

Research Article

Comparative Analysis of Time Series Prediction Algorithms on Multiple Network Function Data of NWDAF

Dasheng Chen , Qi Song, Yinbin Zhang, Ling Li, and Zhiming Yang

Research Institute of China Telecom Corporation Limited, Guangzhou 510660, Guangdong, China

Correspondence should be addressed to Dasheng Chen; chends4@foxmail.com

Received 25 August 2023; Revised 18 November 2023; Accepted 29 December 2023; Published 13 January 2024

Academic Editor: Janez Perš

Copyright © 2024 Dasheng Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the emergence and vigorous development of 5G technology, there is a significant surge in network usage and traffic, resulting in heightened complexity within network and IT environments. This exponential increase in activity produces a plethora of events, making conventional systems inadequate for the efficient management of 5G networks. In comparison to 4G technology, 5G technology brings forth a host of new features, one of which is the network data analytics function (NWDAF). This function grants network operators the flexibility to either employ their own data analytics methodologies based on machine learning (ML) and deep learning (DL) into their networks. In this paper, we present a dataset named “NWDAF-NFPP” for network function performance time series prediction, collected from a laboratory at China Telecom. The dataset is carefully anonymized to ensure maximum realism and comprehensiveness, while safeguarding sensitive information. It encompasses eight categories of network functions, with data collected at five-minute intervals. The availability of this dataset provides valuable resources for researchers to conduct time series prediction research on network element performance. Following data collection, a total of six models were employed for network element performance prediction, encompassing both machine learning and deep learning approaches. This diverse set of models was carefully chosen to ensure comprehensive coverage of different techniques and algorithms. Through the comparison and analysis of these models, we aim to evaluate their predictive capabilities and identify the most effective approach for network element performance prediction. This comparative analysis will provide valuable insights into the strengths and limitations of each model, aiding in informed decision-making for network optimization and management strategies in the future.

1. Introduction

The rapid development of 5G networks presents challenges and issues to network analytics. The high speed and low latency characteristics of 5G networks contribute to increased network traffic and complexity, placing greater pressure on network analytics [1]. The processing and analysis of large-scale real-time data streams necessitate more powerful computing and storage capabilities, as well as more efficient algorithms and models [2]. Furthermore, the flexibility and customization options of 5G networks may result in more complex network architectures and topologies. This complexity adds to the challenges faced in network analytics,

as it requires consideration of interactions among multiple network slices, services, and applications.

NWDAF (network data analytics function) was introduced in the context of 5G networks as a result of the evolving demands and challenges in network analytics. 5G networks, with their high data speeds, low latency, and enhanced connectivity, presented a need for advanced analytics capabilities to effectively manage and optimize these complex networks. The concept of NWDAF was conceptualized and proposed by the 3rd Generation Partnership Project (3GPP), a global standardization organization responsible for developing mobile communication technologies [3]. The NWDAF operates as a network function (NF) within the 5G core network,

utilizing machine learning and artificial intelligence algorithms to analyze data pertaining to past, present, and future states [4]. Its primary function is to provide analytical support to various 5G control plane NFs, such as the policy control function (PCF) responsible for traffic policy management and the network slice selection function (NSSF) tasked with instantiating and selecting network slices based on data inputs [5]. This integration of machine learning and AI capabilities empowers the NWDAF to effectively drive data analytics operations and enhance the overall efficiency and performance of the 5G Core network.

The network function performance prediction is a key functionality within the NWDAF. It collects data from various network functions and applies advanced analytics techniques to predict the future performance of network functions [6]. By analyzing historical performance data, NWDAF identifies patterns and trends in the performance time series of network functions. It then leverages machine learning algorithms and predictive models to forecast the future performance of these elements. This functionality enables network operators to make informed decisions regarding resource allocation, capacity planning, and network optimization [7]. Network operators can proactively identify potential congestion, troubleshoot performance issues, and optimize their network infrastructure accordingly. This ultimately leads to improved network reliability, enhanced user experience, and better overall network management [8].

There is a significant scarcity of time series datasets available for predicting the performance of network functions (NFs), with even fewer datasets originating from real-world network environments. The availability of the dataset proposed in this paper offers researchers a valuable resource for conducting research in network element performance prediction. By utilizing this dataset, researchers can develop and evaluate predictive models to enhance the understanding and optimization of network operations. The careful collection and preprocessing of this dataset ensure its reliability and suitability for academic research purposes. Its availability opens up exciting possibilities for further advancements in the field of network analytics and optimization.

The dataset utilized in this study is obtained from a laboratory at China Telecom, offering an exceptionally accurate representation of real-world data.

In this study, we used six models for experiments, followed by comparative analysis. Our main goal is to provide a relatively comprehensive benchmark method for predicting the performance of various NFs to the relevant industry and provide more ideas for model differentiation comparison to maximize prediction rate, while considering processing time, which is a very important factor in real-time applications. The rest of the paper is organized as follows: Section 2 introduces the relevant knowledge of NWDAF and research work on NF performance time series prediction. In Section 3, we present the dataset employed in this study, elaborate on the methodology employed for feature processing, and describe various proposed models utilized during the experiment. Section 4 presents and

analyzes the obtained results. Section 5 provides conclusions and prospects.

2. Related Works

Intelligent cellular networks based on state-of-the-art AI/ML technologies have been extensively researched in the past decade. Casellas et al. highlight the importance of integrating AI/ML techniques for the control, management, and orchestration of various components within 5G networks. However, they do not specifically mention the role of NWDAF in this context [9].

The NWDAF, which is the network data analytics function, has been introduced in the 5G core network to facilitate data analytics and machine learning model training. It is anticipated that NWDAF will assume a critical role and serve as an indispensable functional entity in the forthcoming AI-native 6G wireless network [10]. Quality of experience (QoE) refers to the level of user satisfaction and dissatisfaction with an application or service. QoE management involves three steps: modeling, monitoring, and controlling [11]. Predicting network function performance is part of the monitoring phase, and the subsequent control phase makes decisions based on the predicted results obtained here. Kao et al. propose a native QoE sustainability architecture for 5G and B5G networks. The architecture utilizes standard interfaces of NWDAF to facilitate the exchange of analytical data between a 5G system (5GS) and application domains. It integrates QoE predictors to effectively conduct the aforementioned QoE management procedures.

Mhedhbi et al. propose the utilization of importance sampling techniques and a modified detection threshold, known as the M-KNN scheme, to enhance prediction performance [12]. In the paper [4], three ML models, namely, linear regression, long-short-term memory, and recursive neural networks, are applied to investigate the estimation of behavior information and network load prediction capabilities of NWDAF. To minimize the mean absolute error in network load prediction, three different models are utilized by comparing the model prediction value with the actual generated data. It is worth noting that the dataset used in the study was synthetically generated based on the 3GPP specifications.

Given the expected proliferation of connected devices in 5G systems, centralizing all data for analytics purposes is deemed inefficient. Hernández-Chulde et al. propose a distributed architecture to perform network analytics by applying ML techniques in the context of network operation and control of 5G networks [13]. The proposed distributed analytics architecture involves a centralized NWDAF instance and multiple distributed instances colocated with other NFs, solely collecting data from those colocated NFs. Díaz González et al. [14] effectively enhanced local performance with minimal additional costs by selectively aggregating updates from other components in the global model. This approach was further validated through their application of LSTM for time series prediction.

Li et al. [15] propose a model transfer framework based on intracluster federated learning. This framework enables the transfer of models by facilitating information exchange among network elements. The experimental outcomes validate the effectiveness of this framework in enhancing the efficiency of traffic prediction.

3. Materials and Methods

In this section, we provide a description of the time series dataset utilized in our study for training various predictive models. Furthermore, we present an overview of several well-established time series predictors and classical performance indicators commonly used for evaluating time-series predictions. The purpose of this paper is to conduct a comprehensive experimental comparison of these predictive factors, aiming to provide guidance for researchers in the field when selecting models.

3.1. Datasets. This paper introduces a dataset named “NWDAF-NFPP” (network function performance prediction), which consists of 6 columns: “beginTime,” “counterId,” “elementIp,” “elementType,” “measObjLdn,” and “measResult”. It was collected from a laboratory at Research Institute of China Telecom Corporation Limited [16]. The dataset has been meticulously sorted based on the “beginTime” column, encompassing a time span from June 23, 2023, 00:00 AM to June 27, 2023, 11:45 AM. We divided our dataset into a training set and a test set using the timestamp of June 27, 2023, at 00:00 (midnight) as the cutoff point. The data collected before this timestamp was used for training, while the data collected from this timestamp onwards was designated for testing. The quantity information is shown in Table 1. Notably, the “measResult” column assumes the role of target values for our future time series prediction analysis.

Table 2 illustrates the presence of four nonnumeric features within the NWDAF-NFPP dataset in our study. The term “counterId” refers to the category ID of the predicted metric, which corresponds to the category of the target prediction column “measResult.” In the dataset, the performance metric column “measResult” corresponds to specific performance indicators for different network function (NF) elements. For the AMF NF, “measResult” represents the count of initial registration requests. For the EDS NF, it denotes the count of ENUM local query requests. The I-CSCF and LDRA NFs have their “measResult” column associated with the total count of received messages. Likewise, for the PSBC NF, “measResult” indicates the count of IMS registration requests. The S-CSCF NF’s “measResult” column reflects the total count of sent messages. In the case of the SMF NF, “measResult” is the count of UE-initiated PDU session establishment requests. Finally, for the UDM NF, “measResult” represents the count of AUSF authentication service requests. However, it is important to note that the specific meanings of these performance indicators do not have any influence on the subsequent experimental procedures. In the experiment, we treat all the mentioned metrics as numerical values uniformly. The “elementIp” indicates the IP address of the network element, while the

TABLE 1: Breakdown of data records in the NWDAF-NFPP.

Dataset	Total
Train	16217
Test	1704

TABLE 2: Features of NWDAF-NFPP dataset.

No.	Feature	Type	No.	Feature	Type
1	counterId	Nonnumeric	3	elementType	Nonnumeric
2	elementIp	Nonnumeric	4	measObjLdn	Nonnumeric

“elementType” denotes the specific type of network element. Within the dataset, eight types of network functions are present: “PSBC,” “S-CSCF,” “EDS,” “LDRA,” “I-CSCF,” “UDM,” “SMF,” and “AMF” [17]. Lastly, the “measObjLdn” refers to the link code that establishes connectivity between network functions.

