DeepSmart: A Deep Learning Strategy for Real-time B5G/6G Edge Analytics and Anomaly Detection

Bruce Mareri, Ruijie Ou
School of Computer Science and Engineering
University of Electronic Science and Technology of China (UESTC)
Chengdu, China
Email: bmareri@gmail.com, ORJ@uestc.edu.cn

Yu Pang
School of Electro-optical Engineering
Chongqing University of Posts and Telecommunications (CUPT)
Chongqing, China
pangyu@cqupt.edu.cn

***Abstract*—Wireless communication has increased significantly in recent years. To address future connectivity requirements, some researchers are focused on Beyond 5G (B5G) and Sixth-Generation (6G) wireless technology that capitalizes on Internet of Things (IoT) technologies to convert sensory data into actionable knowledge. Intelligent factories that are networked require real-time, low-latency applications. As IIoT devices become more widely deployed, real-time data processing at the network edge rather than in cloud data centers is critical. As a result, deep learning may be a viable choice for real-time processing. This research proposes DeepSmart, a deep learning-powered framework for IIoT forecasting and anomaly detection is proposed in this study. DeepSmart's hierarchical architecture for processing correlated time series workflow model, constructed with long short-term memory (LSTM) as a significant component, is demonstrated. DeepSmart is evaluated using real-world datasets, and the results demonstrate that it outperforms established classical approaches in forecasting.**

***Keywords*-6G networks; deep learning; network slicing; anomaly detection; demand forecasting;**

I. Introduction

55.7 billion IoT-enabled devices will be part of an IoT infrastructure by 2025. According to [1], enterprises must prepare a flexible, scalable, data-driven IoT infrastructure to connect with mission-critical infrastructure and business innovations on edge devices for equipment maintenance and monitoring manufacturing operations [2]. Data and control signals must be transmitted effectively to establish autonomous, networked, responsive, and effective systems to achieve this objective. Connectivity allows them to share and evaluate data. It converts industrial machinery and businesses into intelligent systems that improve performance, productivity, and adaptability [3].

Time-Sensitive Networking (TSN) supporting services that include Ubiquitous mobile ultra-broadband (*uMUB*), Mobile broadband reliable low latency communication (*MBR-LLC*), Ultra-high speed with low latency communications (*uHSLLC*), Massive machine-type communication (*mMTC*), Ultra-high data density (*uHDD*), Human-centric services (*HCS*), and Multi-purpose services (*MPS*) will be used in future networks [4]. In an Industrial Scenario, the ideal system should prioritize time-sensitive control data for factory floor determinism and dependability, which necessitates traffic scheduling and prioritization. Time-sensitive data sharing can alleviate congestion, allowing for more efficient plant-production monitoring, management, and reporting [5]. Translating data into useful knowledge remains challenging in time-series monitoring and prediction. For starters, the industrial time series is vast and expanding. Second, IoT users anticipate real-time data processing. Third, both short- and long-term forecasts must be reliable [6].

Edge devices are located close to IoT Sensor nodes offering low-latency, high-bandwidth access and compute services. While computationally efficient, traditional forecasting methods fail to accurately capture the dynamic link between the objective and the time series current data [7]. Deep learning has emerged as the dominant technique for solving some of the most complex challenges in artificial intelligence. Speech recognition and translation, as well as other tasks requiring sequential learning, have significantly benefited from advances in Recurrent Neural Networks, particularly long short-term memory (LSTM)-based models [8].

To address the challenges mentioned earlier, Based on the latest edge computing and deep learning, we focus on forecasting for edge devices to achieve accurate and timely anomaly detection for multi-type time-series data while considering the increase in differentiated resource requirements. Outliers are targeted using the proposed method's hierarchy and seasonal decomposition elements, which detect anomalies. The following are some of the most significant outcomes of our efforts:

- We present DeepSmart, a conceptual deep learning-powered framework for predictive analysis and anomaly detection of Beyond 5G/6G (B5G/6G) sensory data.
- Implement an attention-based recurrent neural network that uses LSTM to capture the fine-grained properties of time series data and uses the LSTM module to accurately and timely detect anomalies.
- Finally, we examine alternatives and conduct extensive

experiments to evaluate the proposed framework. Furthermore, the testing results indicate that DeepSmart can accurately forecast high-accuracy demand data.

The remaining sections of the paper are structured as follows. In Section II, we investigate the related literature. The DeepSmart architecture and deep learning model design are proposed in Section III. Experiments on forecasting and anomaly detection are discussed in Section IV. Finally, in Section V, we conclude the paper and discuss potential future work.

