusted. These findings demonstrate, in summary, that the GRU model operates rather effectively on the test data. High R^2 values and low RMSE, MSE, MAE, and MAPE values show that the model has been properly trained and that the predictions are quite close to the actual data. A graph displays the model's actual and anticipated CPI values for the test data. In Figure 6, this graph is shown.

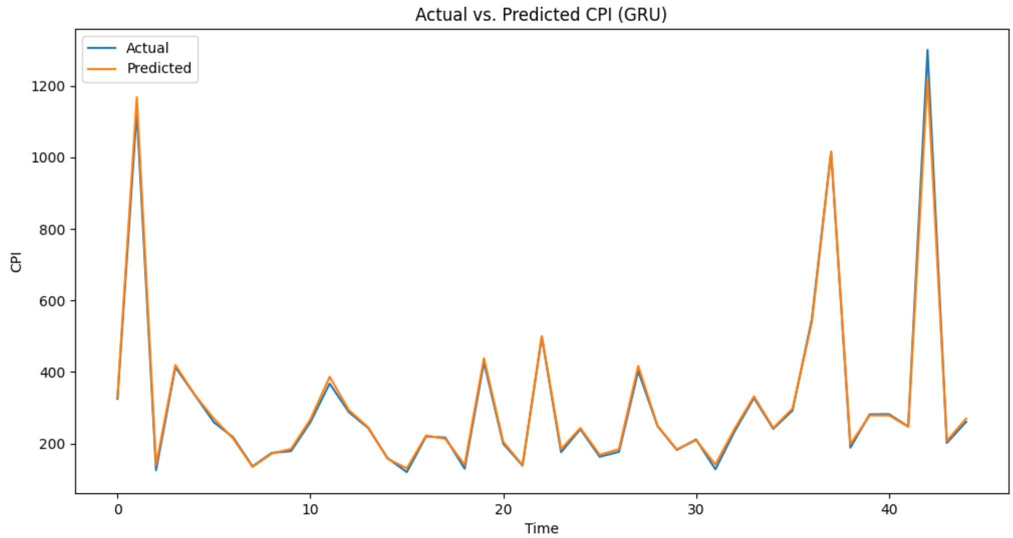

Figure 6. The Actual and Predicted CPI Values of The GRU model on The Test Data

Using the same training and test data, machine learning techniques were assessed for performance after the prediction procedures were carried out using LSTM and GRU deep learning models. ANN, RF, and KNN models were included into the research in this field.

However, unlike the LSTM and GRU deep learning models, this model used a single layer of 50 units, specified using the ReLU activation function, prior to constructing the ANN model for the prediction process. Next, a thick coating is applied. The model was output using this. The Adam optimization approach was used to train the ANN model on the training set of data in order to update the weight values. The chunk size was set at 32 and the model was trained for 100 epochs throughout the training procedure. Furthermore, the model's training loss processes were tracked and its performance was assessed using test and training datasets. The epoch value was set to 100, the dropout rate was set to 0.2, and the learning rate was set to 0.001. To facilitate comparison, the hyperparameter values, methods, and functions used in this model are the same as those found in deep learning models. This makes it feasible to compare how well machine learning and deep learning models perform using training and test data.

The ANN model was trained on the training set for 100 epochs after it was created. The training loss for each period was noted. At the end of each epoch, the training loss was tracked using the "History" variable. The training loss values that were observed during the training procedure were shown on a graph. The training loss values graph for the ANN model is shown in Figure 7.

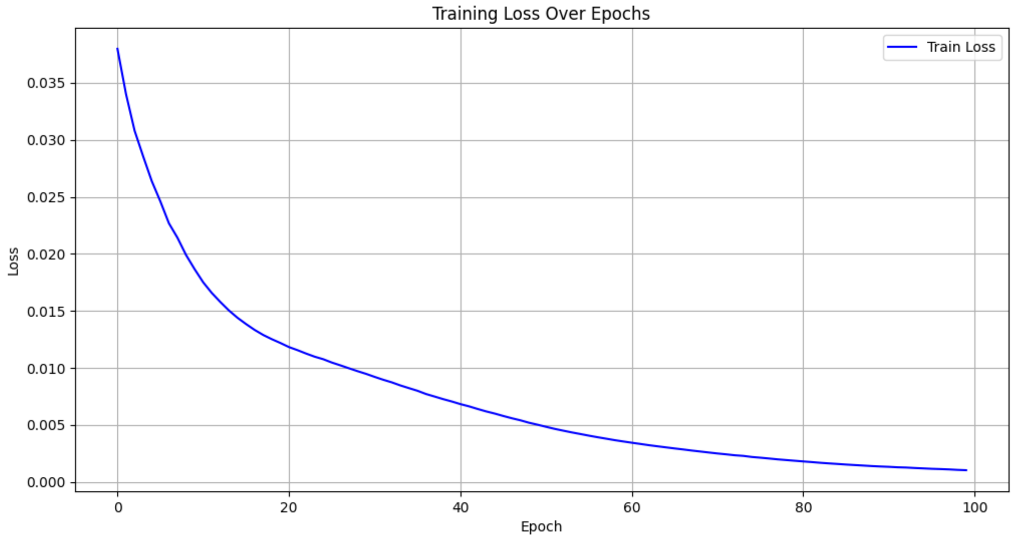

Figure 7. Loss Values of The ANN Model During Training

The ANN model's training procedure and the loss values that were attained after training are shown in Figure 7. There were 100 epochs in the training procedure. Every epoch symbolizes the process of refining the model's prediction power and adjusting it to the data. The image illustrates how the model has a large initial error rate during training, but that error rate quickly drops as training progresses. A reduced training error suggests that the model begins to have a deeper understanding of the data and improves its prediction accuracy.

After the training performance of the ANN model was evaluated, the trained model was used to make predictions on the training and test data. Then, the scales for these predictions were set to their initial values. Table 5 shows the results of calculations using RMSE, MSE, MAE, MAPE and R^2 to determine the errors between predicted and actual values on the training and test data.

Table 5. Error Calculations of ANN Model Predictions on Traingng and Test Data

ANN Model	**Training**	**Test**
RMSE	46.7935	37.8692
MSE	2189.6387	1434.0799
MAE	39.4928	33.2364
MAPE	0.1900	0.1441
$\boldsymbol{R^2}$	0.9674	0.9762

From the analysis of the prediction performance of the ANN model on the training and test data in Table 5, it is clearly seen that the model works quite well. The MAPE number shows a very small error percentage, while the RMSE, MSE, and MAE values are also quite low. Also, the high R^2 value indicates that the model does a good job of explaining the data. The model performs well on the test data set. The graph in Figure 8 shows the actual and expected CPI values of the ANN model based on the test data.