In our dataset, we have identified eight types of network functions, namely, PSBC, S-CSCF, EDS, LDRA, I-CSCF, UDM, SMF, and AMF. Let us take a closer look at each of these network functions and their respective roles within the NWDAF (network data analytics function) ecosystem [18]:

- (1) *PSBC*. Signify a specific functionality resulting from the amalgamation of P-CSCF and SBC within a network setup. P-CSCF, integral to the IP multimedia subsystem (IMS), manages signaling traffic between user devices and the network, overseeing call session establishment, termination, and data forwarding. SBC, situated at the network edge, provides functions for VoIP communication management, including security features, call routing, media conversion, and traffic adjustment
- (2) *S-CSCF (Serving-Call Session Control Function)*. It acts as a user-facing session control function in the IMS architecture, responsible for managing and controlling user sessions for VoIP and multimedia services. It interacts directly with end users, handling authentication, authorization, and service session establishment
- (3) *EDS (ENUM/DNS)*. Comprising two logical network elements, ENUM and DNS, it serves as a universal addressing system within the IMS domain for route resolution across the entire network, without engaging in routing or forwarding functionalities
- (4) *LDRA (Low-Level Data Router and Authentication Server)*. Denotes a data router and authentication server strategically deployed at a provincial level within the network architecture. This nomenclature implies its positioning within a lower administrative or geographical tier in contrast to HDRA, which

signifies a data router operating across broader regional boundaries

- (5) *I-CSCF (Interrogating-Call Session Control Function)*. It serves as an edge node in the IMS architecture, positioned at the boundary of the IMS core network, responsible for receiving user data from external networks and routing it to the appropriate S-CSCF. In the context of IMS architecture, the S-CSCF and I-CSCF play distinct roles
- (6) *UDM (User Data Management)*. It handles user data storage and management, encompassing identity, policies, and session states. UDM ensures user authentication, authorization, and configuration in the 5G core network. In addition to data storage, UDM communicates with other components within the 5G core network (such as AMF and SMF) to ensure access and utilization of necessary user data across network elements
- (7) *SMF (Session Management Function)*. It oversees and controls data sessions within the 5G network, managing data transmission, quality of service (QoS), and security. Its primary objective revolves around ensuring efficient and secure communication between mobile devices while executing various session management tasks
- (8) *AMF (Access and Mobility Management Function)*. It oversees the access and registration processes for user devices. It manages user authentication, device tracking, and access control for mobile users interacting with the network. AMF is also responsible for facilitating smooth transitions during initial access and handovers between different network services for mobile users

3.2. Evaluation Metrics. In general, within the context of time series prediction, several commonly employed evaluation metrics can be found. These metrics serve as valuable indicators for assessing the performance of predictive models and are widely recognized in the scientific community:

3.2.1. MAE. Mean absolute error is a statistical metric used in machine learning to measure the average magnitude of errors between predicted and actual values. It calculates the absolute difference between each predicted value and its corresponding actual value and then takes the average of these differences. A lower MAE indicates better accuracy and a closer fit between predicted and actual values. For ground truth time series y and predicted time series y_t of length T , it is computed as

$$\text{MAE} = \frac{1}{T} \sum_{t=1}^T (|y - y_t|). \quad (1)$$

3.2.2. RMSE. Root mean squared error measures the average magnitude of errors between predicted and actual values. It considers both the direction and magnitude of errors. By

TABLE 3: Statistical overview of network functions.

No.	Element type	Train	Test	No.	Element type	Train	Test
1	PSBC	1391	142	5	I-CSCF	2782	284
2	S-CSCF	2782	284	6	UDM	3948	426
3	EDS	1391	142	7	SMF	1316	142
4	LDRA	1291	142	8	AMF	1316	142

taking the square root of the average of squared differences, RMSE provides a more sensitive measure of error compared to MAE. Lower RMSE values indicate better accuracy and a closer fit between predictions and actual values. For ground truth time series y and predicted time series y_t of length T , it is computed as

$$\text{RMSE} = \sqrt{\frac{1}{T} \sum_{t=1}^T (y - y_t)^2}. \quad (2)$$

3.2.3. RMSPE. Symmetric mean absolute percentage error is a statistical metric used to measure the accuracy of forecasts. It calculates the average percentage difference between predicted and actual values, giving equal weight to overestimations and underestimations. It provides a symmetric view of errors, providing a balanced evaluation of prediction accuracy. The lower the SMAPE value, the better the accuracy of the forecast. For ground truth time series y and predicted time series y_t of length T , it is computed as

$$\text{RMSPE} = 100^* \sqrt{\frac{1}{T} \sum_{t=1}^T \frac{(y - y_t)^2}{y_t}}. \quad (3)$$

Due to the large fluctuation and irregularity in the experimental dataset, along with the presence of zero values, this study has opted to utilize root mean square percentage error (RMSPE) as the evaluation metric for assessing the performance of models.

3.3. Experimental Method

3.3.1. Cross-Validation. In this study, a five-fold cross-validation approach was employed to assess the predictive performance of the model. The methodology for this verification technique is as follows: Firstly, the entire dataset was randomly partitioned into five nonoverlapping subsets. Then, in each iteration, one subset was chosen at random as the validation set to evaluate the model, while the remaining four subsets were used as the training data to train the prediction model. This process was repeated five times, ensuring that each subset served as the validation set exactly once. Finally, the results obtained from the five iterations were averaged to derive the final prediction outcomes.

Cross-validation is a widely utilized method for evaluating machine learning models, particularly in scenarios where the dataset is limited in size. Although less commonly employed in deep learning, due to the relatively large computational expenses, it can still be successfully applied when

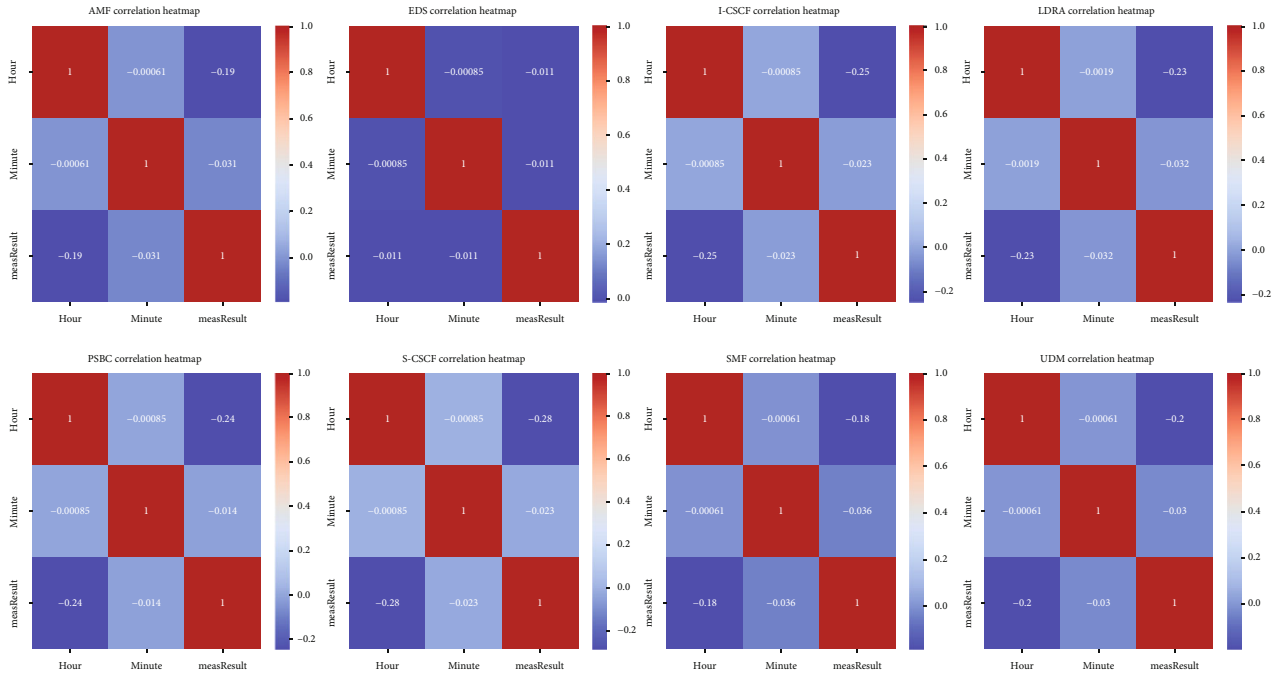


FIGURE 1: Correlation matrix heatmap: visualizing relationships among network functions.

the available data is relatively small. Considering the constraints of the deep learning training process, the adoption of a five-fold cross-validation scheme allows for a robust assessment of the model’s performance in this study.

3.3.2. Independent Testing. Compared with cross-validation, independent testing is time-consuming and logically simple. First, the algorithm is trained on the training set. Secondly, the parameters of the model are adjusted by observing the performance of the model according to the evaluation indicators each time. At the same time, independent testing is also a method to test the effect of the model. Generally, independent test sets are used to verify the effect of the model in the end of experiments. The specific way is to use independent test sets as common data to compare with other methods to be compared. The above two experimental methods have been applied in this study. Generally, cross-validation and independent testing experiments at the same time will make the experimental results more convincing.

3.3.3. The Proposed Predictive Frameworks. In this study, we aim to conduct a comprehensive comparative analysis of multiple time series forecasting models on the given dataset. Our goal is to evaluate the performance of these models across various dimensions. By utilizing a diverse set of forecasting techniques, we can gain valuable insights into their effectiveness in capturing different patterns and trends in the data. To compare the models, we will primarily utilize the root mean squared percentage error (RMSPE) metric, which quantifies the prediction errors. RMSPE calculates the percentage difference between the predicted and actual values and then takes the square root of the mean of the squared differences. This metric is commonly used in the evaluation of regression models and provides a measure of

the relative accuracy of predictions while accounting for the scale of the target variable.

By employing the RMSPE metric, we can effectively assess the performance of the models in terms of their ability to accurately predict the target variable. Lower values of RMSPE indicate better predictive performance, as it signifies smaller prediction errors relative to the actual values. This comparative analysis allows us to identify the model that exhibits superior predictive capabilities in our study.