II. Related Works

In [9], the ideal ANN for accumulating predictions from data streams and time-series data is presented. It is proposed to combine LSTM with Naive Bayes models. Parallel retrieval and validation of these trends and forecasts via anomaly detection. The LSTM predicts data streams, whereas the Naive Bayes model detects anomalies based on the LSTM's predictions. In [10], the application of AE and LSTM for forecasting the energy production of solar systems is explained. The deep feature fusion method highlights the effectiveness of AEs in feature extraction and feature learning. By layering Deep AE (noise reduction) and contractive AE (enhanced feature recognition), the ability to learn features with more negligible effect from background noises is improved. Because of this, [9] suggests the implementation of LSTM in complicated IoT scenarios to identify long-term data relationships. LSTM is an RNN variation that incorporates memory units. These memory units can recall significant prior states while forgetting the insignificant ones. To predict the behavior of energy systems in smart grids [11], it is required to implement more intelligent systems to make precise forecasts of future energy consumption. Using a weight-sharing recurrent neural network and parallel streaming analytical programming, a real-time public transportation crowd prediction system is developed in [12] where real-time analysis is required to respond promptly to unexpected incidents. Authors In [13] present a CNN-based real-time solution for traffic sign detection and recognition.

Regarding anomaly detection, we established that an Autoregressive integrated moving average (ARIMA) is still used to identify outliers in the time-series data stream [14, 15]. In these works, time-series decomposition was utilized as a vital tool for analyzing data set abnormalities. Anomaly detection has been applied to periodic and non-periodic multivariate time series points. Authors in [16] employ LSTM (long short-term memory) to detect anomalies.. Such research demonstrates a mechanism for unsupervised anomaly detection utilizing mean absolute relative error (AARE).

This research aims to build an adaptive real-time forecasting model to support anomaly detection for elastic demand in IIoT scenarios. Our approach undertakes several activities to improve the process of forecasting: seasonal adjustment, error cycle elimination, trend analysis, residual verification, and irregular cycle amortization. As a result, enhances forecasting and anomaly detection.

III. DeepSmart Proposal

A. Proposed Framework

This study focuses on smart grids and buildings with time-series sensor readings. We integrate edge computing architecture with advanced deep learning models to address monitoring challenges, forecast IIoT real-time sensory data, draw correlations from historical and operational data, and conduct forecasting tasks. DeepSmart uses an architecture based on recurrent neural networks to distinguish between essential and non-essential characteristics. DeepSmart utilizes a technique of temporal attention to forecast future demand and identify anomalies based on decomposition remainders. The architecture of DeepSmart is depicted in Figure 1. Additional details are presented below:

- *Edge Devices:* Edge devices, such as cars, sensors, and smart buildings, are often agents and clients. Multiple IoT apps execute and perform sensing activities concurrently in the utility layer, continuously creating and transmitting time-series data to the linked edge devices.
- *Forecasting model:* That is adapting time series data to elastic demand scenarios. It includes an Augmented Dickey-Fuller (ADF) model and a deep learning model that can predict network demand behavior regardless of where it originates. Existing forecasting algorithms benefit from these modifications. Elastic demand scenarios cannot be forecasted using current methodologies because they lack the necessary corrective variables.
- *Unified Computing Server (UCS) edge database:* Servers with high computing capability are typically used as the Cloud Aggregator. Which are utilized to collect data, do preliminary processing, and direct the data flow to the network level. UCS and edge databases can process and store sophisticated data in the core layer. Data from multivariate time series can be mined for features and patterns using the proposed deep learning models offered by the UCS components. Based on its comprehensive deep learning models, UCS can effectively anticipate the future time series of heterogeneous IoT sensor data, enabling intelligent routers and gateways to conduct real-time monitoring and predictive analytics.

This section presents the attention-based model utilized by LSTM. Fine-grained time series data is collected, and anomalies are detected using an LSTM model. Then, we present an ADF technique for improving predicting performance.