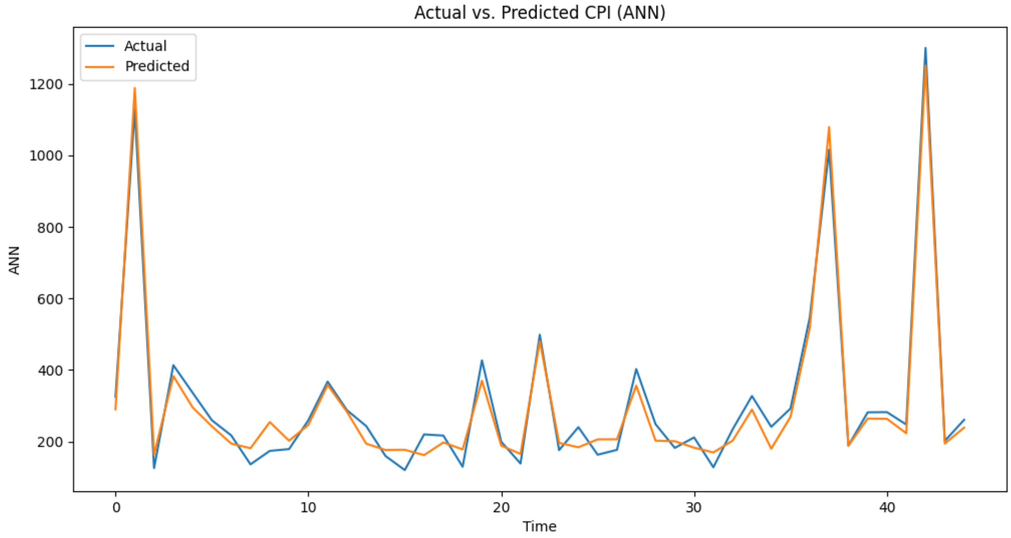

Figure 8. The Actual and Predicted CPI Values of The ANN model on The Test Data

However, the Python program "RandomForestRegressor" was used to generate the Random Forest (RF) regression model. Several different decision trees were used to generate this ensemble model. Because they were trained on a randomly chosen portion of the data, these trees vary from one another. By doing this, the model's variance is decreased and its prediction accuracy is improved.

Multiple decision trees are combined into an ensemble algorithm, which is what the Random Forest model really is. Using decision trees as the foundational algorithm. These trees are tree structures that carry out regression analysis or categorize data based on the values of independent variables. After each tree has made a forecast by deciding what to do based on the data, the trees are joined. Since each tree is trained using a randomly chosen subset of data, each one is unique thanks to the Random Forest method. In conclusion, Random Forest essentially mixes many trees with the ensemble approach and the Decision Tree algorithm. As a result, the model is able to provide more accurate and consistent forecasts. The hyperparameter "n_estimators" is used in this model. The number of decision trees the model produces is determined by this hyperparameter. In general, a model becomes more stable and resilient with more trees. On the other hand, an excessive number of trees may lengthen the model's training period. This parameter's value in this model is 100. The epoch value was set to 100 in other models. The preferred value in this model was the same as in the other prediction models. Additionally, this model, like the other prediction models, makes use of the "random_state" hyperparameter. This is a randomness-related hyperparameter. When using the same dataset and hyperparameters, it is utilized to get consistent results. Repeatability is ensured by using a specified number. Additionally, "max_depth=10" is the maximum depth of the trees, and "min_samples_leaf=2" is the lowest number of leaf samples required. The maximum depth of any decision tree is determined by the variable "max_depth". The intricacy of the data determines the depth of the trees. In general, deeper trees are able to capture more intricate interactions. Conversely, "min_samples_leaf" establishes the bare minimum of leaf samples. This limits the maximum size of a leaf. Reduced values raise the possibility of overfitting but may result in more complicated models.

Following its creation, the Random Forest model was trained on a training set consisting of 100 trees. Table 6 displays the training outcomes' performance.

Table 6. Error Calculations of RF Model Predictions on Training Data

	RF
RMSE	31.4479
MSE	988.9718
MAE	6.3359
MAPE	0.0377
$\boldsymbol{R^2}$	0.9955

The RF model works rather well on the training data, according to these findings on the training data analysis of the information in Table 6. To put it another way, the model has a low error rate when predicting the training set. These findings do not, however, show how successfully the model is applied to actual data. Thus, the way it performs on test data has more significance. The actual and predicted CPI values of the RF model on the training set of data are shown on the graph in Figure 9.

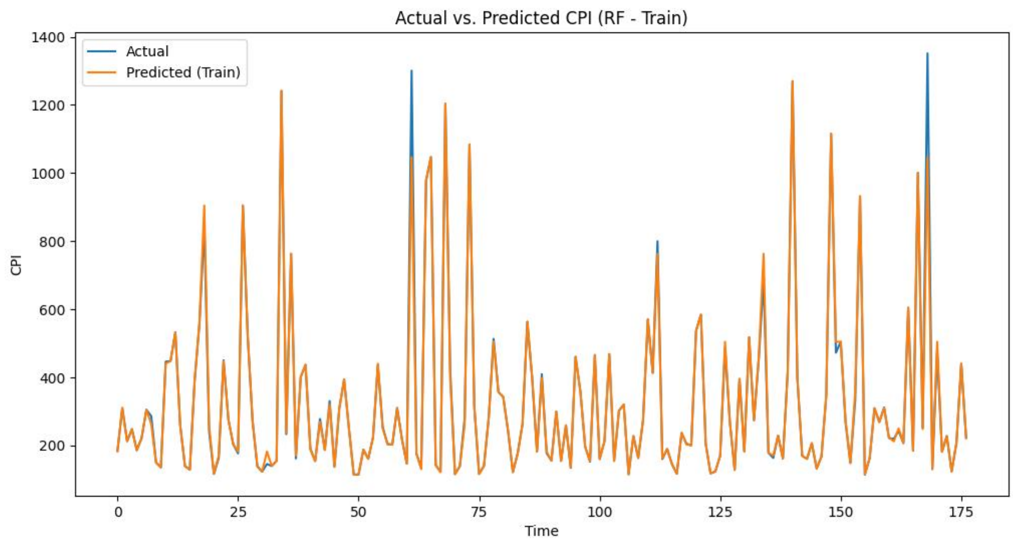

Figure 9. The Actual and Predicted CPI Values of The RF model on The Training Data

Following the RF model's training, predictions were produced using the model's training on test data. Then, the scales for these forecasts were adjusted to their initial values. Table 7 displays the results of the calculations made using RMSE, MSE, MAE, MAPE, and R^2 to determine the errors between the anticipated and actual values on the test data.