3.4. Time Series Prediction Algorithm

3.4.1. ARIMA. The ARIMA model [19], short for autoregressive integrated moving average model, is a widely used time series forecasting technique in various research fields. It requires three crucial parameters to be specified:

- (1) *Autoregressive Order (p)*. This parameter represents the number of lagged observations included in the model. It captures the linear relationship between the current observation and its historical values
- (2) *Integrated Order (d)*. The integrated order parameter refers to the degree of differencing performed on the time series data. It is used to stabilize the series and make it stationary by eliminating trends and seasonality
- (3) *Moving Average Order (q)*. The moving average order parameter denotes the number of lagged forecast errors that are considered in the model. It captures the short-term dependencies between observations

These three parameters, namely, the autoregressive order (p), the integrated order (d), and the moving average order

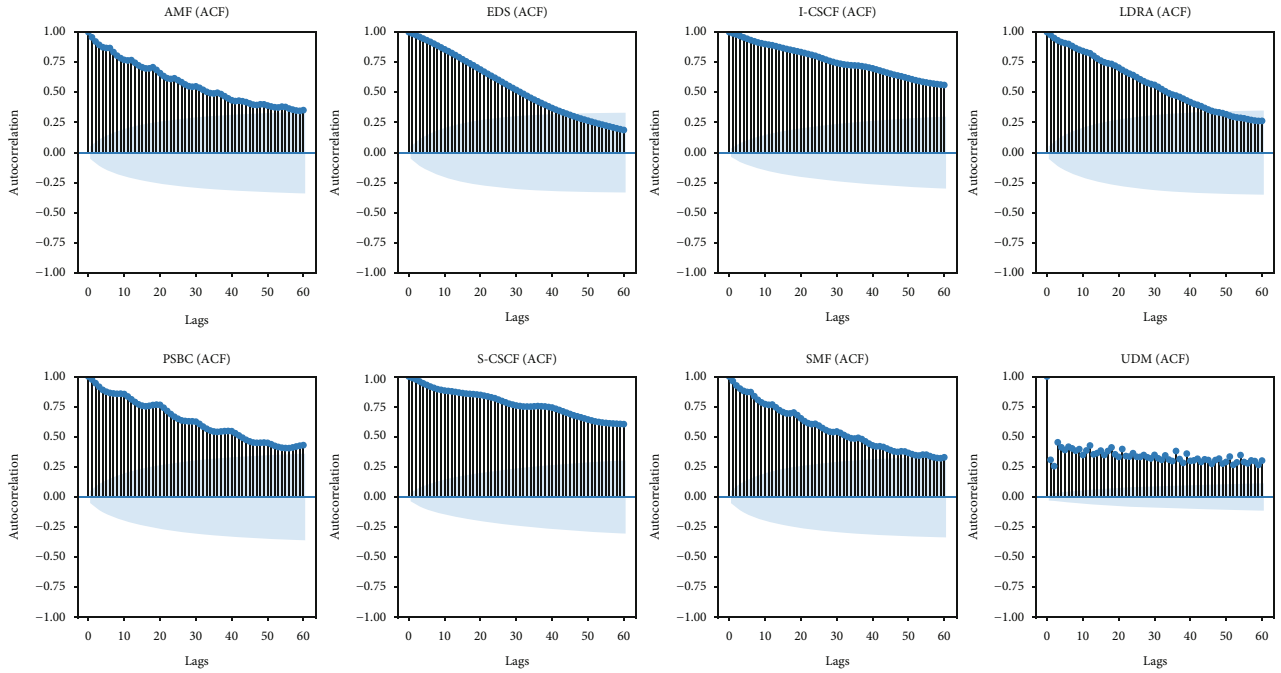


FIGURE 2: Autocorrelation function (ACF) among network functions.

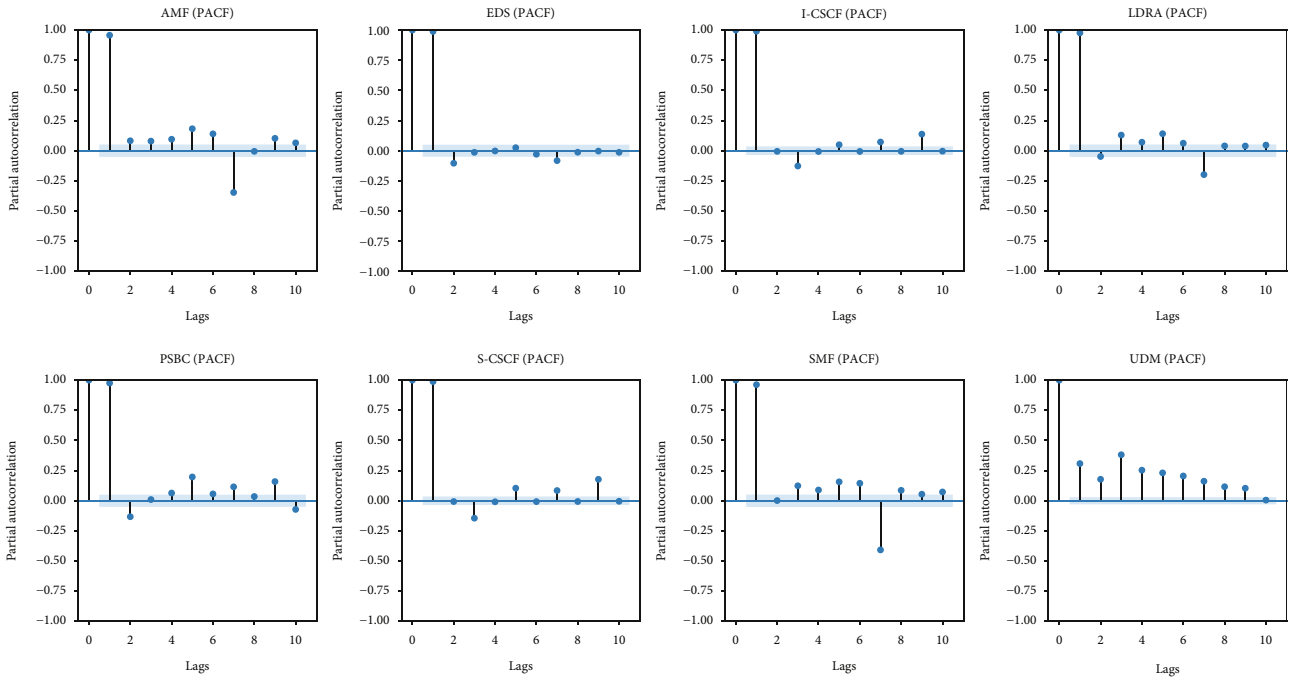


FIGURE 3: Partial autocorrelation function (PACF) among network functions.

(q), play a crucial role in determining the effectiveness of the ARIMA model in capturing the underlying patterns and making accurate forecasts in time series analysis.

By carefully selecting appropriate values for these parameters, researchers can ensure the model’s ability to effectively capture and explain the dynamics of the data, ultimately contributing to the advancement of scientific knowl-

edge and understanding in their respective fields. However, the ARIMA model assumes linearity and stationarity in the data, which may limit its effectiveness for complex and non-linear time series patterns.

3.4.2. Random Forest. The random forest model [20] offers a powerful and versatile machine learning approach.

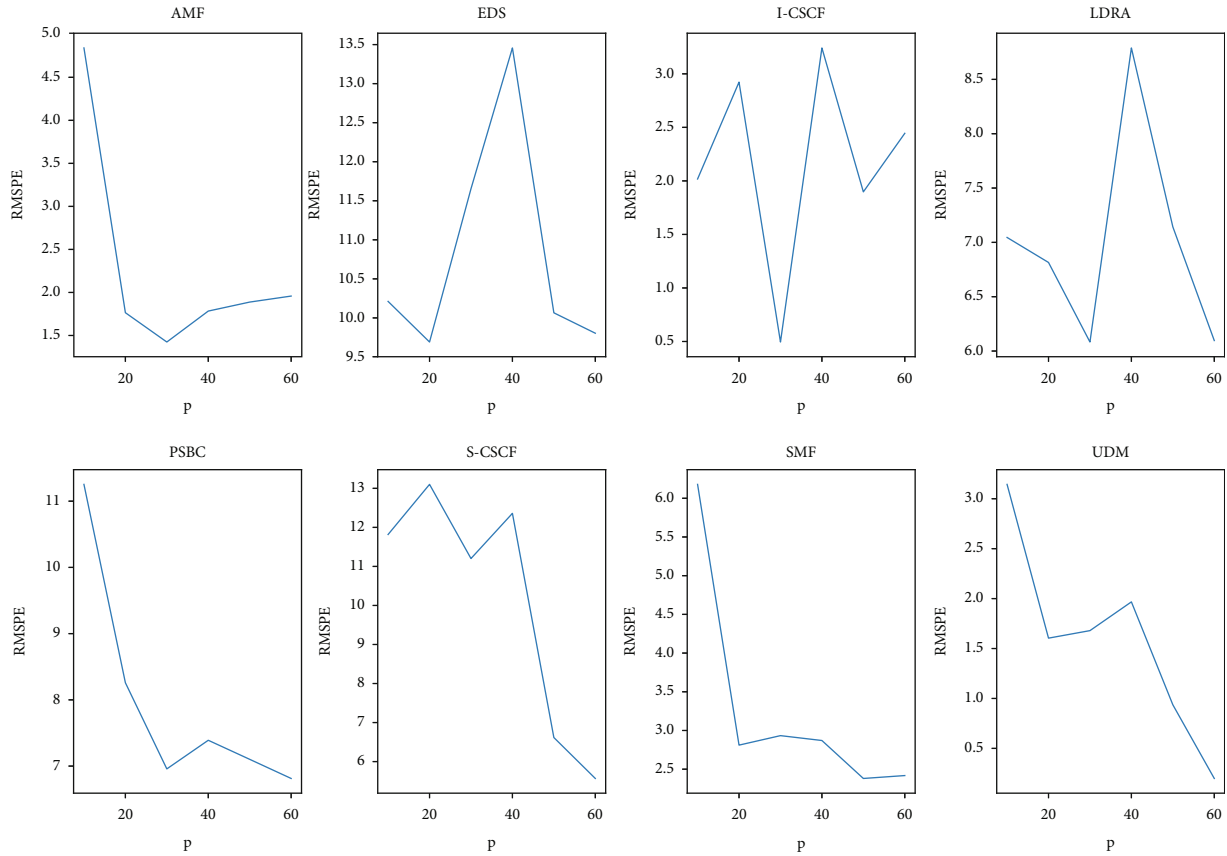
FIGURE 4: RMSPE indicator curves of different p values of ARIMA.

TABLE 4: Confirmed parameter of ARIMA predictor.