B. Augmented Dickey-Fuller Model

A time series may be expressed mathematically if the samples show autocorrelation [18]. For example, if we look at the autocorrelation between the current and previous values

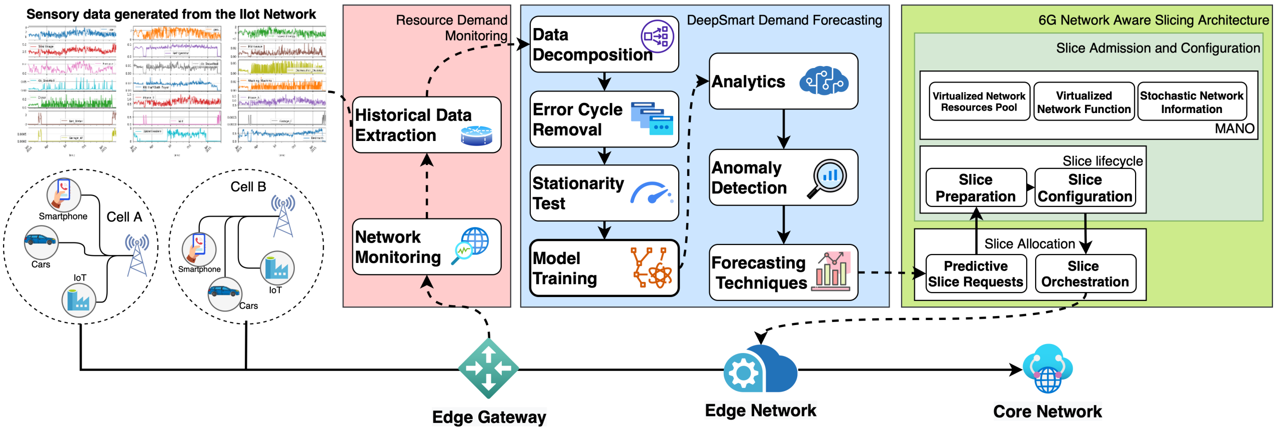


Figure 1: Interaction between the proposed Deepsmart approach and 6G network slicing reference framework. [17]

of $Y_t \epsilon Y_{t-1}$, we see that they are highly interdependent. Correlation between a pair of samples r over a given period t is calculated using the autocorrelation function (ACF). For each pair of observations $(y_1, y_2, y_3, \ldots)$, the ACF is calculated for each pair $((y_1, y_2), (y_3, y_4), (y_5, y_6), \ldots, (y_n, y_{n+1}))$ [19]. An ACF's autocorrelation coefficients can be represented using Equation 1, which specifies the time series interval and average as T and $\bar{y}$, respectively.

$$r_k = \frac{\sum_{t=k+1}^{T}(y_t - \bar{y})(y_{t-k} - \bar{y})}{\sum_{t=k+1}^{T}(y_t - \bar{y})^2} \tag{1}$$

1) Data Decomposition: Deconstructing the time series Z_t, where $t \in \{1, \ldots, N\}$, into its Seasonality (S_t), Trend (R_t), and Cycle error (E_t), while maintaining a constant mean and variance is the goal of the data decomposition task. These cumulative features are represented as $Z_t = S_t + R_t + E_t$. *Seasonality* [20] reflects annual, monthly, weekly, or daily oscillations in a time series. Where $S_{t+p} = S_t$ represents a succession of time-dependent data during period p. A seasonal component S_t defined by a moving average technique is eliminated from the initial data $Z_t - S_t = C_t + R_t + E_t$ where C_t represents the cyclic components. *Trend* R_t of a time t indicates the behavior fluctuation and frequency of a time series defined by: $R_t = \bar{y} + Z_t$, where $\bar{y} = \frac{1}{n}\sum_{t=1}^{n} y_t$ and where $\bar{Y}$ is expected for each $Y_1, Y_1, \ldots, Y_n$. Represented in Figure 2a.

2) Error Cycle Removal: Trend R_t and seasonality S_t evaluations are inhibited by low-amplitude cycles. An effective forecasting system should eliminate erroneous cycles and random noises N_t. Thus, the proposed ADF model employs a Moving Mean to investigate the trend and eliminate low-amplitude data represented by $L_t = S_t + R_t + N_t$ where L_t is the data at time t, with $t \in \{1, \ldots, T\}$. Moreover, outliers can be eliminated using a Loess [21] approach to estimate the relationship between nonlinear variables (See Figure 2a).