Table 7. Error Calculations of RF Model Predictions on Test Data

	RF
RMSE	42.7597
MSE	1828.3920
MAE	17.1808
MAPE	0.0532
R^2	0.9696

The Random Forest (RF) model's performance was assessed using the error calculations shown in Table 6 for the test set of data. The degree to which the forecasts differ from the actual values is shown by the RMSE. Better model performance is shown by a lower RMSE. In this instance, it was discovered that the RF model's RMSE was 42.7597. This figure shows that there is an average 42.76 unit inaccuracy in the CPI estimates using the test data. Consequently, this score suggests that the model's test performance is subpar or unacceptable. The mean squared of the predictions' deviations from the actual values is provided by the Mean Squared Error (MSE). Better model performance is indicated by a lower MSE. The RF model's mean square error is 1828.3920. This shows that the test data's mean squared error for the model is 1828.39 units. The mean absolute deviation (MAE) of the predicted values from their actual values is shown. A low MAE denotes a well-performing model. 17.1808 is the RF model's MAE. This shows that the model's average absolute error on the test set of data is 17.18 units. The MAPE function determines the mean percentage difference between the predicted and actual values. A model with a lower MAPE performs better. The RF model's MAPE was found to be 0.0532. This shows that the model's average absolute percentage error on the test set of data is 5.32%. R^2 value represents the model's fit. The model fits better when the values are closer to 1, which ranges from 0 to 1. The RF model's R^2 value is 0.9696. This suggests that the majority of the independent variables can be well explained by the model in relation to the dependent variable. In conclusion, it can be concluded that the RF model performs rather well in tests based on the provided error computations. A graph displays the RF model's actual and anticipated CPI values for the test data. In Figure 10, this graph is shown.

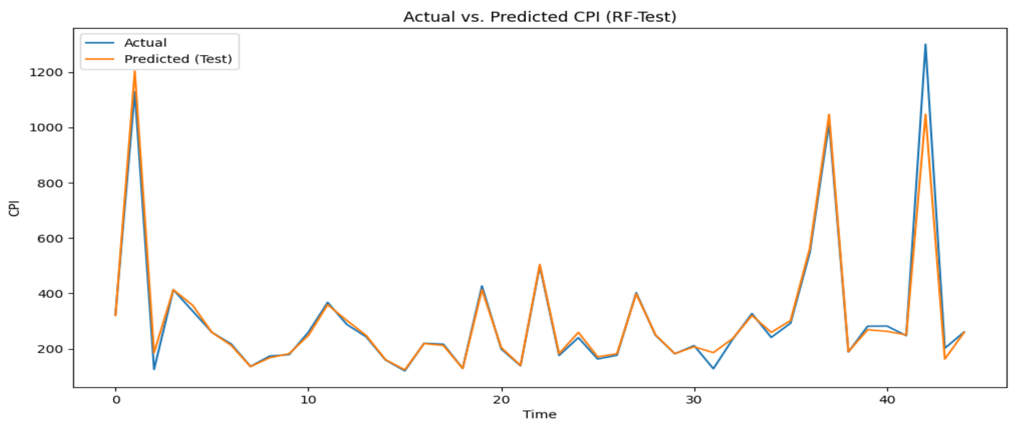

Figure 10. The Actual and Predicted CPI Values of The RF model on The Test Data

Using the Python program "KNeighborsRegressor," the kNN regression model—the final machine learning model used in this work for CPI prediction—was developed. The k-NN technique is used by this model to determine the link between dependent and independent variables. The number of neighbors to be employed in the k-NN method is determined by the parameter "n_neighbors" (Number of Neighbors). N_neighbors in this model equals 5. This implies that each forecast will make use of the five closest neighbors. On the other hand, the weighting of the neighbors in the forecast is determined by the "weights" parameter. Uniform and distance are two popular options. While distance weights are inversely proportional to neighbor distance, uniform assigns the same weight to each neighbor. Weights in the model are "uniform," meaning that every neighbor has the same weight. When determining the distance between neighbors, the distance metric to be used is specified via the option "metric (distance metric)". One may employ a variety of metrics, including Minkowski, Manhatten, and Euclidean distances. Since the metric in this paradigm is "euclidean," Euclidean distance is used.

Following the creation of the KNN model, the training set (X_train and y_train) was used to train the k-NN model. Table 8 displays the performance of the training outcomes.

Table 8. Error Calculations of kNN Model Predictions on Training Data

	kNN
RMSE	128.4507
MSE	16499.5922
MAE	54.8671
MAPE	0.1061
$\boldsymbol{R^2}$	0.7548

The k-NN model's performance on the training set of data is assessed by the findings in Table 8. Models with low error values, such RMSE and MAE, are well-trained. R^2 and MAPE scores also show how predictive the model is. Reducing these error numbers could be preferable for the model to function better. The actual and predicted CPI values of the kNN model on the training set are shown in the graph in Figure 11.

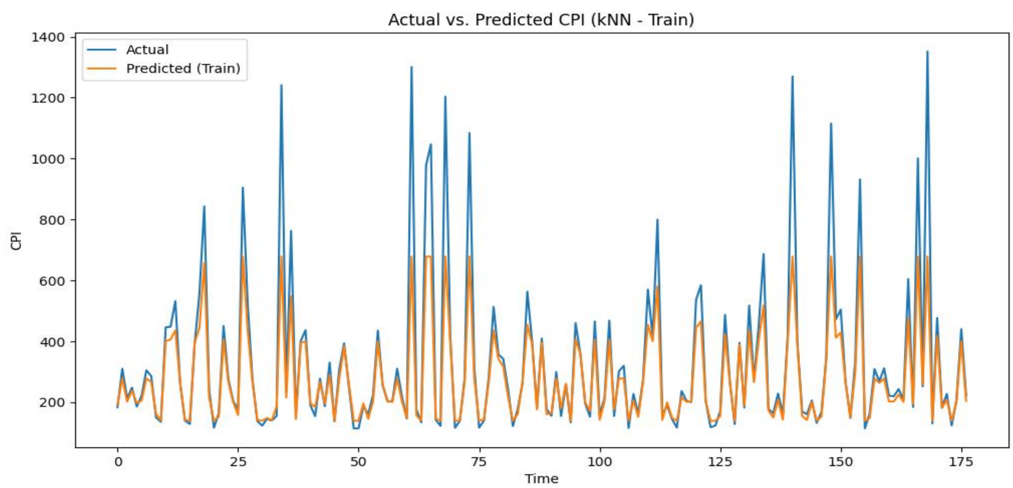

Figure 11. The Actual and Predicted CPI Values of The kNN model on The Training Data

Predictions were produced using the trained model on the test data after the training of the kNN model. Then, the scales for these forecasts were adjusted to their initial values. Table 9 displays the results of the calculations made using RMSE, MSE, MAE, MAPE, and R^2 to determine the errors between the anticipated and actual values on the test data.