No.	Element type	Parameter (p, d, q)	No.	Element type	Parameter (p, d, q)
1	PSBC	$(60, 0, 2)$	5	I-CSCF	$(30, 0, 2)$
2	S-CSCF	$(60, 0, 2)$	6	UDM	$(60, 0, 2)$
3	EDS	$(20, 0, 2)$	7	SMF	$(50, 0, 2)$
4	LDRA	$(60, 0, 2)$	8	AMF	$(30, 0, 2)$

However, it is important to note that random forest is not inherently designed for time series data. Nevertheless, there are adaptations and extensions of random forest specifically tailored for time series prediction tasks.

One important parameter in a time series forecasting model based on random forest is the “MaxLags” parameter. It determines the maximum number of lagged observations that are included as input features in the model. Lagged observations refer to past values of the target variable that are used as predictors for future values.

Choosing an appropriate value for the MaxLags parameter is crucial as it directly influences the model’s ability to capture temporal dependencies and patterns. Too few lags might result in underutilizing valuable historical information, leading to suboptimal predictions. Conversely, including too many lags may introduce noise and overfitting, compromising the model’s accuracy on unseen data.

3.4.3. ExtraTrees. The ExtraTrees model [21] is another powerful machine learning algorithm. Similar to random forest, ExtraTrees is a type of ensemble learning method that combines multiple decision trees to make predictions. However, unlike random forest, ExtraTrees adopts a more random approach in building individual decision trees, leading to increased diversity and reduced bias. Consistency with the random forest model is observed in many of its parameters.

3.4.4. LGBM. The LGBM (light gradient boosting machine) model [22] stands out as a powerful machine learning algorithm. LGBM belongs to the category of gradient boosting methods, which iteratively builds an ensemble of weak prediction models, typically decision trees, to form a strong predictive model.

3.4.5. DeepAR. The DeepAR model [23] is a deep learning approach proposed by Amazon for time series forecasting.

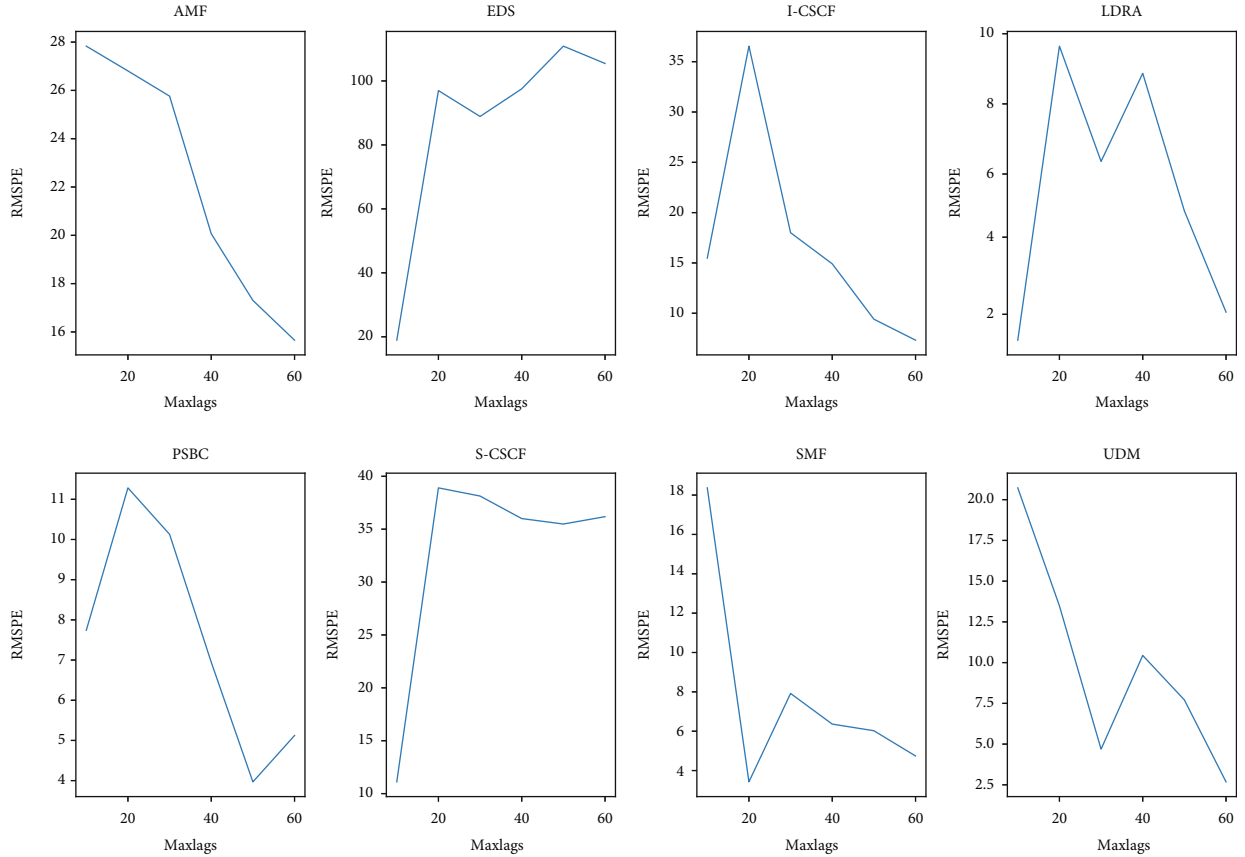


FIGURE 5: RMSPE indicator curves of different MaxLags values of RF.

It combines the power of recursive neural networks (RNNs) and long short-term memory (LSTM) structures to handle complex time series data with seasonality, trends, and periodicity. By leveraging historical data patterns and contextual information, the DeepAR model generates probabilistic forecasts for future time points. It utilizes various parameters to optimize its performance in time series forecasting. One of these parameters is N_PAST , which plays a crucial role in determining the historical context considered for prediction. N_PAST represents the number of past time steps that the model takes into account when generating forecasts. By adjusting N_PAST , the model can capture different levels of historical information, influencing the accuracy and complexity of the predictions.

3.4.6. Autoformer. Autoformer model [24] is a novel predictor for long-term time series forecasting. It maintains the residual and encoder-decoder structure but introduces a decomposition forecasting architecture. By incorporating decomposition blocks as internal operators, autoformer effectively separates long-term trend information from predicted hidden variables. This design allows for progressive decomposition and refinement of intermediate results during the forecasting process. Inspired by stochastic process theory, it replaces self-attention with an autocorrelation mechanism that identifies subseries similarity based on series periodicity and aggregates similar subseries from underlying periods. Theoretically, this predictor is better

TABLE 5: Confirmed MaxLags of RF predictor.

No.	Element type	MaxLags	No.	Element type	MaxLags
1	PSBC	50	5	I-CSCF	60
2	S-CSCF	50	6	UDM	60
3	EDS	10	7	SMF	20
4	LDRA	10	8	AMF	60

suites for long-term data and may not be suitable for the data used in this study.

3.4.7. One-Hot Feature Representation. One hot coding is commonly employed in numerous experimental studies to extend the values of discrete data features into a European space, where each discrete feature value corresponds to a distinct point in this space. By encoding discrete features using the one-hot method, it becomes easier to calculate distances between features, which not only enhances the interpretability but also ensures a more reasonable measure of feature dissimilarity. The one-hot feature extraction technique, as highlighted in the works of Rodríguez et al. [25], finds extensive applications in various fields such as sequence recognition, natural language processing (NLP), and related domains. In the majority of cases, this approach yields outstanding experimental outcomes. As mentioned above, the NWDAF-NFPP

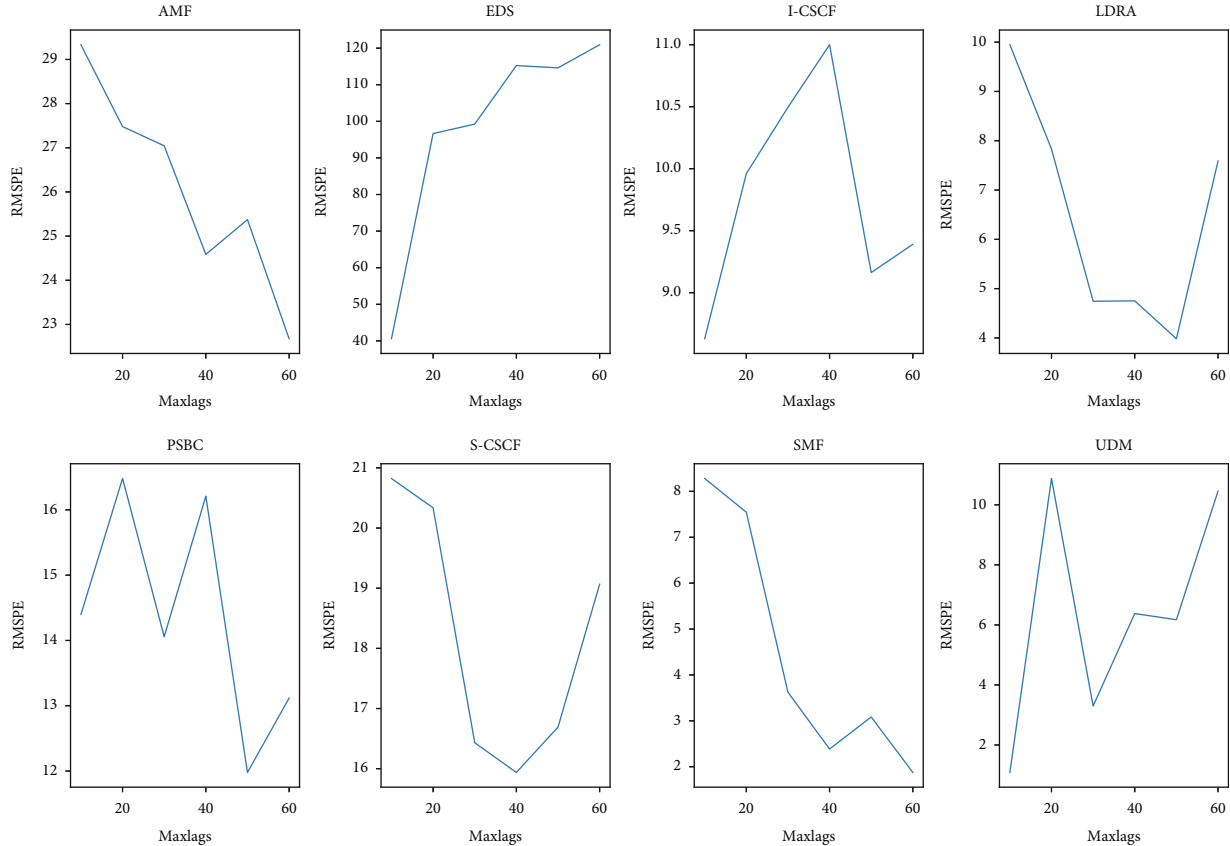


FIGURE 6: RMSPE indicator curves of different MaxLags values of ET.

dataset has 4 nonnumeric features. We need to convert all nonnumeric features into digital representation.