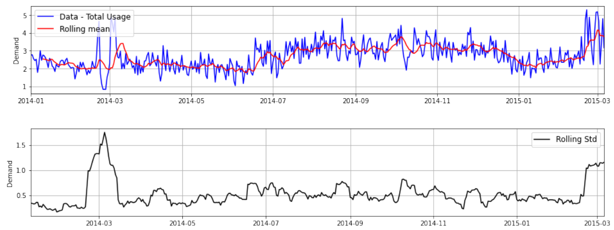
3) Stationarity Test: The Stationarity Test evaluates a data time series long-term, exponential, or regular patterns. It should be emphasized that approaches that rely on time series that are not always stationary may be unable to forecast effectively based on variations in gradient levels over short or long periods. In this scenario, we consider a time series (Y_i for $i = 1, 2, \ldots, n$) to be a sum total of the deterministic trend T_i, and random walk W_i, explained by the equation $W_k = W_{k-1} + U_k$, where U_k is defined as the collection of random variables with 0 mean and constant variance σ^2. Therefore to calculate the stationary error E_i, we follow the equation $Y_i = T_i + W_i + E_i$. Stationarity analysis is used in the training set to increase model reliability and standardize the samples. After the stationary evidence, cycle errors that destabilize forecasting capacity can be eliminated as presented in Figure 2b. Following these ADF procedures, the processed data are transferred to the forecasting technique discussed in the next section.

C. LSTM Forecasting and Anomaly Detection Model

This paper utilizes a variation of a multivariate LSTM [22] to forecast sensor time series data to detect abnormalities accurately. Forecasting uses past samples to estimate future moves. These approaches follow seasonality and trend. However, they cannot handle time series with uncertain patterns. Using the ADF model as a forecasting method should improve its performance.

In order to identify anomalies, we employ an anomaly score M_n, which is determined by the equation $M_n = (\rho_n - \delta)^T \sigma^{-1} (\rho_n - \delta)$. For each of the sequences X^i, a reconstruction error defined by $\rho_n = |x_n^i - x_n^{'i}|$ and error vectors ρ_n, are utilized to estimate the parameters δ and σ of a Normal distribution $N(\delta; \sigma)$ using Maximum Likelihood Estimation method. Furthermore, to detect if a parameter θ at a given point in a sequence time is normal or abnormal in an unsupervised environment. Precision P and Recall R

(a) Visualization of the Seasonality and Trend in the pre-processed time series



(b) Stationarity Test and Moving Mean application

Figure 2: Visualizations for the ADF model's operations: *Decomposition* (Section III-B1), *Error cycle removal* (Section III-B2), and *Stationarity test* (Section III-B3).

are applied on the anomaly score which is derived by the equation $M_n \geq \zeta$ ($\zeta = max\ F_\theta = \frac{(1+\theta^2)\times P\times R}{\theta^2 P+R}$).

D. Slice Allocation

All access points equipped with edge computing servers upload their computation jobs to the Slice Aware Network slicer for the considered Scenario. A projected network slice is created to ensure that service requirements are satisfied, with bandwidth and processing resources allocated. Based on the prediction task information derived by the ADF model's forecasting values. Each forecast value represents a client's bandwidth need for a certain time period. Thus, an algorithm should define the most appropriate slice structure in the network infrastructure to compose a network slice for the time periods defined based on the predetermined slice network configuration. Although , the Slice configuration is outside the scope of this study. We refer interested readers to [17] for more detailed information on slice allocation.

IV. Experiment and Results

A. Evaluation Setup

Using DeepSmart, we undertake two experiments looking into time series prediction. We explore data preparation, parameter configuration, and comparison metrics. Then, we compare the prediction outcomes of DeepSmart and other baseline techniques to diverse sensory time series. For our analysis, we use a real-world, crowd-sourced dataset [23] consisting of a collection of sensor-collected time-series datasets from various fields. Examples of the power demand dataset include energy meter readings. These datasets include both regular and erroneous subsequences. The abnormal subsequences in the power demand dataset indicate a malfunctioning electricity meter. These datasets are essential to training the model for anomaly detection. The forecasting and anomaly detection algorithm is implemented in Python, using Keras and Tensorflow as backend.

B. Evaluation Metrics

For evaluating the forecasting performance of the models, we use *mean absolute percentage error (MAPE), root mean squared error (RMSE)* representing the error rate by the square root of MSE, and *R-Squared (R^2 Score)* that are derivatives of Mean Absolute Error (MAE) $\sum_{i=1}^{n}|y_i - \hat{y}|$ and Mean Squared Error (MSE) $\sum_{i=1}^{n}(y_i - \hat{y})^2$ that represents the coefficient of how well the values fit compared to the original values as defined in Table I.

