Journal Homepage: -www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

Article DOI:10.21474/IJAR01/19257
DOI URL: http://dx.doi.org/10.21474/IJAR01/19257

RESEARCH ARTICLE

ENHANCING TIME SERIES FORECASTING ACCURACY WITH DEEP LEARNING MODELS: A COMPARATIVE STUDY

Sonia Rani, Ravinder Kaur and Chitra Desai
Department of Computer Science, National Defence Academy, Pune, India.

Manuscript Info

Manuscript History
Received: 09 June 2024
Final Accepted: 11 July 2024
Published: August 2024

Key words:-
Time Series Forecasting, Deep Learning Models, ARIMA, Random Forest, RNN, LSTM, GRU

Abstract

This study offers a detailed comparison of both traditional and advanced deep learning models in the context of time series forecasting, with a specific focus on ARIMA, Random Forest, Recurrent Neural Networks (RNNs), Long Short-Term Memory networks (LSTMs), and Gated Recurrent Units (GRUs). In line with open science principles, it utilizes publicly accessible datasets to guarantee the reproducibility of its findings and broaden their relevance. The research meticulously approaches preprocessing and thoroughly investigates model architectures and hyperparameters to establish solid benchmarks for performance evaluation. It uniquely employs the Root Mean Square Error (RMSE) as the primary metric to assess forecasting accuracy across different datasets. This singular focus on RMSE enables a precise understanding of model performance, highlighting the exact conditions under which each model excels or falls short, considering dataset characteristics such as size and complexity. Additionally, the study explores the interpretability of these models to provide insights into the decision-making processes underlying deep learning predictions. The results of this analysis yield essential recommendations for selecting optimal modeling techniques for time series forecasting, significantly advancing theoretical knowledge and practical applications in the field. By narrowing the gap between advanced machine learning techniques and their effective deployment in forecasting tasks, this study guides practitioners and researchers toward informed model selection based on RMSE performance.

Introduction:-

The burgeoning field of time series forecasting has witnessed significant advance-ments with the integration of both traditional statistical models and cutting-edge deep learning approaches. Deep learning models have shown remarkable capabilities in capturing intricate temporal dependencies [1], handling nonlinearity [2], and provid-ing highly accurate predictions. However, their utility and effectiveness in the context of time series forecasting compared to traditional machine learning methods remain a subject of investigation. This juxtaposition of methodologies offers a unique opportunity to explore the strengths and limitations inherent to each class of models when applied to the predictive analysis of temporal data. Central to the efficacy of these models is their ability to discern patterns and dependencies within time series data, which often encapsulates complex behaviors and trends relevant across a myriad of applications, from financial market predictions to energy consumption forecasting.

Corresponding Author:-Ravinder Kaur
Address:-Department of Computer Science, National Defence Academy, Pune, India.

This paper conducts a thorough evaluation of traditional statistical models, specifically ARIMA [3] and Random Forest [4], against advanced deep learning models, including Recurrent Neural Networks (RNNs) [5], Long Short-Term Memory networks (LSTMs) [6], and Gated Recurrent Units (GRUs) [7] . This selection spans a broad spectrum of modeling techniques, from time-tested statistical methods to neural networks adept at managing long-term dependencies and nonlinear data structures.

Committed to the principles of open science, the study utilizes publicly accessible datasets to ensure the reproducibility of results and relevance across diverse domains. This transparent approach not only promotes accountability but also facilitates a broader engagement with the research community, allowing for the validation and extension of the findings.

Through this investigation, the paper aims to provide a cohesive understanding of the relative strengths and weaknesses of each modeling approach within the context of time series forecasting, offering valuable insights into model selection and applica-tion in real-world scenarios.

Methodology:-

This study aims to evaluate the performance of traditional machine learning algo-rithms like ARIMA and Random Forest alongside advanced deep learning models such as RNNs, LSTMs, and GRUs in time series forecasting. Through a literature review, the research traces the development of forecasting techniques from classical methods to modern deep learning, highlighting the increasing importance of the latter. Utilizing publicly available datasets ensures the reproducibility and relevance of the findings. The study includes preprocessing these datasets for consistent analysis across models, followed by an in-depth examination of each model's architecture and hyperparameters to optimize their forecasting accuracy. Performance is evaluated using the Root Mean Square Error (RMSE), providing a direct measure of accuracy while also considering the datasets' characteristics. Additionally, an analysis of model interpretability seeks to enhance the understanding of advanced models' decision-making processes. The study concludes with targeted recommendations for model selection in time series forecasting, bridging the gap between theoretical advancements and practical application, and offering a framework for informed model selection.