4. Results and Discussion

The column “measResult” in the dataset represents the performance metric values that serve as the prediction targets for the model. However, we have identified certain instances where these values are given as zeros, which deviates from the actual observations. In order to rectify this discrepancy, we have employed linear interpolation to fill in the missing data. This decision was made based on the expertise of domain specialists to ensure accurate representation of the dataset.

However, this leads to a problem of excessive data volatility, which is abnormal. Therefore, we addressed this issue through code implementation by constraining some outliers within a range of 2 standard deviations. Any data points exceeding this range were set to the upper limit, while data points falling below this range were set to the lower limit within the specified range. This approach helps to ensure that the data remains within a reasonable range and mitigates the impact of extreme values.

Regarding the dataset, it covers the time period from June 23, 2023, 00:00 to June 27, 2023, 11:45. The dataset was categorized into eight network element types for more effective analysis and experimentation. To create the training and test sets, we utilized June 27, 2023, 00:00 as the cut-off

TABLE 6: Confirmed MaxLags of ET predictor.

No.	Element type	MaxLags	No.	Element type	MaxLags
1	PSBC	50	5	I-CSCF	10
2	S-CSCF	40	6	UDM	10
3	EDS	10	7	SMF	60
4	LDRA	50	8	AMF	60

point. Specifically, the training set consists of data before June 27, 2023, 00:00, while the test set includes data from June 27, 2023, 00:00 to June 27, 2023, 11:45. The categorization of the dataset based on network element types is presented in Table 3.

4.1. Feature Engineering of Network Functions. Based on relevant domain knowledge and considering the actual data, we extract the hour and minute from the timestamp as new feature columns to be included in the model training. Next, we will assess the necessity of these features by performing correlation analysis.

Subsequently, an individual analysis was conducted on eight categories of network element data. The experimental results, as illustrated in Figure 1, were represented using correlation matrix heatmaps. We considered a correlation coefficient value of approximately 0.2 or higher as an indication

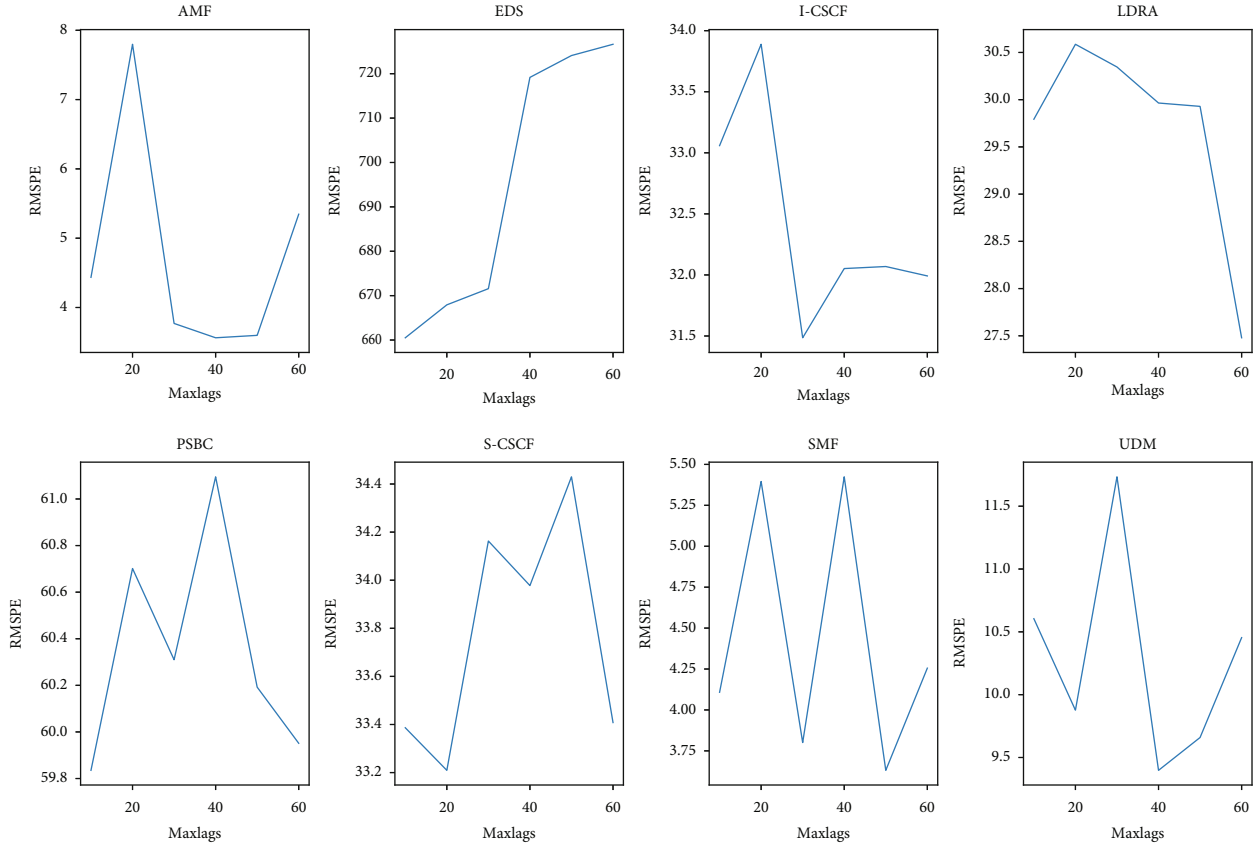


FIGURE 7: RMSPE indicator curves of different MaxLags values of LGBM.

of correlation. From the graph, it is evident that the majority of network element data exhibit a significant correlation between their hourly features and the prediction target, with values centered around 0.2. Therefore, it can be inferred that the hourly features in the dataset exhibit a significant correlation with the prediction target. In the subsequent experiments, we incorporated the hour features into the training process for these elements.

4.2. Time Series Predictor Optimization. The preliminary determination of parameter ranges for the ARIMA model can be facilitated by analyzing the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots. These plots provide insights into the correlation structure of the time series data, aiding in the identification of suitable parameter values.

The ACF plot illustrates the decay of the autocorrelation coefficients as the lag increases. Points on the plot that rapidly decline and become close to zero indicate the presence of truncation. Truncation points suggest the potential order of the autoregressive (AR) component (p) in the ARIMA model.

Similarly, the PACF plot displays the decay of the partial autocorrelation coefficients. Points that rapidly decrease and approach zero signify truncation. Truncation points in the PACF plot can guide the determination of the moving average (MA) component (q) in the ARIMA model.

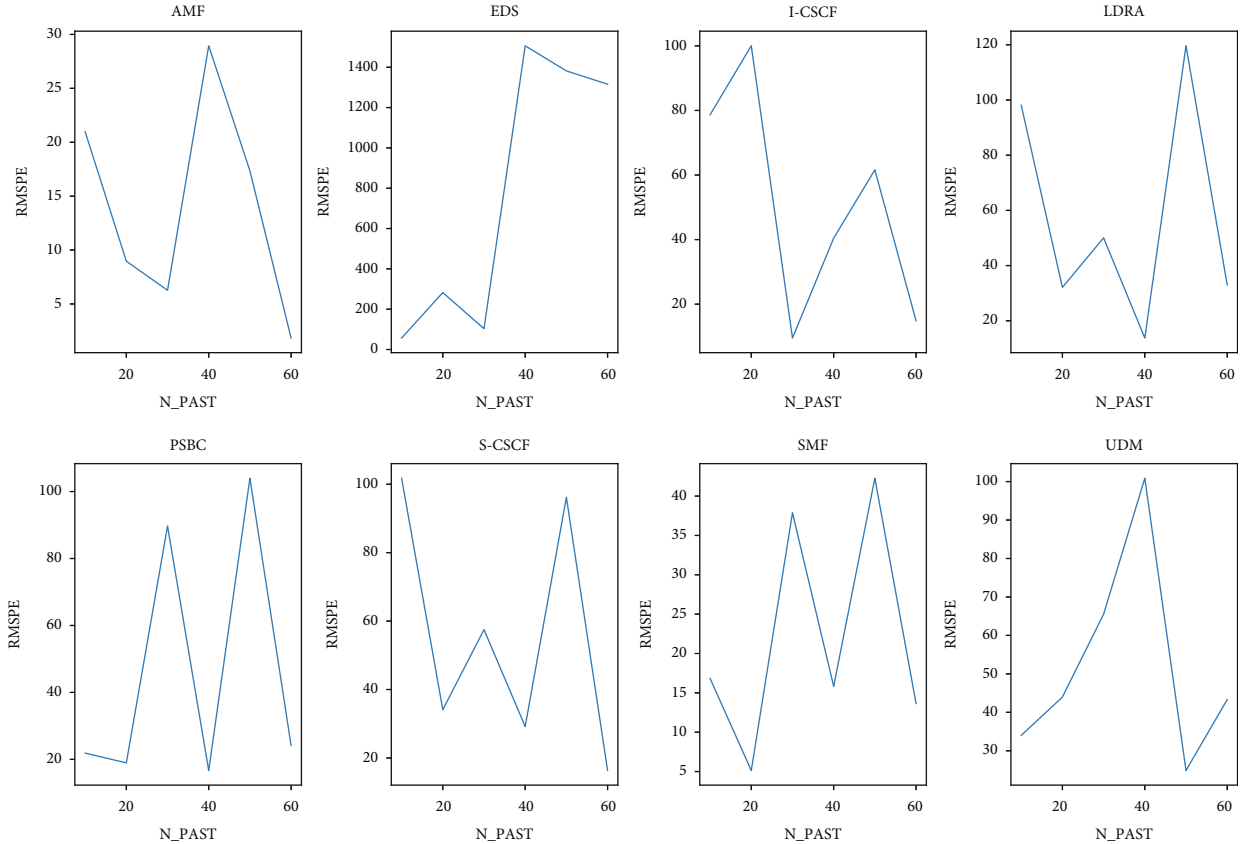
TABLE 7: Confirmed MaxLags of LGBM predictor.

No.	Element type	MaxLags	No.	Element type	MaxLags
1	PSBC	10	5	I-CSCF	30
2	S-CSCF	20	6	UDM	40
3	EDS	10	7	SMF	50
4	LDRA	60	8	AMF	40

However, it is important to note that these truncation points serve as initial guidelines and should be further evaluated and refined using additional methods. Techniques such as grid search with model evaluation metrics (RMSE and MAPE) and domain knowledge can be employed to finalize the selection of appropriate values for p and q .