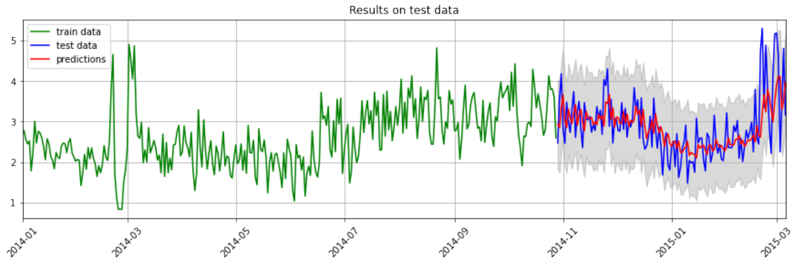
Table I: Performance Metrics

Metric	Notation
Mean Absolute Percentage Error	MAPE = $\frac{100\%}{n}\sum_{i=1}^{n} = \left\lvert\frac{y_i-\hat{y}}{y_i}\right\rvert$
Root Mean Squared Error	RMSE = $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y})^2}$
R^2 Score	R^2= $1-\frac{\sum_{i=1}^{n}(y_i-\hat{y})^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2}$

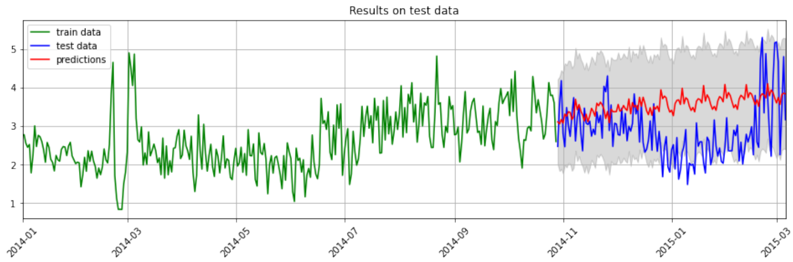
C. Results

In a similar simulation environment, the performance of ARIMA [24], Auto ARIMA [25], SARIMAX [26], LSTM univariate [27] models were compared to our model. All models are commonly employed for general forecasting and anomaly detection tasks. The performance of each model is tested using data from actual environments (such as power demand).

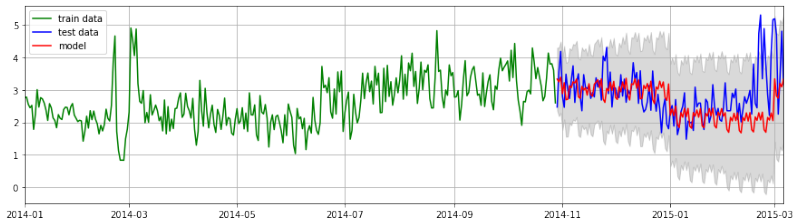
1) Training Performance: First, we perform accuracy training on the models using the dataset. Moreover, measure the performance of the models against each other considering computational time to perform forecasting and accuracy in training for the forecasting. From Table II it is evident that the



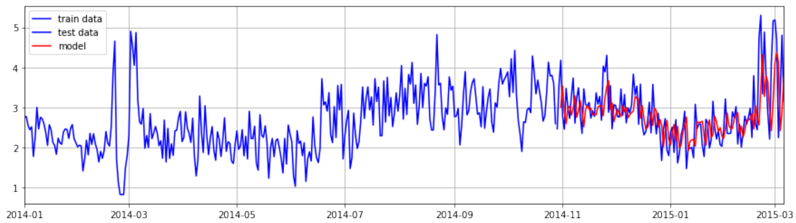
(a)



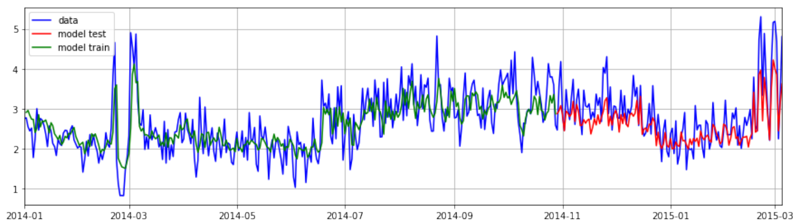
(b)



(c)



