Journal of Information Systems and Telecommunication
Vol.14, No.4, July-September 2026, 182-195
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

A Hybrid Deep Learning Model Based on BLSTM-GRU for Enhanced Call Volume Forecasting in Call Centers

Armaghan Nikfar[1], Sasan Harifi[1*], Javad Mohammadzadeh[1]

[1].Department of Computer Engineering, Ka.C., Islamic Azad University, Karaj, Iran

Received: 05 Feb 2026/ Revised: 04 Sep 2026/ Accepted: 23 Sep 2026

Abstract

This research aims to develop a forecasting model for call center call volume by incorporating characteristics unique to each country and call center environment. Using historical call data, calendar events, and other relevant input variables, the goal is to create a framework that is capable of accurately simulating and predicting the volume of incoming calls over specified time periods. For this purpose, we employ advanced recurrent neural network architectures, including MLP, RNN, LSTM, and a hybrid BLSTM-GRU model, to analyze temporal patterns in call data. The BLSTM-GRU model, in particular, uses two-way and gating processing mechanisms to more effectively capture complex temporal dependencies. The dataset consists of daily call volume over a 4-year period along with text features formatted in Excel for ease of training and evaluation. Furthermore, the forecasting framework integrates a comparison mechanism that compares the predicted call volume against the actual values to assess the accuracy and robustness of the model. Experimental results show that the BLSTM-GRU model outperforms standard architectures, achieving a forecasting accuracy of 94.12%, whereas the optimized single LSTM baseline reaches 81.35%. To increase operational efficiency, the framework also generates shift schedules that comply with call center standards and prevent call queues by recommending the required number of agents. The superior performance of the proposed BLSTM-GRU model highlights its potential application in various contact centers around the world and provides a scalable solution for intelligent contact center management.

Keywords: Long Short-Term Memory (LSTM); Call Center; BLSTM-GRU; Hybrid Deep Learning; Prediction.

1- Introduction

Accurately forecasting call volume in call centers is a critical operational challenge that directly impacts service quality and staffing efficiency. According to Hussain et al [1], predicting traffic flow from live data streams necessitates the development of robust data-driven models. This task is inherently challenging due to the complex, dynamic nature of traffic patterns and their temporal dependencies. A key difficulty lies in accurately capturing the diverse and often unpredictable variations in traffic behavior. While early efforts relied on statistical methods and shallow neural networks with limited success, recent advances in deep learning have demonstrated significant potential, offering more accurate and reliable predictions.

To enhance the understanding of spatial and temporal dependencies in traffic data, some researchers have explored traditional machine learning techniques such as Support Vector Machines (SVM). For instance, Feng et al [32] proposed an Adaptive Multi-Kernel SVM with Spatial–Temporal Correlation (AMSVM-STC) for short-term traffic flow prediction. This approach focuses on modeling the inherent nonlinearity and randomness of traffic behavior at specific Points of Interest (POIs) by leveraging probabilistic modeling techniques. Although these methods offer valuable insights, they often struggle to scale with complex temporal sequences compared to modern deep learning frameworks.

Convolutional neural networks (CNNs) are another popular approach for traffic prediction, known for their ability to model spatial features. Ma et al. [3] proposed a CNN-based method that learns traffic as images and predicts large-scale, network-wide traffic speed with high accuracy.

While the above approaches focus on general traffic forecasting using pre-built machine learning and deep learning models, our research addresses a specific and operationally critical area, which is call volume forecasting in call centers. Previously, a BLSTM model [4] has been trained and fitted with historical call data from an Iranian

✉ **Sasan Harifi**
harifi@iau.ac.ir

call center (Shatel) and incorporated specific features of the Persian calendar, such as national and religious holidays. The model predicted both hourly call volume and the optimal number of agents per shift. It incorporated micro-level operational factors, including the average number of calls per minute per agent, the probability of agent unavailability due to downtime, and the average call handling time per shift. By integrating these factors, this model can proactively suggest accurate staffing schedules to reduce queue formation and improve customer satisfaction.

We propose a novel hybrid architecture that combines bidirectional LSTM (BLSTM) and Gated Recurrent Unit (GRU) to further enhance prediction accuracy and robustness. The goal of this integration is to enhance the long-term dependency absorption capability of BLSTM and the lightweight and efficient structure of GRU, resulting in a more robust model that is able to more effectively address complex call center traffic patterns. We also want to use the Shatel call center shift scheduling policies to schedule experts on shifts so that the improved model can output a year of incoming calls with the required experts, both respecting the probabilistic principles and considering the expert shift scheduling policies. The main contributions of this paper are a novel hybrid BLSTM-GRU for call volume forecasting; integration of seven culturally specific calendar features (Nowruz, Ramadan, Muharram, etc.); a probabilistic agent scheduling mechanism that accounts for long-call variability; and a generalizable architecture adaptable to any country's calendar.