Literature Review:-

The progression of time series data modeling showcases a remarkable evolution from foundational statistical methods to the advanced capabilities of deep learning. Early techniques such as moving averages and exponential smoothing [8] set the stage by offering insights into trends and seasonality within historical data. This period of exploration laid the groundwork for the groundbreaking development of the ARIMA model [3] in the 1970s by George Box and Gwilym Jenkins, providing a robust framework for managing autoregressive integrated moving average processes and revolutionizing time series analysis.Almost simultaneously, the late 1960s introduced state space models and the Kalman filter [9], presenting powerful tools for navigating the challenges of noisy data. The following decade, the 1980s, saw further advancements with Robert Engle's introduction of Autoregressive Conditional Heteroskedasticity (ARCH) [10] and Tim Bollerslev's extension with Generalized ARCH (GARCH) models [11], targeting the dynamic nature of volatility in financial series.

As the 1990s unfolded, neural networks began to leave their mark on time series forecasting, culminating in the advent of Recurrent Neural Networks (RNNs), which were specifically designed for sequential data analysis. Despite facing initial hurdles like the vanishing gradient problem, the introduction of Long Short-Term Memory Networks (LSTMs) by Sepp Hochreiter and Jürgen Schmidhuber [6] in 1997 marked a significant breakthrough, effectively learning long-term dependencies. This narra-tive continued to evolve with the proposal of Gated Recurrent Units (GRUs) by Kyunghyun Cho et al. [7] in 2014, simplifying the LSTM architecture while retaining its depth in capturing long-term data sequences.Deep learning models, particularly RNNs, LSTMs, and GRUs, have demonstrated unparalleled proficiency in dissecting sequential data, uncovering patterns, and learning dependencies that elude traditional models [1]. Their ability to autonomously identify and leverage relevant features from raw time series data [12] reduces the reliance on manual feature engineering, offering a more efficient approach to data analysis. These models excel in handling high-dimensional datasets [13], showcasing their capacity to manage and interpret complex data without sacrificing performance.

Moreover, deep learning techniques are adept at modeling the non-linear relation-ships inherent in time series data [2], a capability crucial for applications requiring nuanced forecasting, such as agricultural production. Their flexibility allows for fine-tuning to accommodate specific data characteristics like seasonality and volatility [14], ensuring their applicability across a variety of datasets and challenges. The inte-gration of diverse data types within a unified analytical framework [15] further un-derscores the versatility of deep learning in time series analysis, enabling a compre-hensive understanding of complex phenomena.Deep learning's superior predictive accuracy stems from intricate architectures capable of discerning complex patterns, leading to enhanced forecasting outcomes [2]. The scalability of these models [16] aligns with the demands of large-scale datasets, leveraging modern computing re-sources for efficient training and analysis. Certain deep learning models also offer real-time data processing [17], a critical feature for scenarios demanding immediate insights, such as financial trading or network security.

In addition to their operational advantages, deep learning models demonstrate ro-bustness against common data issues like noise and missing values [18], reinforcing their reliability and superiority in maintaining data integrity throughout the analysis. This journey from the foundational statistical methods to the cutting-edge techniques of deep learning models encapsulates the dynamic expansion and refinement of time series data analysis, highlighting a continuous trajectory towards more sophisticated and effective forecasting tools.

Data

The "daily-total-female-births.csv" dataset, available on the jbrown-lee/Datasets GitHub repository [19], provides daily female birth counts in California during 1959, offering a basis for evaluating deep learning models like RNNs, LSTMs, and GRUs against traditional methods such as ARIMA, Exponential Smoothing, and Random Forest. Its simplicity facilitates clear comparisons of predictive performance, compu-tational efficiency, and ease of use across these models. Serving as a critical tool in predictive analytics, this dataset enables comprehensive time series forecasting analysis, catering to researchers, educators, and practitioners in the field.

Stationarity

Stationarity tests [20] play a varied role across different modeling approaches like ARIMA, Random Forest, and deep learning models. For ARIMA models, which are based on statistical assumptions about time series data, ensuring stationarity is essen-tial. These models require the data's statistical properties, such as mean and variance, to remain constant over time, and tests like the Augmented Dickey-Fuller (ADF) [21] test are often used to check for stationarity before model fitting. In contrast, machine learning models like Random Forest do not require data to be stationary, as they do not make the same statistical assumptions and instead focus on capturing patterns through feature engineering. Similarly, deep learning models designed for sequence data, such as RNNs, LSTMs, and GRUs, can handle non-stationary data by learning complex patterns, including trends and seasonality, directly from the data. While making data stationary is not a prerequisite for these latter models, doing so can sometimes enhance their performance by simplifying the underlying structures in the data they need to learn.