Table 9. Error Calculations of kNN Model Predictions on Test Data

	kNN
RMSE	127.2801
MSE	16200.2484
MAE	47.5299
MAPE	0.0884
$\boldsymbol{R^2}$	0.7314

The error calculations of the k-Nearest Neighbors (kNN) model's predictions on the test data are shown in Table 9. The value of the RMSE is reported as 127.2801. This figure shows that on the test data, the kNN model produces an average error of 127.2801 units. There is a 16200.2484 MSE value. This number shows that there is an average error of 16200.2484 units squared between the actual values and the predictions of the kNN model. There is a 47.5299 MAE value. This is the mean of the absolute deviations between the kNN model's predicted values and actual values. The value of MAPE is reported as 0.0884. This figure shows that, on average, there is a 0.0884% percentage error rate between the actual values and the predictions made by the kNN model. The value of R^2 is shown as 0.7314. The model's ability to explain the variation in the test data is shown by the R^2 value. The model performs better the closer the value gets to 1. With a coefficient of 0.7314, the test data's variance is around 73.14% explained by the model. Given that the model accounts for 73.14 percent of the variation in the test data, the R^2 value suggests that the model has adequate explanatory power. In contrast to other error measures, it still contains some error, hence it may need to be improved. A graph displays the actual and anticipated CPI values of the kNN model based on the test data. In Figure 12, this graph is shown.

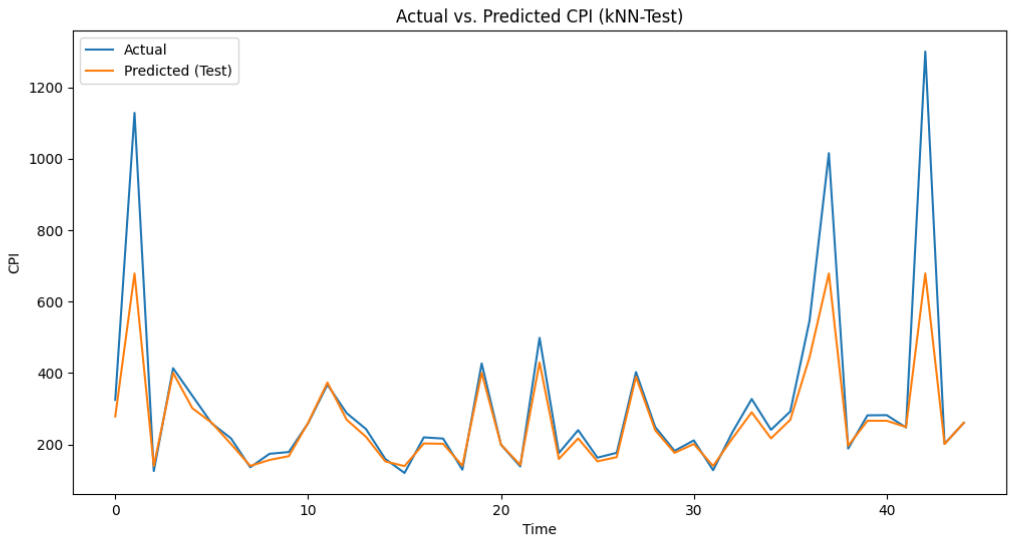

Figure 12. The Actual and Predicted CPI Values of The kNN model on The Test Data

In summary, Table 10 presents a summary of the test findings for all forecasting models utilized in this study.

Table 10. Error Calculations of All Models Predictions on Test Data

	LSTM	GRU	RF	ANN	KNN
RMSE	26.6693	13.1784	42.7597	37.8692	127.2801
MSE	711.2542	173.6715	1828.3920	1434.0799	16200.2484
MAE	18.8214	6.1201	17.1808	33.2364	47.5299
MAPE	0.0688	0.0157	0.0532	0.1441	0.0884
$\boldsymbol{R^2}$	0.9882	0.9971	0.9696	0.9762	0.7314

The error estimates of the predictions produced by five different models on the test data are shown in Table 10's findings. From various angles, each error measure aids in assessing a model's performance. With respect to this data, the GRU model (13.1784) has the lowest RMSE value. The model with the greatest error and RMSE value is kNN (127.2801). This indicates that the GRU model's predictions on the test data are, on average, more accurate than those of the other models.

Nonetheless, GRU (173.6715) is the model with the lowest MSE value. kNN (16200.2484) is the model with the greatest error and MSE value. Once again, it is evident that the GRU model's predictions have lower mean square error than those of the other models.

Conversely, GRU (6.1201) is the model with the lowest MAE value. kNN (47.5299) is the model with the greatest absolute error value and MAE value. The GRU model demonstrates that, in absolute terms, the predictions are more in line with the actual results.

Furthermore, GRU (0.0157) is the model with the lowest MAPE value. The MAPE value (0.0884) of the kNN model is the highest. It is evident that the GRU model's predictions have a less % error rate.

Lastly, GRU (0.9971) has the highest R^2 value of all the models, meaning it can account for almost 99.71% of the variation in the test data. The kNN model has the lowest R^2 value (0.7314).

These findings demonstrate that when all error indicators are taken into account, the GRU model performs the best on the test data. The predictions are more likely to match the real values when the GRU model with the lowest RMSE, MSE, MAE, and MAPE values is used. The model that most effectively describes the variation in the data is the GRU model with the greatest R^2 value. Not to mention, the LSTM model works wonderfully. In comparison to other models, the values of RMSE and MAPE are very low. In general, ANN (artificial neural network) and RF (random forest) models also perform well, although not as well as LSTM and GRU models. Among all the error measures, the kNN model performs the worst. The values of RMSE and MAE in particular are rather high.