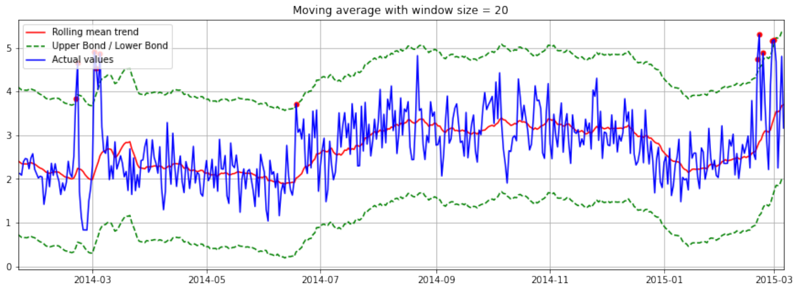
(d)



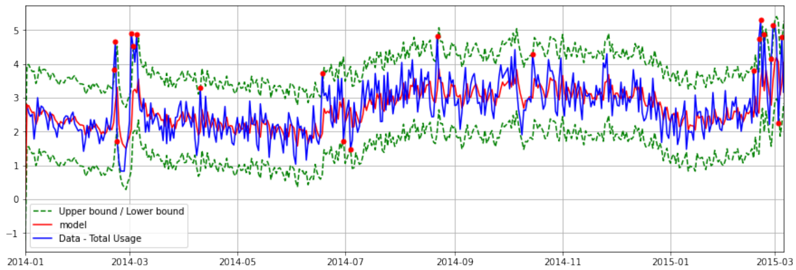
(e)

Figure 3: Forecasting comparison between (a) ARIMA [24], (b) Auto ARIMA [25], (c) SARIMAX [26], (d) LSTM univariate [27] and (e) DeepSmart

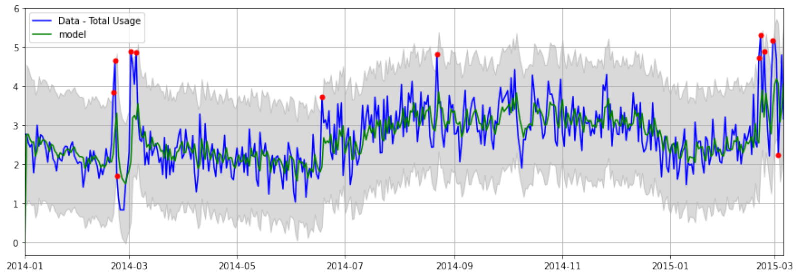
DeepSmart model has performed significantly well compared to the other models with the lowest MAPE of 0.123 and RMSE scores of 0.48001. As a result, this demonstrates that the DeepSmart model can be used in real-time forecasting scenarios.



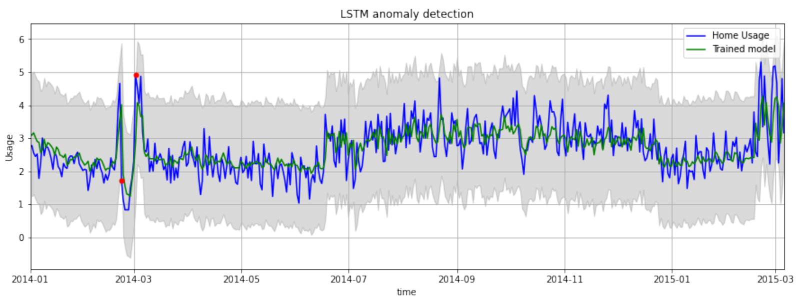
(a)



(b)



(c)



(d)

Figure 4: Anomaly detection comparison (a) ARIMA [24], (b) Auto ARIMA [25], (c) SARIMAX [26], and (d) DeepSmart

2) Forecasting Results: Second, we compare the predictive accuracy of the proposed model to that of alternative approaches. We base the prediction accuracy performance on the RMSE, MAPE, and R^2 Score of the models on the training set and testing sets. Based on the results acquired from the training phase in Section IV-C1 with results presented in Table II. The proposed model has the best relevant results compared to other methods. For better statistical comparison, we perform a forecasting test to compare if the results will match the ground truth. Promising results show DeepSmart (see Figure 3e) to have higher robustness while forecasting sensory data compared to the alternatives presented in Figure 3.