The Augmented Dickey-Fuller (ADF) test is applied to the dataset here. ADF [20] is a statistical test used to determine whether a time series is stationary. Specifically, it tests for the presence of a unit root, a condition that indicates non-stationarity. The Augmented Dickey-Fuller (ADF) test results show a test statistic of -4.8083 and a very small p-value of approximately 0.000052, with critical values at -3.4487 (1% level), -2.8696 (5% level), and -2.5711 (10% level). Given that the test statistic is more negative than all the critical values and considering the low p-value, there is strong evidence to reject the null hypothesis of non-stationarity. This indicates that the time series is stationary, meaning its statistical properties do not change over time. This stationarity is a desirable property for many time series analysis techniques, sug-gesting that the series can be modeled without the need for differencing to stabilize its mean and variance.

Models

Before exploring into the practical experiment conducted to compare the forecasting abilities of ARIMA, Random Forest, RNN, LSTM, and GRU models, it is crucial to provide a brief overview of each model to set the context for the experimental analy-sis.

ARIMA (Autoregressive Integrated Moving Average) is a traditional statistical model used in time series forecasting. It combines autoregressive features with mov-ing averages and integrates differencing of observations to account for trends and non-stationarity in data. ARIMA models are widely recognized for their effectiveness in capturing linear relationships and seasonality in historical data, making them a staple in the arsenal of time series forecasting.

Random Forest is an ensemble learning method that operates by constructing mul-tiple decision trees during training and outputting the mode of the classes (classifica-tion) or mean prediction (regression) of the individual trees. Random forests correct for decision trees' habit of overfitting to their training set, providing a more robust prediction by averaging multiple trees. Despite not being inherently designed for time series data, Random Forest can be adeptly used for forecasting by incorporating time-based features.

Recurrent Neural Networks (RNNs) are a class of neural networks that are specifi-cally designed to recognize patterns in sequences of data such as time series, speech, text, or financial data. RNNs are capable of retaining information from previous inputs in their internal state, using it to influence the output of later inputs. This makes them particularly suitable for applications where the context and order of data points are crucial. Long Short-Term Memory (LSTM) networks are an advanced type of RNN specifically designed to avoid the long-term dependency problem, allowing them to remember information for longer periods of time. By incorporating mechanisms called gates, LSTMs can selectively remember or forget information, making them highly effective for complex time series forecasting tasks where long-term dependencies are prevalent.Gated Recurrent Units (GRUs) are a variation of LSTMs designed to simplify the model architecture without compromising the ability to capture dependencies in sequential data. GRUs combine the input and forget gates into a single update gate, reducing the complexity of the model and the computational burden, while still effectively modeling time series data.

These models represent a spectrum from traditional statistical methods to cutting-edge deep learning approaches in time series forecasting. Each has its strengths and ideal use cases, which the following practical experiment aims to explore and com-pare, providing insights into their relative performance and applicability to different forecasting tasks.

Experiment and Results:-

This section presents 5 experiments and their results for the models under study.

ARIMA

To determine the optimal ARIMA model for our time series analysis, a stepwise search aimed at minimizing the Akaike Information Criterion (AIC) was performed, for results see figure 1. The AIC helps in identifying a model that strikes the right balance between fitting the data well and keeping the model complexity low to avoid overfitting. This search led to the selection of the ARIMA(1,1,1) model as the best candidate, characterized by its simplicity and effectiveness in capturing the series' dynamics with just one autoregressive term, one differencing step, and one moving average term.

The ARIMA(1,1,1) model's implementation revealed important details about its performance, including coefficients and their significance, through the model sum-mary as shown in figure 2. This summary provided a comprehensive view of how well the model fits the data and ensured that the residuals met the necessary assump-tions, underlining the model's adequacy for forecasting with precision and reliability. This methodical approach to model selection underscores the analytical process in achieving accurate time series forecasting.