Consequently, it seems that the GRU model performs the best on the test data. As a result, the deep learning model GRU may be regarded as this study's most successful model. While deep learning models (GRU and LSTM) outperform machine learning models (RF and ANN), the former two also perform well. It seems that KNN is a less appropriate model for this specific dataset.

In order to interpret the results of the GRU model, which performed best in both the training and testing phases, the Shapley additive annotation (SHAP) method was applied. The SHAP method shows that each feature is assigned a relevance value by SHAP for a given prediction. In particular, a positive SHAP value means that the input has a positive impact on the output, while a negative SHAP value means that the input has a negative impact on the output (Nguyen et al., 2023). As shown in Figure 13, the SHAP values for the GRU model are obtained by ranking the features according to the sum of the magnitudes of the SHAP values during both the training and testing phases when applying the SHAP values to indicate the distribution of the effects of each feature on the output of the technique. The color represents the value of the feature (high red or low blue).

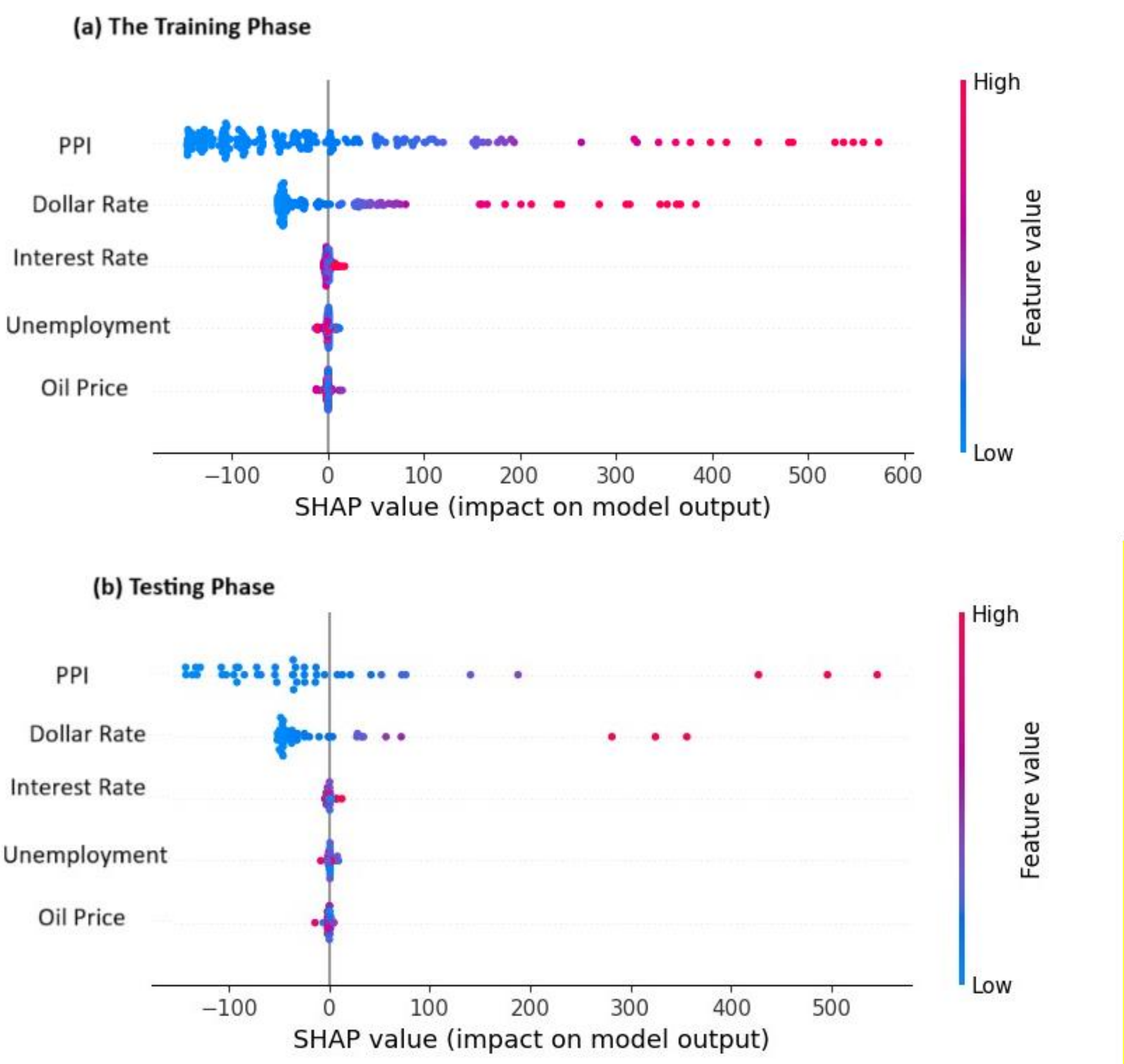

Figure 13. Feature Importance Based on the GRU Model for (a) the Training Phase and (b) the Testing Phase.

When the SHAP values of the training phase of the GRU forecasting model are analyzed, the blue and red dots of the PPI values have a wide distribution, indicating that these values have different effects from low to high. The wide range of SHAP values indicates that the PPI has a significant impact on the GRU model output. On the other hand, the blue dots, which indicate that the dollar exchange rate generally positively affects the GRU model output, have a more concentrated distribution at low values. However, the presence of red dots indicates that high values of the dollar exchange rate can also have significant effects. In the case of interest rates, SHAP values have less variance and generally do not increase from low to high interest rates. This suggests that interest rates have a generally low impact on the model output. The SHAP values for the unemployment rate show a broad distribution, and the equal distribution of blue and red dots indicates that the model treats the unemployment rate variable in a neutral way. Oil prices seem to have a slight negative impact on the model output, but both low and high values can have an impact on the model output.

However, an analysis of the SHAP values for the testing phase of the model shows that, similar to the training phase, PPI values have a low to high impact but a lower variance in the testing phase. Although the dollar exchange rate generally affects the model output positively, its effect is concentrated in a slightly narrower range than in the training phase. Interest rates have a small effect, as in the training phase, and this effect does not increase from low to high interest rates. The impact of the unemployment rate shows a similar distribution to the education phase, but both low and high values can have significant effects. Oil prices have a negative impact on model output, which may imply that higher oil prices reduce model output in some cases. As we saw in the training phase, the SHAP values of oil prices show an impact at both low and high values. This suggests that oil prices are an important feature for the model and can play a decisive role in output.

Overall, the plots of both stages show the consistency of the model in the learning process and the impact of the features. It also allows us to understand how well the model generalizes with new data and which features the model is more sensitive to. The SHAP values also reveal which features affect the model output the most and in which value ranges these features are more effective, which increases the explainability of the model and can be used to make better decisions.