	MSE	RMSE	MAE	MAPE	R^2 score
ARIMA [24]	0.41487	0.644	0.490	0.171	0.279
Auto-Arima SARIMA [25]	1.100	1.049	0.905	0.369	-0.911
SARIMAX [26]	0.732	0.856	0.596	0.196	-0.271
LSTM Univariate [27]	0.47459	0.68891	0.522	0.236	0.182
DeepSmart (LSTM)	0.23041	0.48001	0.376	0.123	0.602

Table II: Comparative Performance For forecasting

3) Anomaly Detection : Finally, we must compare the proposed model and competing methods for anomaly identification. After conducting the experiments, we determine the ranking of outliers. As indicated in Figure 4 in comparison to the ground truth as illustrated in (Total Usage) in Figure 2a. we can conclude that there are no significant differences in finding anomalies between the approaches used. The LSTM model does not offer a tight fit for this series, but it is aware of its low performance and generates wide confidence intervals. From an initial eye-test of the anomaly detection, we can conclude it is a reasonable assumption that the values detected as anomalous should be considered reasonable within. According to experimental results, the proposed model outperforms four other IoT forecasting models, demonstrating that the DeepSmart model is suitable for the task of forecasting and anomaly detection environments. The combined LSTM and ADF model eliminates insignificant variables and cycle errors. Consequently, the proposed model can accurately predict real-time time series data and identify anomalies.

As mentioned earlier, the results of the experiments suggest that the proposed DeepSmart performs well in predicting IIoT-sensory time series. Consequently, the presented model detects anomalies correctly and accurately forecasts time series data. Consequently, the proposed framework for efficient error elimination is applicable and practical in real-world applications.

V. Conclusion

This paper describes the conceptual design of the Deepsmart architecture and presents an attention mechanism for a deep learning model. DeepSmart is designed to process real-time industrial and smart IoT data series to provide accurate forecasting. This study offers a hierarchical architecture for processing IoT time series and discusses the entire Deepsmart workflow. Deepsmart outperforms other baseline approaches in time series prediction in real-world sensory dataset testing, and it has the potential to be used as an edge forecasting and anomaly detection infrastructure in our future work.

3GPP considers network slicing necessary for major characteristics of vital services, including Ubiquitous mobile ultra-broadband (*uMUB*), Mobile broadband reliable low latency communication (*MBRLLC*), Ultra-high speed with low latency communications (*uHSLLC*), Massive machine-type communication (*mMTC*), Ultra-high data density (*uHDD*), Human-centric services (*HCS*), and Multi-purpose services (*MPS*) .Considering that the forecasting model provides data on user behavior and service demand, DeepSmart's proposal can be used to promote the implementation of these services. As a result, network providers' network slices can be planned, and network resource consumption is projected to improve network planning for coverage expansion, urban computing solutions, and more.

Acknowledgment

This work is supported in part by the Natural Science Foundation of China under Grant No. 61806040 and Grant No. 61771098; in part by the fund from the Department of Science and Technology of Sichuan Province under Grant No. 2020YFQ0025 and Grant No. 2022YFG0314; in part by the Natural Science Foundation of Guangdong Province under Grant No. 2021A1515011866; in part by the Social Foundation of Zhongshan Sci-Tech Institute under Grant 420S36, and in part by the fund from Intelligent Terminal Key Laboratory of Sichuan Province under Grant No. SCITLAB-1018.

References

[1] N. Sami, T. Mufti, S. S. Sohail, J. Siddiqui, D. Kumar *et al.*, "Future Internet of Things (IOT) from Cloud perspective: Aspects, applications and challenges," *Internet of things (IoT)*, pp. 515–532, 2020.

[2] K. Yu, L. Tan, S. Mumtaz, S. Al-Rubaye, A. Al-Dulaimi, A. K. Bashir, and F. A. Khan, "Securing critical infrastructures: deep-learning-based threat detection in IIoT," *IEEE Communications Magazine*, vol. 59, no. 10, pp. 76–82, 2021.

[3] S. Vitturi, C. Zunino, and T. Sauter, "Industrial communication systems and their future challenges: Next-generation Ethernet, IIoT, and 5G," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 944–961, 2019.

[4] L. U. Khan, I. Yaqoob, M. Imran, Z. Han, and C. S. Hong, "6G wireless systems: A vision, architectural elements, and future directions," *IEEE access*, vol. 8, pp. 147 029–147 044, 2020.

[5] J. Prados-Garzon, L. Chinchilla-Romero, P. Ameigeiras, P. Muñoz, and J. M. Lopez-Soler, "Asynchronous time-sensitive networking for industrial networks," in *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. IEEE, 2021, pp. 130–135.

[6] M. Z. Chowdhury, M. Shahjalal, S. Ahmed, and Y. M. Jang, "6G wireless communication systems: Applications, requirements, technologies, challenges, and research directions," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 957–975, 2020.