Random Forest

In exploring the applicability of machine learning models to time series forecasting, an experiment utilizing the Random Forest algorithm was conducted. The focus was on assessing the model's forecasting accuracy using synthetic time series data. This data was generated to include both a linear trend component and random noise, simulating real-world time series characteristics.

```
Performing stepwise search to minimize aic
 ARIMA(2,1,2)(0,0,0)[0] intercept   : AIC=2463.056, Time=0.87 sec
 ARIMA(0,1,0)(0,0,0)[0] intercept   : AIC=2650.760, Time=0.03 sec
 ARIMA(1,1,0)(0,0,0)[0] intercept   : AIC=2565.234, Time=0.08 sec
 ARIMA(0,1,1)(0,0,0)[0] intercept   : AIC=2463.584, Time=0.14 sec
 ARIMA(0,1,0)(0,0,0)[0]             : AIC=2648.768, Time=0.02 sec
 ARIMA(1,1,2)(0,0,0)[0] intercept   : AIC=inf, Time=0.69 sec
 ARIMA(2,1,1)(0,0,0)[0] intercept   : AIC=2461.271, Time=0.39 sec
 ARIMA(1,1,1)(0,0,0)[0] intercept   : AIC=2460.154, Time=0.26 sec
 ARIMA(0,1,2)(0,0,0)[0] intercept   : AIC=2460.722, Time=0.28 sec
 ARIMA(2,1,0)(0,0,0)[0] intercept   : AIC=2536.154, Time=0.17 sec
 ARIMA(1,1,1)(0,0,0)[0]             : AIC=2459.074, Time=0.11 sec
 ARIMA(0,1,1)(0,0,0)[0]             : AIC=2462.221, Time=0.06 sec
 ARIMA(1,1,0)(0,0,0)[0]             : AIC=2563.261, Time=0.04 sec
 ARIMA(2,1,1)(0,0,0)[0]             : AIC=2460.367, Time=0.16 sec
 ARIMA(1,1,2)(0,0,0)[0]             : AIC=2460.427, Time=0.27 sec
 ARIMA(0,1,2)(0,0,0)[0]             : AIC=2459.571, Time=0.10 sec
 ARIMA(2,1,0)(0,0,0)[0]             : AIC=2534.205, Time=0.08 sec
 ARIMA(2,1,2)(0,0,0)[0]             : AIC=2462.366, Time=0.43 sec

Best model:  ARIMA(1,1,1)(0,0,0)[0]
Total fit time: 4.186 seconds
```

Fig. 1:- Stepwise Search to Minimise AIC and Find Best ARIMA model.

SARIMAX Results

Dep. Variable:	y	**No. Observations:**	365
Model:	SARIMAX(1, 1, 1)	**Log Likelihood**	-1226.537
Date:	Thu, 29 Feb 2024	**AIC**	2459.074
Time:	06:48:22	**BIC**	2470.766
Sample:	0 - 365	**HQIC**	2463.721
Covariance Type:	opg		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
ar.L1	0.1252	0.060	2.097	0.036	0.008	0.242
ma.L1	-0.9624	0.017	-56.429	0.000	-0.996	-0.929
sigma2	49.1512	3.250	15.122	0.000	42.781	55.522

Ljung-Box (L1) (Q):	0.04	**Jarque-Bera (JB):**	25.33
Prob(Q):	0.84	**Prob(JB):**	0.00
Heteroskedasticity (H):	0.96	**Skew:**	0.57
Prob(H) (two-sided):	0.81	**Kurtosis:**	3.60

Fig. 2:- Model Summary (ARIMA Model).

The dataset comprised 365 data points, representing a sequence with a trend and variability. Feature engineering played a pivotal role in adapting this time series data for the Random Forest model. Specifically, lagged features were created to capture the series' temporal dependencies, a crucial step since machine learning models do not inherently consider the order of observations.

The data was then split into training and testing sets, adhering to a strategy that re-spects the sequential nature of time series data. This setup ensured that the model was trained on past data to predict future values. The Random Forest model was configured with 100 estimators, leveraging its ensemble learning capability to reduce overfitting and improve prediction accuracy.Upon training the Random Forest model on the engineered features, predictions were made on the test set. The model's perfor-mance was evaluated using the Mean Squared Error (MSE) metric, which quantifies the average squared difference between the observed actual outcomes and the predictions. This metric was chosen for its ability to penalize large errors more heavily, offering a clearer picture of the model's predictive accuracy.

This experiment's findings provide insight into the effectiveness of Random Forest regressors in handling time series data, particularly when proper feature engineering is employed to incorporate temporal information. Through this methodical approach, the study advances the understanding of machine learning applications in forecasting, underlining the potential of ensemble methods like Random Forest in achieving accurate time series predictions.

Deep Learning Models: RNN, LSTM, GRU

This paper explores the efficacy of deep learning techniques in time series forecasting, utilizing the "daily-total-female-births.csv" dataset which comprises 365 daily observations of female births over a year. The investigation encompasses three key models tailored for sequential data analysis: Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs), each offering distinct advantages in processing time series data.