Conclusion and Discussion

The performance of several deep learning and machine learning models is analyzed and compared in this research with an emphasis on CPI (consumer price index) predictions. The results highlight the study's significance and uniqueness. The results demonstrate that, when compared to other models, the deep learning model GRU performs the best. This suggests that CPI prediction may benefit greatly from the use of deep learning techniques. However, it is vital to pay attention to a few key areas in order to comprehend how this research varies from previous studies in the literature.

Although many previous studies have tackled generic inflation forecasting, this work concentrates on CPI forecasting from a theoretical standpoint. This indicates a theoretical difference in the research. Furthermore, a method that hasn't been used in certain research has been adopted: the usage of deep learning models (LSTM and GRU). The findings have important practical ramifications. This research offers a useful methodology for CPI forecasting, particularly for financial firms, central banks, and economic experts. Forecasts with greater accuracy are crucial for investment choices and economic policies. It's important to consider some of the study's shortcomings. For example, more research may be conducted on data constraints and model hyperparameter optimization. Furthermore, including economic variables outside the CPI estimate might represent a new field of study.

In the literature, comparable research are contrasted with the findings of this investigation. Rodriguez-Vargas (2020) forecasted inflation in Costa Rica using K-nearest neighbor, random forests, and LSTM. LSTM and KNN produced the greatest outcomes for him. The performance of the LSTM model in this investigation is comparable to the findings reported in the literature.

That being said, the kNN model's performance cannot be stated to be the same. However, Işığıçok et al. (2020) used CPI data to evaluate the ARIMA and ANN (artificial neural networks) approaches for future inflation estimates. It was shown in the research that the two approaches produced comparable outcomes. Similar outcomes are also seen in this study's performance of the ANN model. The LSTM model was used by Savitri et al. (2021) to predict inflation in emerging nations. They demonstrated how well the LSTM model predicts inflation. This study's LSTM model has achieved comparable results.

However, this research concentrates on a relatively narrow subject, namely CPI (consumer price index) forecasting, while several of the studies in the literature review part address generic inflation forecasting. The research differs conceptually from the others because of this. Furthermore, deep learning models like GRU (gated recurrent unit) and LSTM (long short-term memory) are included in this research. This research differs from several others since it includes deep learning models in addition to the more conventional machine learning or statistical models used in some other studies. Additionally, the findings of this research indicate a reduced mistake rate using the MAPE (mean absolute percentage error) metric, but the kNN model exhibits a larger error rate compared to previous studies. This also demonstrated a distinction.

It can provide some recommendations for further investigation. Studies may be carried out, in particular, using bigger data sets and other economic indicators. A comparative analysis of several deep learning architectures might potentially be intriguing. Lastly, research in this area may benefit from studies looking at how the model's outputs might be used to economic decision-making procedures. The usefulness of deep learning models in CPI forecasting is shown in this work, which adds to the body of knowledge. Future research may be motivated by the findings, which provide economic researchers and decision-makers a valuable foundation.

In terms of prediction accuracy, the results show that the GRU model performs better than the LSTM, RF, ANN, and KNN algorithms. It has been noted that it produces results. The study's conclusions will be very helpful to investors and decision-makers as they will enable them to increase the precision of their inflation projections. Future research may benefit from include factors like foreign currency reserves, the monetary policy of the central bank, and commodity prices (such those of wheat and gold) in contrast to the ones employed in this particular study. The study findings may also be compared using other time series, deep learning, and machine learning techniques.

The SHAP analysis conducted in this study provides guidance in understanding the economic dynamics behind the GRU model forecasts. Looking at the SHAP values, we can say that the factors that the GRU model takes into account most in economic forecasts are PPI and the dollar exchange rate. This information can be used to prioritize macroeconomic policy development and identify vulnerabilities to economic fluctuations. For example, if the PPI has a high positive impact, it may be important to develop policies to counter changes in producer prices and maintain price stability. Similarly, the magnitude of the impact of the dollar exchange rate on model output emphasizes the importance of foreign exchange policies and international trade on the economic outlook. If the dollar exchange rate is characterized by high SHAP values, the impact of exchange rate fluctuations on economic performance can be monitored particularly carefully, and macroeconomic measures can be taken in this area. In addition, the consistency between the specifications in the testing phase, which shows how well the model generalizes and applies what it has learned in the training phase, indicates that the model produces reliable results with external data sets as well. This supports policymakers' decisions based on real-world data and helps them forecast future economic conditions based on the model's predictions. SHAP values also allow for a more detailed examination of the relationships between economic indicators. For example, consider the interaction between the unemployment rate and interest rates. Such analysis allows policymakers to better assess the potential consequences of economic policy interventions and understand complex economic relationships. Overall, SHAP analysis is a powerful tool for understanding the contribution of features in GRU model forecasts, and this information enriches the economic forecasting and policymaking processes. A detailed analysis of the factors contributing to model outputs can help improve economic strategies and policies, providing insights needed to make more informed decisions.

Relating the results of the analysis to Turkey's current economic situation will increase the practical applicability and meaning of the study. Covering the period from January 2005 to June 2023, the analysis also includes various economic events and policy changes that the Turkish economy was exposed to during this period. During this period, important developments such as the global financial crisis, fluctuations in exchange rates, increases in energy prices, and the decisions taken by policymakers regarding the economy. It is important to evaluate the effects of these events and changes on the independent variables analysed in this study and, thus, how they affect the CPI. The global financial crisis of 2008–2009 had a significant impact on the Turkish economy. The sharp increases in exchange rates during the crisis period increased import costs, which directly put upward pressure on the CPI. This process shows that the GRU model is sensitive to exchange rate changes, and the importance of CPI forecasts increases during crisis periods. On the other hand, the rise in Brent oil prices increased energy costs throughout the analysis period. The increase in energy prices led to an increase in CPI by raising both direct consumer energy prices and general production costs. This situation emphasises the importance of taking into account the impact of energy prices on CPI and fluctuations in energy prices in forecasts. In addition, changes in interest rates and other monetary policy decisions of the Central Bank of the Republic of Turkey affected consumer loan interest rates and thus consumer expenditures from time to time. While increases in interest rates may have a downward impact on CPI by limiting consumer spending, decreases in interest rates may increase inflationary pressures by stimulating consumption. The timing and effects of policy changes are important factors to be considered in macroeconomic forecasts.