By incorporating these analyses, researchers can obtain a preliminary range of p and q values, providing a foundation for subsequent model optimization and forecasting.

As illustrated in Figure 2, the ACF plot with eight subplots, each subplot exhibits different characteristics. Some of them have lag values reaching 60 without a significant decreasing trend. However, excessively high lag values can lead to overfitting, which is not desirable. For most of the plots, the autocorrelation coefficients intersect with the confidence interval around lag 50. Therefore, we intend to use lag 50 as the initial value for the parameter “ p ” in the ARIMA model.

FIGURE 8: RMSPE indicator curves of different N_{PAST} of DeepAR.

It is important to note that selecting an appropriate lag value involves considering the balance between capturing autocorrelation patterns and avoiding overfitting. By choosing a lag value where the autocorrelation coefficients intersect with the confidence interval, we ensure that the estimated coefficients are not significantly different from zero. The specific lag value of 50 is based on this consideration.

As illustrated in Figure 3, the PACF plot displays eight subplots, each exhibiting slight differences. However, in each subplot, there is a significant decline observed around lag 2. Based on this observation, we propose selecting 2 as the initial value for the “ q ” parameter in the ARIMA model. It insights into the direct relationship between an observation and its lagged values while controlling for the effects of intervening observations. Analyzing the PACF plot allows us to identify significant lags where the autocorrelation drops noticeably. By choosing lag 2 as the initial value for the “ q ” parameter, we take into account the significant decline observed around this lag in all of the PACF subplots. This choice ensures that the ARIMA model captures important autocorrelation patterns while maintaining parsimony.

It is essential to consider these preliminary findings as a part of the broader methodology for time series analysis and modeling. Further validation and evaluation of the selected ARIMA model should be performed using appropriate diagnostics and model selection criteria. Based on observations and experience, the d parameter in the ARIMA model is

TABLE 8: Confirmed N_{PAST} of DeepAR predictor.

No.	Element type	N_{PAST}	No.	Element type	N_{PAST}
1	PSBC	40	5	I-CSCF	30
2	S-CSCF	60	6	UDM	50
3	EDS	10	7	SMF	20
4	LDRA	40	8	AMF	60

set to 0. This choice is made considering the data’s characteristics and assuming no differencing is needed for stationarity.

In accordance with the conducted experiment, the q parameter of the ARIMA model was fixed at 2, and a further investigation was carried out to explore the impact of different p values. The parameter search ranged from 10 to 60 with a step size of 10. By referring to Figure 4, the minimum values for the RMSPE indicators of various network functions were identified: $q = 30$ for the AMF element, $q = 20$ for the EDS element, $q = 30$ for the I-CSCF element, $q = 60$ for the LDRA element, $q = 60$ for the PSBC element, $q = 60$ for the S-CSCF element, $q = 50$ for the SMF element, and $q = 60$ for the UDM element. These critical data points have been determined and will be used as the basis for forthcoming comparative experiments. The final confirmed parameter values are documented in Table 4.

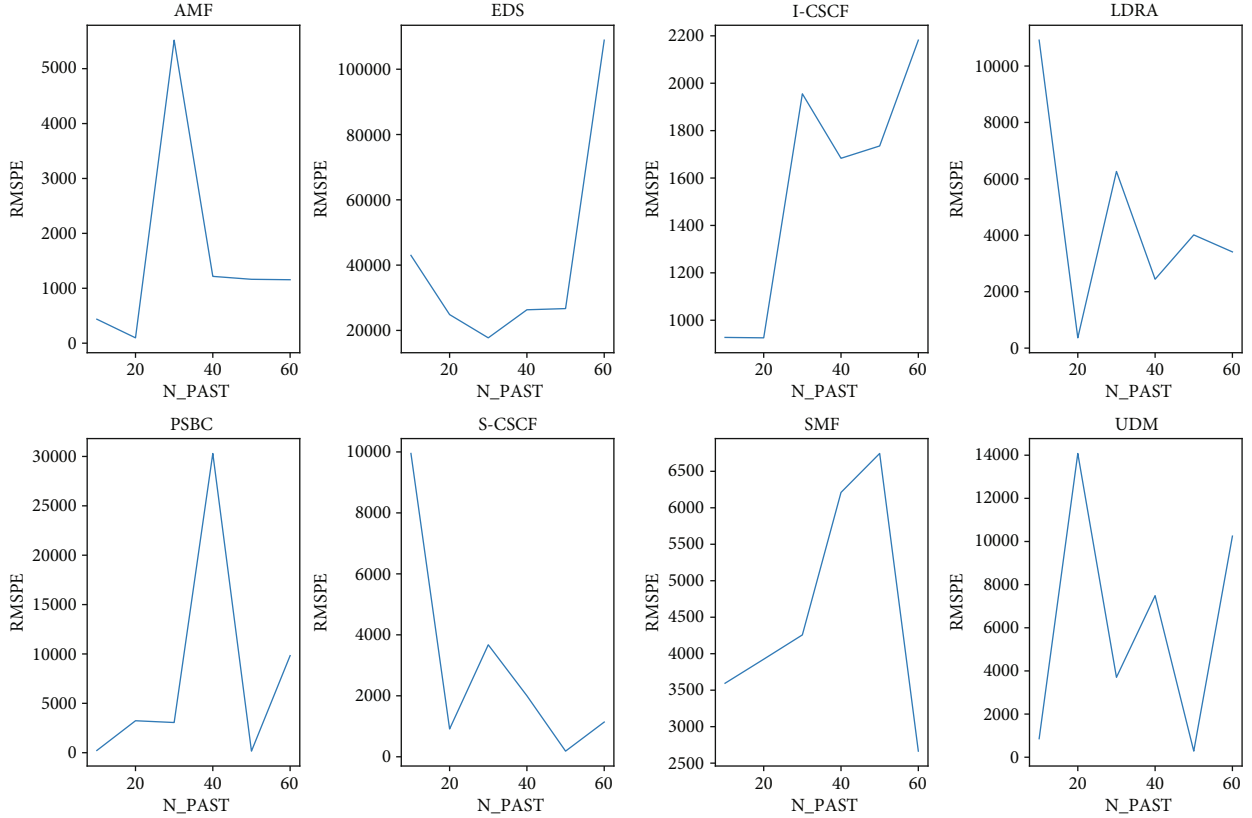


FIGURE 9: RMSPE indicator curves of different N_PAST of autoformer.

In the following step, we conducted time series prediction experiments using the random forest model on eight network functions to investigate the impact of different MaxLags values. In the random forest time series prediction model, the MaxLags parameter plays a crucial role. It determines the number of lagged observations included in the model, allowing the algorithm to consider past values as predictors for future predictions. By adjusting the MaxLags parameter, we can control the memory or dependence of the model on past observations.

In our experimental setup, we conducted a parameter search from 10 to 60, with a step size of 10, to explore the impact of different MaxLags values on the prediction performance of the random forest model for eight network functions. The goal was to identify the optimal MaxLags values that minimize the root mean square percentage error (RMSPE) for each specific network element.

The selection of appropriate MaxLags values is crucial as it affects the model's ability to capture the underlying patterns and dependencies in the time series data. By setting higher MaxLags values, the model can consider a longer history of observations, potentially capturing long-term patterns and trends. On the other hand, lower MaxLags values allow the model to focus on more recent observations, which may be more relevant for short-term predictions.

Through our analysis and examination of Figure 5, we determined the optimal MaxLags values for each network element. For example, the AMF, I-CSCF, and UDM ele-

TABLE 9: Confirmed N_PAST of autoformer predictor.

No.	Element type	N_PAST	No.	Element type	N_PAST
1	PSBC	10	5	I-CSCF	10
2	S-CSCF	50	6	UDM	50
3	EDS	30	7	SMF	60
4	LDRA	20	8	AMF	20

ments showed the lowest RMSPE when MaxLags was set to 60. In contrast, the EDS, S-CSCF, and LDRA elements achieved the best results with MaxLags of 10. When MaxLags is 20, the best result is achieved for SMF network functions. Lastly, the PSBC and UDM elements demonstrated improved prediction accuracy with MaxLags set to 50. These critical data points have been determined and will be used as the basis for forthcoming comparative experiments. The final confirmed parameter values are documented in Table 5.

These findings highlight the importance of selecting the appropriate MaxLags value for each specific network element. By choosing the optimal MaxLags value, we can improve the accuracy and overall performance of the random forest model in time series prediction tasks.

Subsequent to the preceding experimental steps involving the random forest model, we proceeded to conduct the same experiments utilizing the ExtraTrees model. The