The analysis begins with an exploration of RNNs, noted for their ability to use his-torical data to forecast future events. The dataset was transformed into sequences suitable for the RNN model, setting a foundation for evaluating deep learning's potential in forecasting. This step established a benchmark for deep learning performance in time series analysis.Following the RNN exploration, attention shifts to LSTM networks, which are engineered to address RNNs' limitations in capturing long-term dependencies. Optimizing the LSTM model involved adjusting its architecture and hyperparameters to leverage its superior memory capabilities, aiming to enhance forecasting accuracy beyond what was achieved with the RNN model. The LSTM's performance was meticulously evaluated, providing insights into its effectiveness in modeling complex temporal patterns within the dataset.

The final phase of the study examines GRUs, which streamline the LSTM design without compromising the ability to model temporal dependencies. The simplicity of GRUs presents an opportunity for computational efficiency, making them an attrac-tive option for time series forecasting. Like its predecessors, the GRU model under-went preprocessing, training, and optimization, with its forecasting accuracy critically assessed. By systematically evaluating the performance of RNNs, LSTMs, and GRUs on the same dataset, this paper offers a comprehensive view of the potential and limitations of each deep learning technique in the context of time series forecasting.

Interpretation and Conclusion:-

Table 1 presents RMSE values for various forecasting models, with Figures 2 to 6 depicting their respective prediction graphs. The GRU model demonstrates superior accuracy, achieving the lowest RMSE of 6.37, signifying its effectiveness in capturing the dataset's underlying temporal patterns.

Table 1:- RMSE for.

Sr.No	Model	RMSE
1	ARIMA(1,1,1)	7.50
2	Random Forest	7.20
3	RNN	6.46
4	LSTM	6.55
5	GRU	6.37

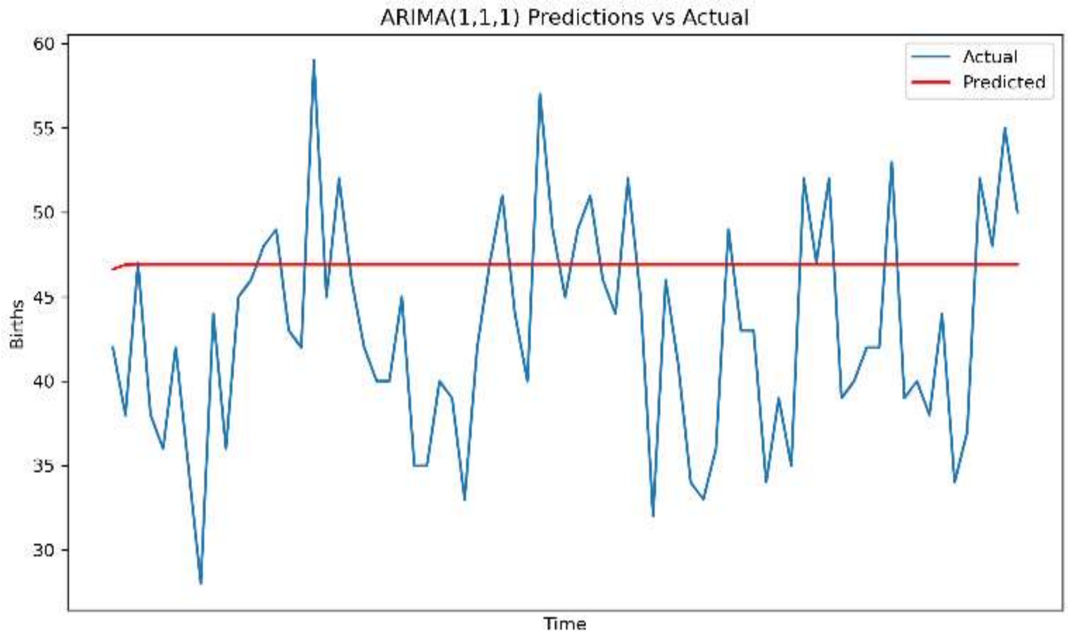

Fig. 3:- Arima(1,1,1) Prediction.

Close behind, RNN and LSTM models exhibit comparable accuracy with RMSE values of 6.46 and 6.55, respectively, suggesting that the increased complexity of LSTM does not yield significant performance benefits over RNN for this dataset.

Traditional models, Random Forest and ARIMA, recorded higher RMSEs of 7.20 and 7.50, indicating less precision in their predictions. Random Forest's marginally better performance over ARIMA suggests that even without being specifically de-signed for time series data, machine learning approaches can outperform classical statistical models through effective feature engineering.