The ability of deep learning models to better understand and forecast macroeconomic data that changes due to economic events and policy changes explains why these models are preferred for forecasting critical economic indicators such as CPI. In particular, advanced artificial intelligence techniques such as the GRU model can effectively model the complexity of economic data and their changes over time. In this context, in economic modelling and forecasting, it is of great importance to understand not only mathematical accuracy but also how economic events and policies are reflected in the models. Relating the results of the analyses to Turkey's current and past economic situation shows that this study not only makes a theoretical contribution but also provides valuable insights for practitioners and policymakers. Such an approach allows the model to be used as an analytical tool that reflects economic realities and contributes to the shaping of economic policies, rather than being merely a technical tool. In conclusion, in the process of developing and applying macroeconomic forecasting models, model selection and the evaluation of forecasting results in the economic context allow for a greater impact of the studies in both scientific and applied fields. Therefore, in future studies, continuously evaluating and analysing the performance of deep learning models and other forecasting techniques on economic data in the light of current economic conditions and policy changes will improve the accuracy of economic forecasts and contribute to shaping economic policies more effectively.

References

Abraham, A. (2005). Artificial neural networks. Handbook of Measuring System Design, edited by Peter H. Sydenham and Richard Thorn, John Wiley & Sons, Ltd. ISBN: 0-470-02143-8.

Alkaff, M., Mustamin, N. F., & Firdaus, G. A. A. (2022). Prediction of crime rate in Banjarmasin City using RNN-GRU model. *International Journal of Intelligent Systems and Applications in Engineering*, *10*(3), 01-09.

Altunöz, U. (2022). The nonlinear and asymetric pass-through effect of crude oil prices on inflation. OPEC Energy Review, 46(1), 31-46. https://doi.org/10.1111/opec.12221

Azam, R., Muhammad, D., & Akbar, S. S. (2012). The significance of socioeconomic factors on personal loan decision (a study of consumer banking in local private banks in pakistan). SSRN Electronic Journal. https://doi.org/10.2139/ssrn.2167960

Basher, S. A. & Sadorsky, P. (2022). Forecasting Bitcoin price direction with random forests: How important are interest rates, inflation, and market volatility?. *Machine Learning with Applications*, 9, 100355.

Bayramoğlu, A. T. & Öztürk, Z. (2017). Inflation forecasting with ARIMA and gray system models. *Journal of Human and Social Sciences Research*, 6(2), 760-776.

Bhat, M. R., Jiao, J., & Azimian, A. (2021). The impact of covid-19 on home value in major texas cities. International Journal of Housing Markets and Analysis, 16(3), 616-627. https://doi.org/10.1108/ijhma-05-2021-0058

Boaretto, G. & Medeiros, M. C. (2023). Forecasting inflation using disaggregates and machine learning. *arXiv preprint* arXiv:2308.11173.

Breiman, L. (2001). Random forests. *Machine learning*, 45, 5-32.

Cheung, K. L. & Fu, A.W. C. (1998), Enhanced nearest neighbor search on the R-tree. *ACM SIGMOD Record 27*(3), 16– 21.

Cho, K., Van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation, *arXiv preprint* arXiv:1406.1078, https://doi.org/10.48550/arXiv.1406.1078.

Cuñado, J. & Gracia, F. P. d. (2005). Oil prices, economic activity and inflation: evidence for some asian countries. The Quarterly Review of Economics and Finance, 45(1), 65-83. https://doi.org/10.1016/j.qref.2004.02.003

Da Silva, I. N., Hernane Spatti, D., Andrade Flauzino, R., Liboni, L. H. B., dos Reis Alves, S. F., da Silva, I. N., ... & dos Reis Alves, S. F. (2017). Artificial neural network architectures and training processes (pp. 21-28). Springer International Publishing.

Dhanabal, S., & Chandramathi, S. J. I. J. C. A. (2011). A review of various k-nearest neighbor query processing techniques. *International Journal of Computer Applications*, *31*(7), 14-22.

Erdem, H. F. & Yamak, R. (2014). The degree of pass-through between the producer price index and the consumer price index. *Anadolu University Journal of Social Sciences, 14*(4), 1-13.

Firdauza, D. N. & Rahadian, Y. (2022). Determining factors of financial performance recovery in bri during the covid-19 pandemic. Economics Development Analysis Journal, 11(1), 49-60. https://doi.org/10.15294/edaj.v11i1.52990

Ganzagh, H. A., Samimi, A. J., Elmi, Z. M. & Tehranchian, A. M. (2023). Comparing Inflation Forecasting Models in Iran: New Evidences from ARDL-D-LSTM Model. *Iranian Journal of Economic Research*, *27*(93), 149-176.

Gao, Y., Wang, R. & Zhou, E. (2021). Stock Prediction Based on Optimized LSTM and GRU Models. *Scientific Programming*, 2021. https://doi.org/10.1155/2021/4055281

Gharaibeh, A. M. O. & Farooq, M. O. (2022). Determinants of bank lending rates: empirical evidence from conventional retail banks in bahrain. Banks and Bank Systems, 17(4), 140-153. https://doi.org/10.21511/bbs.17(4).2022.12

Gritli, M. I. (2021). Price inflation and exchange rate pass-through in tunisia. African Development Review, 33(4), 715-728. https://doi.org/10.1111/1467-8268.12599

Haryono, A. T., Sarno, R. & Sungkono, K. R. (2023). Transformer-Gated Recurrent Unit Method for Predicting Stock Price based on News Sentiments and Technical Indicators. *IEEE Access.*

Hatipoğlu, Ş., Belgrat, M. A., Degirmenci, A., & Karal, Ö. (2021). Prediction of Unemployment Rates in Turkey by k-Nearest Neighbor Regression Analysis. In 2021 Innovations in Intelligent Systems and Applications Conference (ASYU) (pp. 1-5). IEEE.

Helmy, O., Fayed, M. M. S., & Hussien, K. (2018). Exchange rate pass-through to inflation in egypt: a structural var approach. Review of Economics and Political Science, 3(2), 2-19. https://doi.org/10.1108/reps-07-2018-001

Henrich, A. (1994). A distance scan algorithm for spatial access structures. *In: Proceedings of the Second ACM Workshop on Geographic Information Systems*, pp. 136– 143.

Hjaltason, G.R., & Samet, H. (1999). Distance browsing in spatial databases. *ACM Trans. Database Sys. 24*(2), 265–318.