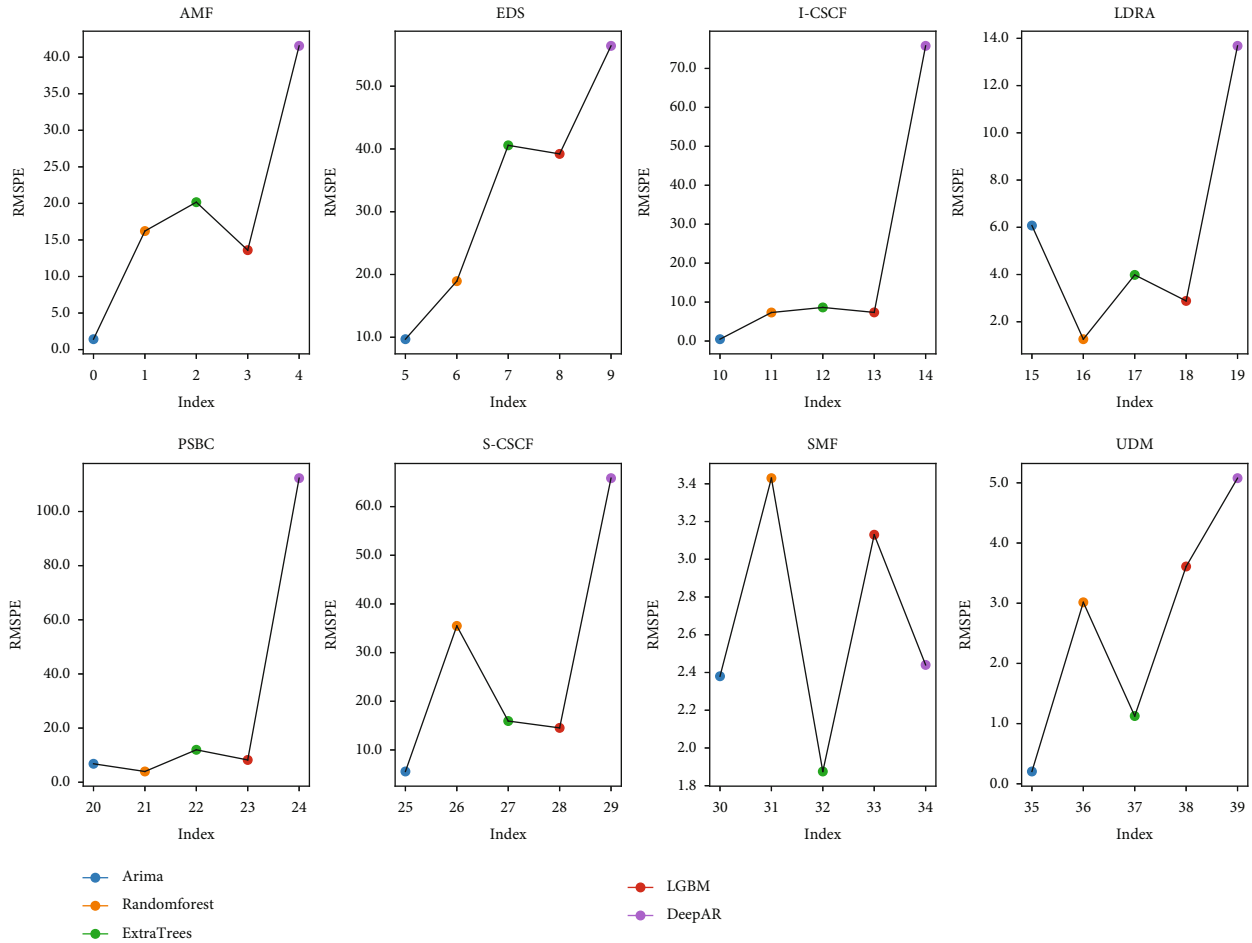


FIGURE 10: Performance comparison of optimal network function models based on RMSPE metric.

results of these experiments are illustrated in Figure 6. Specifically, we sought to determine the optimal MaxLags values across the eight network functions. In contrast to the random forest model, the ExtraTrees model is known for its ability to construct decision trees using random subsets of features and samples. This randomness enhances the diversity and robustness of the model, leading to improved generalization and accuracy. The final confirmed parameter values are documented in Table 6.

The LGBM (light gradient boosting machine) model is a powerful and efficient gradient-boosting algorithm. It excels in handling large-scale datasets with its optimized algorithm and histogram-based approach for split point computations.

In the subsequent analysis, we replaced the existing model with the LGBM model and configured the learning rate to be 0.01. Figure 7 depicts the results of the present experiment. Our objective was to explore the optimal MaxLags for each of the eight network functions. The final confirmed parameter values are documented in Table 7.

In our upcoming experiments, we conducted separate trials for each of the eight types of network element data. Our aim was to determine the optimal value of “N_PAST” for each specific type.

The “N_PAST” refers to the number of past observations (or time steps) used as input to the model for making predictions. By varying “N_PAST” independently for each network element type, we aimed to identify the most suitable value that would yield the best forecasting results for that specific type of data.

In the following experiment, we employ the DeepAR model with a configuration of 3 LSTM layers. Through empirical investigation and evaluation, we analyzed the performance of the DeepAR model across different values of “N_PAST” for each network element type, as illustrated in Figure 8. By selecting the optimal “N_PAST” value for each case, we sought to enhance the accuracy and effectiveness of our time series predictions tailored to the characteristics of each network element type. The final confirmed parameter values are documented in Table 8.

The autoformer model employs a multilayered encoder-decoder structure, allowing it to encode historical information and generate accurate predictions. Through an iterative training process, the model optimizes its parameters to minimize the discrepancy between predicted and actual values. In the following experiment, we employ the autoformer model with a configuration of 3 encoder layers and 3

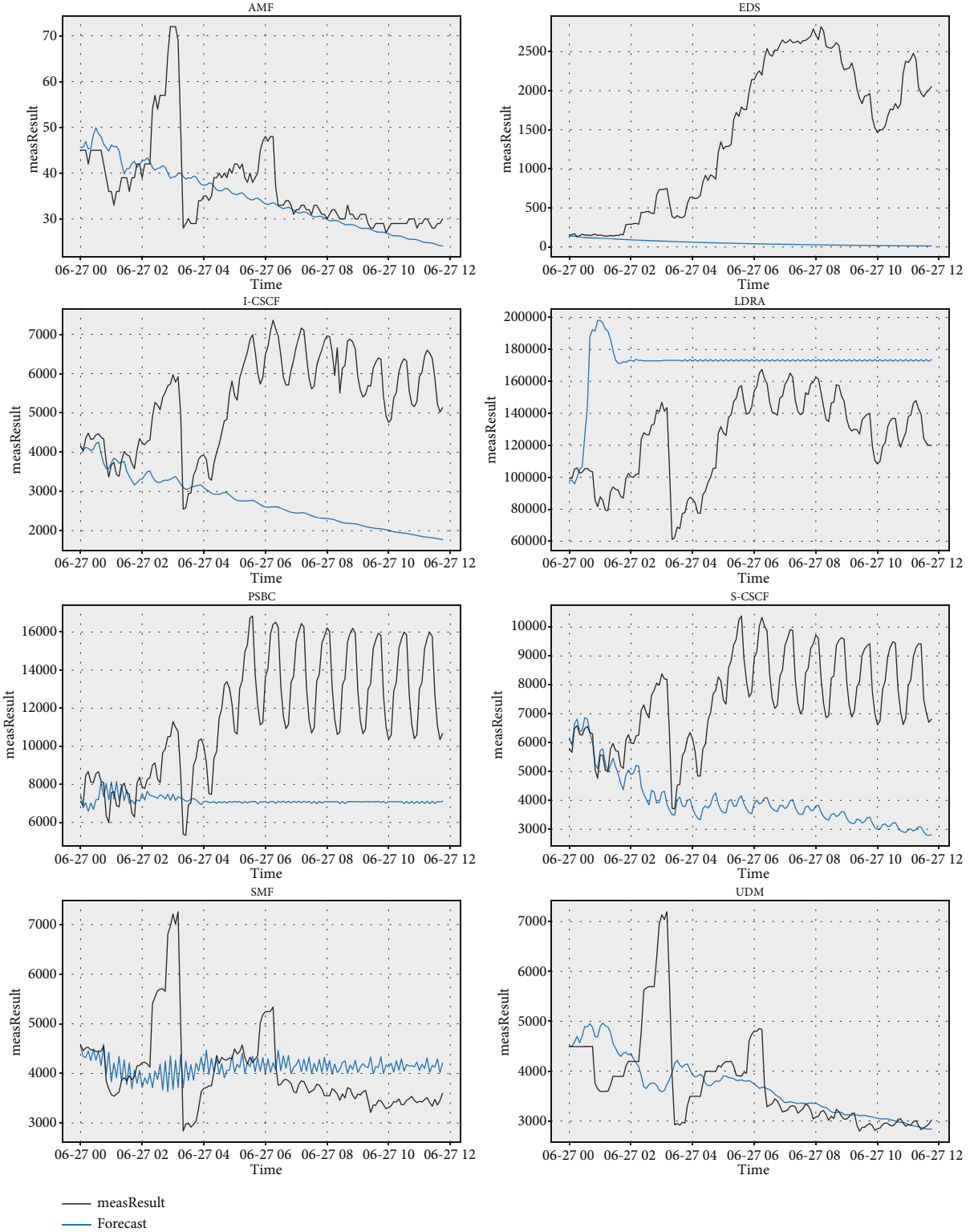


FIGURE 11: Predicted curves for network functions.

decoder layers. We conduct experiments on eight types of network element data individually to determine their respective optimal N_PAST values. The results of these experiments are illustrated in Figure 9.

Furthermore, we present the optimal N_PAST values obtained from the experiments in Table 9.

While autoformer has been proposed as a promising model for long-term time series forecasting, its applicability

TABLE 10: RMSPE of optimal network function prediction models.

Model	Type	RMSPE	Model	Type	RMSPE
ARIMA	AMF	1.42	ARIMA	EDS	9.69
RandomForest	AMF	16.21	RandomForest	EDS	18.93
ExtraTrees	AMF	20.16	ExtraTrees	EDS	40.58
LGBM	AMF	13.60	LGBM	EDS	39.21
DeepAR	AMF	41.55	DeepAR	EDS	56.42
ARIMA	I-CSCF	0.47	ARIMA	LDRA	6.07
RandomForest	I-CSCF	7.32	RandomForest	LDRA	1.26
ExtraTrees	I-CSCF	8.62	ExtraTrees	LDRA	3.98
LGBM	I-CSCF	7.35	LGBM	LDRA	2.88
DeepAR	I-CSCF	75.79	DeepAR	LDRA	13.68
ARIMA	PSBC	6.81	ARIMA	S-CSCF	5.56
RandomForest	PSBC	3.96	RandomForest	S-CSCF	35.48
ExtraTrees	PSBC	11.97	ExtraTrees	S-CSCF	15.93
LGBM	PSBC	8.22	LGBM	S-CSCF	14.52
DeepAR	PSBC	112.31	DeepAR	S-CSCF	65.84
ARIMA	SMF	2.38	ARIMA	UDM	0.20
RandomForest	SMF	3.42	RandomForest	UDM	3.01
ExtraTrees	SMF	1.87	ExtraTrees	UDM	1.12
LGBM	SMF	3.13	LGBM	UDM	3.61
DeepAR	SMF	2.44	DeepAR	UDM	5.07

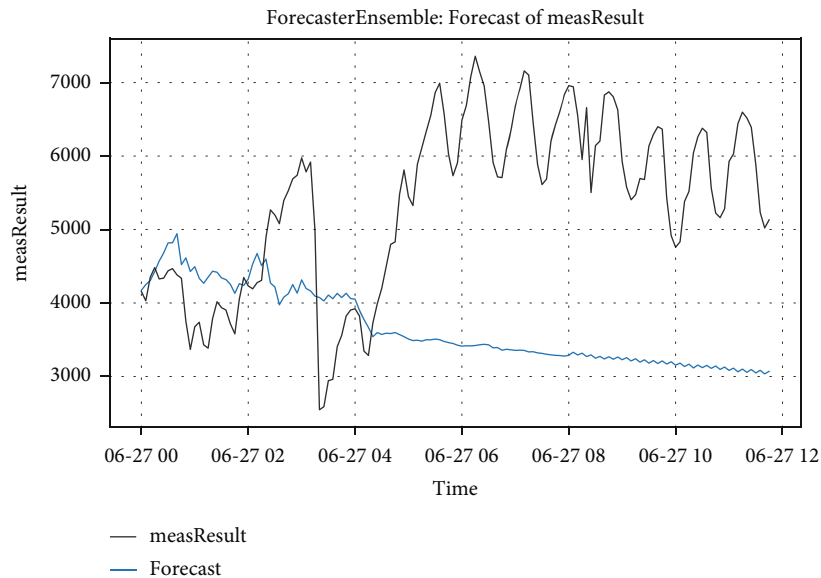


FIGURE 12: Predicted curves of ensemble model for I-CSCF network function.

to the specific dataset used in this study needs to be carefully considered. Based on the experimental results, it is evident that this model is clearly not suitable for short-term data in this study. Therefore, we will not consider comparing it with other models.