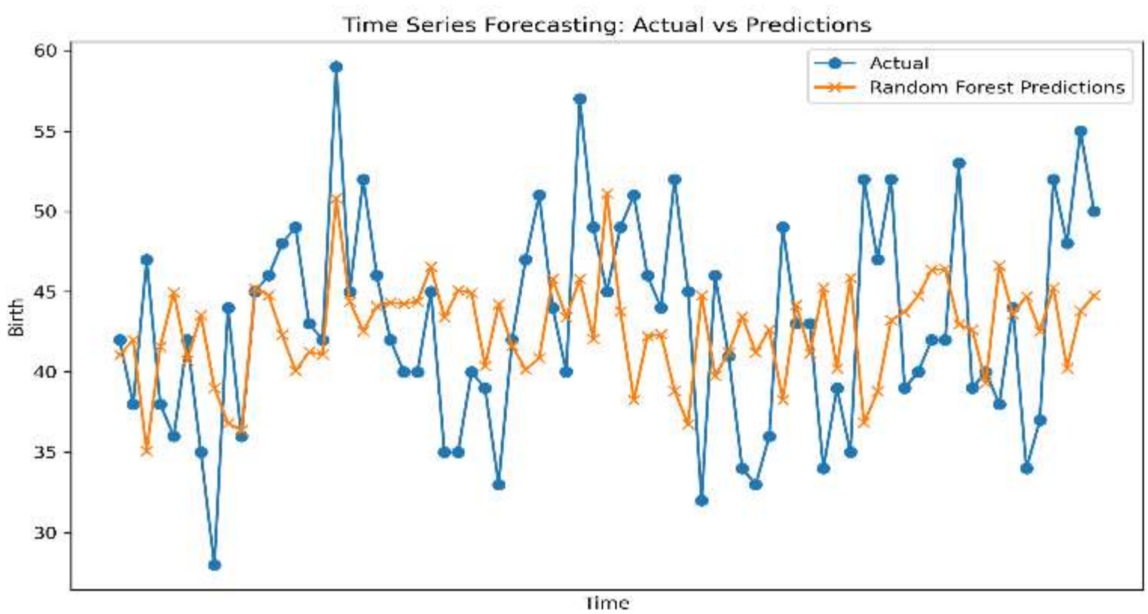

Fig. 4:- Random Forest Prediction.

The collective outperformance of deep learning models (RNN, LSTM, GRU) over traditional ones (Random Forest, ARIMA) underscores their superior capability in modeling complex dependencies within the data, a critical aspect of time series fore-casting. GRU's top performance highlights its optimal balance between model com-plexity and learning capacity, rendering it the most fitting choice for this dataset.

Despite its foundational role in time series analysis, the ARIMA model's higher RMSE points to its limitations in fully capturing the dataset's dynamics compared to the machine learning and deep learning approaches. The slight edge of Random For-est

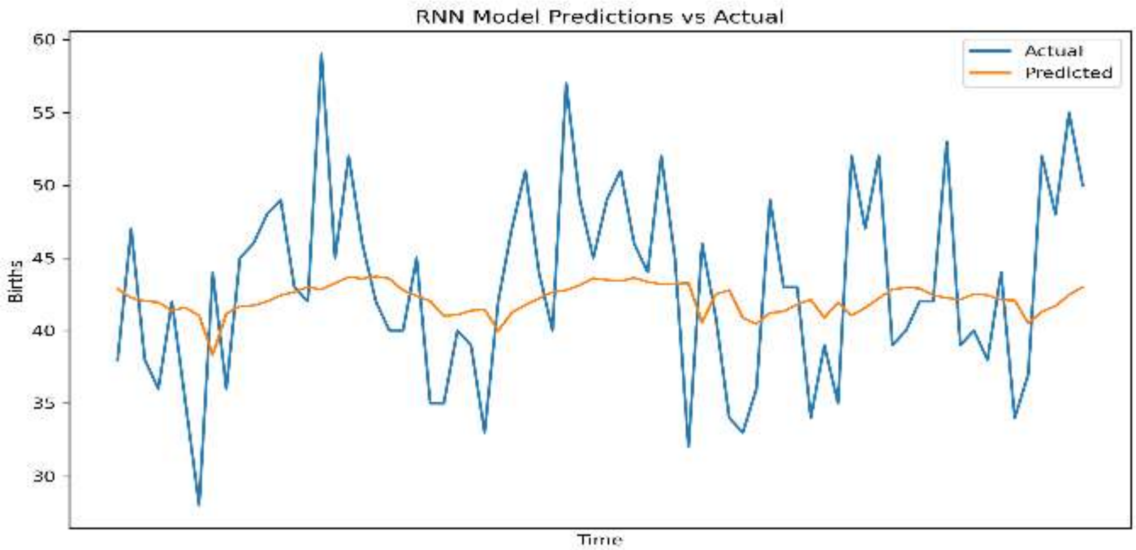

Fig. 3:- RNN Prediction.