Hsing, Y. (2020). A simultaneous-equation model of estimating the response of the consumer price to exchange rate movements in thailand. Business and Economic Research, 10(1), 284. https://doi.org/10.5296/ber.v10i1.16406

Işığıçok, E., Öz, R., & Tarkun, S. (2020). Forecasting and technical comparison of inflation in Turkey with box-jenkins (ARIMA) models and the artificial neural network. *International Journal of Energy Optimization and Engineering (IJEOE)*, *9*(4), 84-103.

Jang, P. Y. & Beruvides, M. G. (2020). Time-varying influences of oil-producing countries on global oil price. Energies, 13(6), 1404. https://doi.org/10.3390/en13061404

Jamil, H. (2022). Inflation forecasting using hybrid ARIMA-LSTM model (Doctoral dissertation, Laurentian University of Sudbury).

Khan, A., Kandel, J., Tayara, H., & Chong, K. T. (2024). Predicting the bandgap and efficiency of perovskite solar cells using machine learning methods. Molecular Informatics, 43(2). https://doi.org/10.1002/minf.202300217

Kubheka, S. (2023). South African inflation modelling using bootstrapped long short-term memory methods. *SN Business & Economics*, *3*(7), 110.

Kitani, R. & Iwata, S. (2023). Verification of interpretability of phase-resolved partial discharge using a cnn with shap. IEEE Access, 11, 4752-4762. https://doi.org/10.1109/access.2023.3236315

Kosztowniak, A. (2022). Credit policy of commercial banks in eu and the asset quality of non-financial corporate loan portfolio in 2009-2021. European Research Studies Journal, XXV(Issue 1), 563-582. https://doi.org/10.35808/ersj/2871

Lee, K., Ayyasamy, M. V., Ji, Y., & Balachandran, P. V. (2022). A comparison of explainable artificial intelligence methods in the phase classification of multi-principal element alloys. Scientific Reports, 12(1). https://doi.org/10.1038/s41598-022-15618-4

Li, S., Huang, H., & Lu, W. (2021). A neural networks based method for multivariate time-series forecasting. IEEE Access, 9, 63915-63924. https://doi.org/10.1109/access.2021.3075063

Lin, Y. & Thompson, H. (2020). Exchange rate pass-through to consumer prices: the increasing role of energy prices. Open Economies Review, 32(2), 395-415. https://doi.org/10.1007/s11079-020-09601-7

Nguyen, T. T., Nguyen, H. G., Lee, J. Y., Wang, Y. L., & Tsai, C. S. (2023). The consumer price index prediction using machine learning approaches: Evidence from the United States. *Heliyon*, *9*(10).

Öniş, Z. & Özmucur, S. (1987), Inflation in Turkey, Istanbul: ITO Publication No: 1987-5.

Rodríguez-Vargas, A. (2020). Forecasting Costa Rican inflation with machine learning methods. *Latin American Journal of Central Banking,* *1*(1–4), 100012. https://doi.org/10.1016/j.latcb.2020.100012.

Rodríguez-Pérez, R. & Bajorath, J. (2020). Interpretation of machine learning models using shapley values: application to compound potency and multi-target activity predictions. Journal of Computer-Aided Molecular Design, 34(10), 1013-1026. https://doi.org/10.1007/s10822-020-00314-0

Sakashita, Y. & Yoshizaki, Y. (2016). The effects of oil price shocks on iip and cpi in emerging countries. Economies, 4(4), 20. https://doi.org/10.3390/economies4040020

Savitri, F. F., Siregar, R. F., Harianto, F. Y. & Napitupulu, H. (2021). Forecasting Inflation in Indonesia using Long Short Term Memory. *In 2021 International Conference on Artificial Intelligence and Big Data Analytics* (pp. 43-49). IEEE.

Saqib, A., Yasmin, F., & Hussain, I. (2023). Does the crime rate respond symmetrically or asymmetrically to changes in governance quality and macroeconomic variables? the application of linear and non-linear ardl. International Journal of Social Economics, 50(12), 1756-1776. https://doi.org/10.1108/ijse-09-2022-0625

Schröder, M. and Hüfner, F. (2002). Exchange rate pass-through to consumer prices: a european perspective. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.304939

Sek, S. K. (2019). Effect of oil price pass-through on domestic price inflation: evidence from nonlinear ardl models. Panoeconomicus, 66(1), 69-91. https://doi.org/10.2298/pan160511021s

Serrano-Cinca, C., Nieto, B. G., & López-Palacios, L. (2015). Determinants of default in p2p lending. Plos One, 10(10), e0139427. https://doi.org/10.1371/journal.pone.0139427

Sevgen, S. C. & Aliefendioğlu, Y. (2020). Mass Apprasial With A Machine Learning Algorithm: Random Forest Regression. *Bilişim Teknolojileri Dergisi, 13*(3), 301-311.

Soylu, Ö. B., Çakmak, İ., & Okur, F. (2018). Economic growth and unemployment issue: panel data analysis in eastern european countries. Journal of International Studies, 11(1), 93-107. https://doi.org/10.14254/2071-8330.2018/11-1/7

Sun, Q., Wang, Z., & Jia, N. (2022). Revisiting the dynamic response of chinese price level to crude oil price shocks based on a network analysis method. Entropy, 24(7), 944. https://doi.org/10.3390/e24070944

The Central Bank of the Republic of Turkey, https://evds2.tcmb.gov.tr/ Access Date: 04.07.2023

Tunalı, H. & Özkan, İ. (2016). An Empirical Analysis of the Relationship between Consumer Confidence Index and Consumer Price Index in Turkey. *Journal of Economic Policy Researches, 3*(2), 54-67.

Usupbeyli, A. and Uçak, S. (2020). The effects of exchange rates on cpi and ppi. Business and Economics Research Journal, 11(2), 323-334. https://doi.org/10.20409/berj.2020.252

Xu, W., Zhang, J., Zhang, Q. & Wei, X. (2017). Risk prediction of type II diabetes based on random forest model. *Proceedings of the 3rd IEEE International Conference on Advances in Electrical and Electronics, Information, Communication and Bio-Informatics,* AEEICB 2017, 382–386. https://doi.org/10.1109/AEEICB.2017.7972337

Yang, C. & Guo, S. (2021). Inflation prediction method based on deep learning. *Computational Intelligence and Neuroscience*, 2021.

Yu, C. D., Huang, J., Austin, W., Xiao, B., & Biros, G. (2015). Performance optimization for the k-nearest neighbors kernel on x86 architectures. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (pp. 1-12).

Financial disclosures

The author declared that this study did not receive any financial support.

Acknowledgements

The author have nothing to acknowledge of any persons, grants, funds, institutions.