[7] B. Krawczyk, J. Stefanowski, and M. Wozniak, "Data stream classification and big data analytics," *Neurocomputing*, vol. 150, no. PA, pp. 238–239, 2015.

[8] D. Bauer, D. Stock, and T. Bauernhansl, "Movement towards service-orientation and app-orientation in man-

ufacturing IT," *Procedia CIRP*, vol. 62, pp. 199–204, 2017.

[9] X. Xie, D. Wu, S. Liu, and R. Li, "IoT data analytics using deep learning," *arXiv preprint arXiv:1708.03854*, 2017.

[10] A. Gensler, J. Henze, B. Sick, and N. Raabe, "Deep Learning for solar power forecasting—An approach using AutoEncoder and LSTM Neural Networks," in *2016 IEEE international conference on systems, man, and cybernetics (SMC)*. IEEE, 2016, pp. 002 858–002 865.

[11] E. Mocanu, D. C. Mocanu, P. H. Nguyen, A. Liotta, M. E. Webber, M. Gibescu, and J. G. Slootweg, "On-line building energy optimization using deep reinforcement learning," *IEEE transactions on smart grid*, vol. 10, no. 4, pp. 3698–3708, 2018.

[12] V. C. Liang, R. T. Ma, W. S. Ng, L. Wang, M. Winslett, H. Wu, S. Ying, and Z. Zhang, "Mercury: Metro density prediction with recurrent neural network on streaming CDR data," in *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*. IEEE, 2016, pp. 1374–1377.

[13] K. Lim, Y. Hong, Y. Choi, and H. Byun, "Real-time traffic sign recognition based on a general purpose GPU and deep-learning," *PLoS one*, vol. 12, no. 3, p. e0173317, 2017.

[14] A. A. Alwan, M. A. Ciupala, A. Baravalle, and P. Falcarin, "HADES: a Hybrid Anomaly Detection System for Large-Scale Cyber-Physical Systems," in *2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC)*. IEEE, 2020, pp. 136–142.

[15] J. Gao, X. Song, Q. Wen, P. Wang, L. Sun, and H. Xu, "Robusttad: Robust time series anomaly detection via decomposition and convolutional neural networks," *arXiv preprint arXiv:2002.09545*, 2020.

[16] M.-C. Lee, J.-C. Lin, and E. G. Gran, "Salad: Self-adaptive lightweight anomaly detection for real-time recurrent time series," in *2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE, 2021, pp. 344–349.

[17] W. Wu, C. Zhou, M. Li, H. Wu, H. Zhou, N. Zhang, X. S. Shen, and W. Zhuang, "AI-native network slicing for 6G networks," *IEEE Wireless Communications*, vol. 29, no. 1, pp. 96–103, 2022.

[18] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[19] J. D. Cryer and K.-S. Chan, *Time series analysis: with applications in R*. Springer, 2008, vol. 2.

[20] N. Gutierrez and M. Wiesinger-Widi, "AUGURY: A time series based application for the analysis and forecasting of system and network performance metrics," in *2016 18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*. IEEE, 2016, pp. 351–358.

[21] A. Dokumentov and R. J. Hyndman, "STR: Seasonal-Trend Decomposition Using Regression," *INFORMS Journal on Data Science*, 2021.

[22] F. Karim, S. Majumdar, H. Darabi, and S. Harford, "Multivariate LSTM-FCNs for time series classification," *Neural Networks*, vol. 116, pp. 237–245, 2019.

[23] S. Lee, S. Iyengar, M. Feng, P. Shenoy, and S. Maji, "Deeproof: A data-driven approach for solar potential estimation using rooftop imagery," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2105–2113.

[24] A. González-Vidal, J. Cuenca-Jara, and A. F. Skarmeta, "IoT for water management: Towards intelligent anomaly detection," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. IEEE, 2019, pp. 858–863.

[25] A. Choudhary, S. Kumar, M. Sharma, and K. P. Sharma, "A Framework for Data Prediction and Forecasting in WSN with Auto ARIMA," *Wireless Personal Communications*, vol. 123, no. 3, pp. 2245–2259, 2022.

[26] J. Dutta and S. Roy, "IndoorSense: context based indoor pollutant prediction using SARIMAX model," *Multimedia Tools and Applications*, vol. 80, no. 13, pp. 19 989–20 018, 2021.

[27] A. Essien and C. Giannetti, "A deep learning framework for univariate time series prediction using convolutional LSTM stacked autoencoders," in *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*. IEEE, 2019, pp. 1–6.