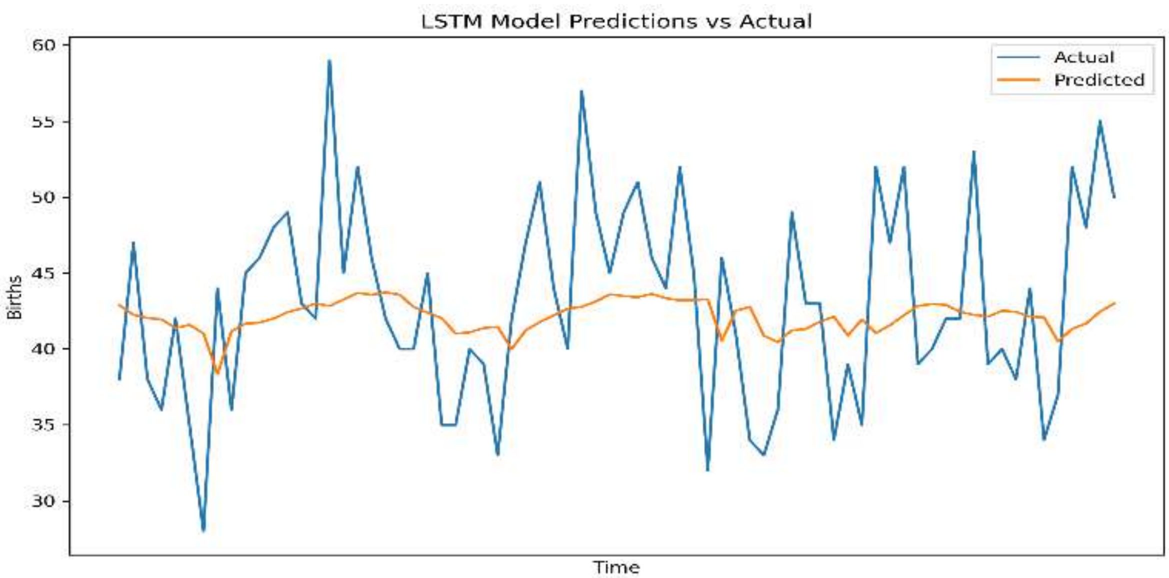

Fig. 6:- LSTM Prediction.

over ARIMA reinforces the potential of ensemble learning strategies in extracting relevant patterns from time series data. This comparative analysis emphasizes the importance of selecting models based on specific dataset characteristics, acknowledging that no one model universally excels. The close RMSE scores across the evaluated models advocate for a thorough examination of different architectures, particularly the nuances between deep learning models, which may lead to notable improvements in forecasting accuracy.

In conclusion, the GRU model demonstrates superior precision on this dataset, highlighting key factors like model complexity, computational efficiency, and predic-tive accuracy. These findings are crucial for choosing appropriate forecasting models, advocating for a balance between deep learning's advanced capabilities and the straightforward nature of traditional approaches.

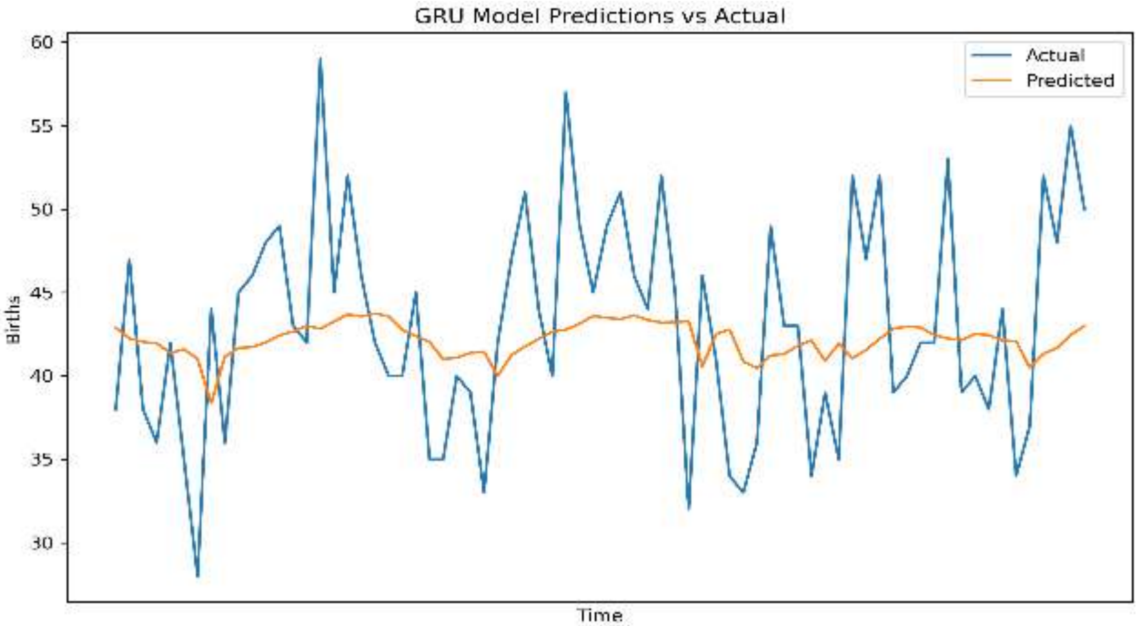

Fig. 7:- GRU Prediction.