4.3. Comparative Experiments and Analysis for Network Functions. Based on the experiments conducted above, optimal parameters have been selected for each type of network element under each model. A separate test was then con-

ducted on the test dataset, evaluating the experiments based on the root mean square percentage error (RMSPE) metric. The experimental results are presented in Figure 10, illustrating the performance comparison between different models. The specific RMSPE of different models are presented in Table 9.

By observing the graph, it can be seen that the optimal models for five network functions are the ARIMA models, specifically for AMF, EDS, I-CSCF, S-CSCF, and UDM. On the other hand, the RF model is identified as the optimal

model for LDRA and PSBC network functions, while the ET model is the optimal choice for the SMF network element. Due to the small size and short time span of the dataset, it is expected that deep learning models, which are suitable for capturing long-range sequential features, would not perform well in this context. The experimental results in this study provide support for this point. In Figure 10, it is evident that the predictive performance of the deep learning model, DeepAR, is unsatisfactory.

Figure 11 illustrates the prediction curves obtained using the optimal models for each network element. The black line represents the actual values to be predicted, while the blue line represents the final prediction curves generated by the models. Although we applied some preprocessing techniques to the dataset, such as filling missing values and constraining ranges, the data still exhibits significant fluctuations, lacking sufficient stability and regularity. The experimental results presented in Figure 11 indicate that the overall time performance curves of the eight network functions are challenging to accurately predict. However, when examined individually, the AMF, UDM, and SMF network functions demonstrated relatively accurate prediction results.

4.4. Ensemble Model Analysis. Ensemble model analysis involves the systematic evaluation and examination of ensemble models, which are a combination of multiple individual models to improve predictive accuracy and stability. Various ensemble techniques, such as bagging, boosting, and stacking, are applied to integrate predictions from diverse base models. In this paper, a method of ensemble model analysis was employed, which involved averaging the predictions of two individual models. The ensemble model will simply take the mean prediction of individual models. This approach is a form of ensemble model analysis.

The I-CSCF network element was chosen to validate our hypothesis. Based on the content of Figure 10 and Table 10, we selected the ARIMA model and the random forest model, which ranked first and second in terms of prediction performance. We constructed an ensemble model and the final prediction is shown in Figure 12. From the graph, it is evident that the ensemble model outperformed the individual models in the first half of the prediction curve. The trend of the curve indicates that the ensemble model achieved more accurate fitting.

Ensemble model analysis demonstrates the efficacy of averaging the predictions of two models, providing valuable insights into the performance and reliability of this approach. This contributes to the wider understanding and adoption of ensemble modeling techniques within the field.

5. Conclusions

This paper presents a comprehensive comparative analysis of time series forecasting techniques applied to predict the performance of network functions in 5G networks. By employing a range of time series forecasting algorithms, the study is aimed at achieving accurate predictions for network function performance and providing a detailed comparison of their respective performances. Despite the

inclusion of both machine learning and deep learning models in this study, along with a thorough comparative analysis, there are still certain limitations that need to be addressed.

In this study, the data was collected at a five-minute interval, spanning a duration of a few days. As a result, the absence of observable seasonal patterns precluded the utilization of the SARIMA model and Prophet model, well-established methods for capturing seasonality feature. Additionally, the dataset lacks long-term sequential features, which hinders the deep learning models from leveraging their inherent advantages. The absence of such characteristics restricts the ability of deep learning models to effectively capture temporal dependencies and patterns over extended periods. To address this limitation, we plan to expand our research by incorporating longer-term experimental data, which will provide a more comprehensive understanding of the capabilities of deep learning models. By including a wider time range, we aim to gain valuable insights into seasonal and long-term sequential features and enhance the robustness of our analysis. Furthermore, we intend to explore the use of real-time data in future work to establish further experimental validation, with the ultimate objective of deploying our findings within real-world production and operational systems.

Data Availability

The datasets generated for this study can be found at <https://github.com/Kevin-chen-sheng/NWDAF-NFPP>.

Conflicts of Interest

The authors declare that there are no conflict of interest and we do not have any possible conflicts of interest.

Acknowledgments

The authors thank the financial support provided by the Research Institute of China Telecom Corporation Limited and the encouragement of colleagues in the Research Institute.

References

- [1] D. Chen, Q. Song, Y. Zhang, L. Li, and Z. Yang, "Identification of network traffic intrusion using decision tree," *Journal of Sensors*, vol. 2023, Article ID 5997304, 9 pages, 2023.
- [2] A. Gupta and R. K. Jha, "A survey of 5G network: architecture and emerging technologies," *IEEE Access*, vol. 3, pp. 1206–1232, 2015.
- [3] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, 2014.
- [4] S. Sevgican, M. Turan, K. Gokarlan, H. B. Yilmaz, and T. Tugcu, "Intelligent network data analytics function in 5G cellular networks using machine learning," *Journal of Communications and Networks*, vol. 22, no. 3, pp. 269–280, 2020.

- [5] M. Ramachandran, T. Archana, V. Deepika, A. A. Kumar, and K. M. Sivalingam, "5G network management system with machine learning based analytics," *IEEE Access*, vol. 10, pp. 73610–73622, 2022.
- [6] X. Lin, "An overview of 5G advanced evolution in 3GPP release 18," *IEEE Communications Standards Magazine*, vol. 6, no. 3, pp. 77–83, 2022.
- [7] W. Chen, J. Montojo, J. Lee, M. Shafi, and Y. Kim, "The standardization of 5G-advanced in 3GPP," *IEEE Communications Magazine*, vol. 60, no. 11, pp. 98–104, 2022.
- [8] A. Chouman, D. M. Manias, and A. Shami, "Towards supporting intelligence in 5G/6G core networks: NWDAF implementation and initial analysis," in *2022 International Wireless Communications and Mobile Computing (IWCMC)*, Dubrovnik, Croatia, 2022.
- [9] R. Casellas, R. Martínez, L. Velasco et al., "Enabling data analytics and machine learning for 5G services within disaggregated multi-layer transport networks," in *2018 20th International Conference on Transparent Optical Networks (ICTON)*, Bucharest, Romania, July 2018.
- [10] C. Zhou and N. Ansari, "Securing federated learning enabled NWDAF architecture with partial homomorphic encryption," *IEEE Networking Letters*, vol. 5, no. 4, pp. 299–303, 2023.
- [11] H.-W. Kao and E. H.-K. Wu, "QoE sustainability on 5G and beyond 5G networks," *IEEE Wireless Communications*, vol. 30, no. 1, pp. 118–125, 2023.
- [12] M. Mhedhbi, S. Elayoubi, and G. Leconte, "AI-based prediction for ultra reliable low latency service performance in industrial environments," in *2022 18th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Thessaloniki, Greece, October 2022.
- [13] C. Hernández-Chulde and C. Cervelló-Pastor, "Intelligent optimization and machine learning for 5G network control and management," in *Paper presented at the Highlights of Practical Applications of Survivable Agents and Multi-Agent Systems*, p. 17, The PAAMS Collection: International Workshops of PAAMS 2019, Ávila, Spain, 2019.
- [14] F. Díaz González, *Federated learning for time series forecasting using LSTM networks: exploiting similarities through clustering*, [M.S. thesis], E.T.S. de Ingenieros Informáticos (UPM), 2019.
- [15] P. Li, Y. Shi, Y. Xing et al., "Intra-cluster federated learning-based model transfer framework for traffic prediction in core network," *Electronics*, vol. 11, no. 22, p. 3793, 2022.
- [16] J. J. T. P. Xia, *China 5G: opportunities and challenges*, vol. 46, article 102295, Elsevier, 2022.
- [17] M. Panek, A. Pomykała, I. Jabłoński, and M. Woźniak, "5G/5G + network management employing AI-based continuous deployment," *Applied Soft Computing*, vol. 134, article 109984, 2023.
- [18] Y. Jeon, H. Jeong, S. Seo, T. Kim, H. Ko, and S. Pack, "A distributed nwdaf architecture for federated learning in 5g," in *2022 IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, January 2022.
- [19] F. Lyu, J. Zhu, Z. Wang et al., "5G Close Loop Proactive Optimization Using Network Data Analysis Function," in *Signal and Information Processing, Networking and Computers. ICSINC 2022*, Y. Wang, Y. Liu, J. Zou, and M. Huo, Eds., vol. 996 of Lecture Notes in Electrical Engineering, Springer, Singapore, 2023.
- [20] K. Abbas, T. A. Khan, M. Afaq, and W.-C. Song, "Ensemble learning-based network data analytics for network slice orchestration and management: an intent-based networking mechanism," in *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*, Budapest, Hungary, April 2022.
- [21] Y.-E. Kim, Y. S. Kim, and H. Kim, "Effective feature selection methods to detect IoT DDoS attack in 5G core network," *Sensors*, vol. 22, no. 10, p. 3819, 2022.
- [22] A. Reyhanoglu, E. Kar, F. E. Kumeç et al., "Machine learning aided NR-V2X quality of service predictions," in *2023 IEEE Vehicular Networking Conference (VNC)*, Istanbul, Turkey, 2023.
- [23] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "DeepAR: probabilistic forecasting with autoregressive recurrent networks," *International Journal of Forecasting*, vol. 36, no. 3, pp. 1181–1191, 2020.
- [24] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: decomposition transformers with auto-correlation for long-term series forecasting," *Advances in Neural Information Processing Systems*, vol. 34, pp. 22419–22430, 2021.
- [25] P. Rodríguez, M. A. Bautista, J. Gonzalez, and S. Escalera, "Beyond one-hot encoding: Lower dimensional target embedding," *Image and Vision Computing*, vol. 75, pp. 21–31, 2018.