References:-

1. Hussain, M.A., et al.: Dynamic Slicing Optimization In 5g Networks Using A Re-cursive Lstm Mechanism With Grey Wolf Optimization. J. Theor. Appl. Inf. Tech-nol. (2024)
2. Parreño, S.J.E., Anter, M.C.J.V.: New Approach For Forecasting Rice And Corn Production In The Philippines Through Machine Learning Models. Multidiscip. Sci. J. (2024)
3. Box, G.E.P., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time Series Analysis: Fo-recasting and Control. John Wiley & Sons, Hoboken, NJ (2015).
4. Breiman, L.: Random Forests. Machine Learning 45(1), 5–32 (2001). https://doi.org/10.1023/A:1010933404324.
5. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Representations by Back-propagating Errors. Nature 323, 533–536 (1986). https://doi.org/10.1038/323533a0.
6. Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. Neural Computation 9(8), 1735–1780 (1997). https://doi.org/10.1162/neco.1997.9.8.1735.
7. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734. Association for Computational Linguistics, Doha, Qatar (2014). https://doi.org/10.3115/v1/D14-1179.
8. Montgomery, D.C., Johnson, L.A., Gardiner, J.S.: Forecasting and Time Series Analysis. McGraw-Hill, New York (1990).
9. Kalman, R.E.: A New Approach to Linear Filtering and Prediction Problems. Jour-nal of Basic Engineering 82(1), 35–45 (1960). https://doi.org/10.1115/1.3662552.
10. The 1980s further expanded the arsenal of time series models with the introduc-tion of Autoregressive Conditional Heteroskedasticity (ARCH) by Robert Engle and later, Generalized ARCH (GARCH) models by Tim Bollerslev, addressing the modeling of time-varying volatility, especially in financial series.
11. Bollerslev, T.: Generalized Autoregressive Conditional Heteroskedasticity. Journal of Econometrics 31(3), 307–327 (1986). https://doi.org/10.1016/0304-4076(86)90063-1.
12. Ke, L., Fang, L., Meng, J., Peng, J., Wu, J., Lin, M.: Identification Of The Aging State Of Lithium-Ion Batteries Via Temporal Convolution Network And Self-Attention Mechanism. J. Energy Storage (2024)
13. Telea, A., Machado, A., Wang, Y.: Seeing is Learning in High Dimensions: The Synergy Between Dimensionality Reduction and Machine Learning. SN Comput. Sci. (2024)

14. Liang, J., Bian, X., Jia, L., Liang, M., Kong, R., Zhang, J.: A Method for Grasp Detection of Flexible Four-Finger Gripper. SSRN Electron. J. (2024). https://doi.org/10.2139/ssrn.4729642
15. Ambroise, C., Grigis, A., Houenou, J., Frouin, V.: Interpretable and integrative deep learning for discovering brain-behaviour associations with stability analysis. ⟨hal-04471394⟩ (2024)
16. Vaselli, F., Cattafesta, F., Asenov, P., Rizzi, A.: End-to-end simulation of particle physics events with Flow Matching and generator Oversampling. ar-Xiv:2402.13684 (2024)
17. Agwu, O.E., Alkouh, A., Alatefi, S., Azim, R.A., et al.: Utilization of machine learning for the estimation of production rates in wells operated by electrical sub-mersible pumps. J. Pet. Explor. Prod. Technol. (2024)
18. Xiao, Y., Adegok, M., Leung, C.S., Leung, K.W.: Robust noise-aware algorithm for randomized neural network and its convergence properties. Neural Netw. (2024)
19. Dataset: GitHub. jbrownlee/Datasets, "daily-total-female-births.csv". (n.d.)Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice.
20. Brockwell, P.J., Davis, R.A.: Introduction to Time Series and Forecasting. Springer, New York (2002).
21. Dickey, D.A., Fuller, W.A.: Distribution of the Estimators for Autoregressive Time Series with a Unit Root. Journal of the American Statistical Association 74(366), 427–431 (1979). https://doi.org/10.1080/01621459.1979.10482531.