The remainder of this paper is organized as follows: Section 2 provides a review of related works in the field of call volume forecasting and the application of deep learning models in time series prediction. Section 3 introduces the proposed hybrid BLSTM-GRU approach. Section 4 presents the experimental results. Finally, Section 5 concludes the paper by summarizing the main findings and suggesting future research directions aimed at improving the scalability and adaptability of the proposed framework for global call center applications.

2- Related Works

Accurate forecasting of incoming call volumes in call centers plays a crucial role in ensuring service quality, reducing customer wait times, and optimizing workforce allocation. An overloaded call center, caused by underestimation of call traffic, leads to long queues, customer dissatisfaction, and service delays. Conversely, overestimation results in unnecessary staffing costs and resource waste. Thus, predictive modeling has become an essential strategy for modern call centers seeking to balance operational efficiency with high service-level standards. According to Aattouri et al [15], predicting incoming call flows enables better organization within call centers. It also reduces the number of lost calls and improves customer satisfaction. Also, Kanthanathan et al [16] said that a call center, customer service agents (CSAs) provide valuable products or information to the provider. A key requirement in a call center is to ensure customer satisfaction by having enough customer service agents (CSAs) to support incoming calls, while avoiding excess staffing to control costs.

Few studies have been conducted on the forecasting of call arrivals. Nikfar and Mohammadzadeh [4] proposed a deep learning-based framework using a Bidirectional Long Short-Term Memory (BLSTM) network to predict hourly call volumes at Shatel, a leading Iranian Internet Service Provider. The model was trained on four years of historical call data and uniquely incorporated temporal features specific to the Persian calendar, including official holidays (e.g., Nowruz), religious events, and weekends. By dynamically encoding these culturally significant features, the model was capable of adjusting to different calendar years without requiring manual feature engineering for each prediction cycle. The BLSTM model captured both forward and backward temporal dependencies in the data, enabling it to identify complex sequential patterns in call traffic.

Beyond predicting the volume of calls, the model extended its functionality to support human resource planning by estimating the number of agents required per hour. This was achieved through the integration of multiple operational factors such as average call handling time, probability of agent availability during breaks, and historical workload fluctuations. These elements were incorporated as input features, allowing the model to generate real-time staffing recommendations that aligned with service-level agreements and workload forecasts. The system output included a comprehensive Excel file containing predictions for each date and hour, along with corresponding queue size estimates and calculated agent requirements. When tested against real call data from the year 1402 (solar Hijri calendar), the model achieved a prediction accuracy of 90.56%. Compared to standalone LSTM and GRU implementations, which yielded accuracies of 89.07% and 88.33%, respectively, the BLSTM model demonstrated superior performance in both prediction precision and practical deployment outcomes. Notably, simulation trials based on BLSTM-generated schedules showed a complete elimination of call queues during test shifts.

These promising results highlighted the effectiveness of our approach in capturing the unique behavioral patterns of call center traffic and supporting intelligent workforce planning. Building on this foundation, the present study introduces a hybrid model that combines the BLSTM architecture with Gated Recurrent Unit (GRU) networks to further enhance predictive performance and computational efficiency.

Esmail et al [23] conducted a study on predicting the volume of incoming calls in a telecommunications call

center at 15-minute intervals using real-world data from 2022–2023. They evaluated five models, including Random Forest, XGBoost, LSTM, GRU, and a GRU–LSTM hybrid using temporal features. Their results showed that deep learning architectures are effective for high-resolution forecasting in call centers.

As mentioned, we want to improve our proposed model to a hybrid BLSTM-GRU model. Numerous studies have shown that combining different deep learning architectures can significantly improve prediction performance, particularly when dealing with complex time-series data such as call center traffic. The idea behind hybrid modeling is to leverage the complementary strengths of each architecture. While BLSTM networks are highly effective in capturing bidirectional temporal dependencies and long-range contextual patterns, GRU models are known for their computational efficiency and ability to handle vanishing gradient problems in relatively shallow structures. Zafar et al [5] proposed a hybrid LSTM-GRU architecture for predicting urban traffic speed using heterogeneous data sources, and reported that the combined model outperformed standalone LSTM and GRU models in terms of both accuracy and stability.

Similarly, Gao et al [6] demonstrated that a hybrid model of optimized LSTM and GRU achieved better results in financial time-series forecasting compared to individual models, particularly when modeling high-frequency, noisy datasets. These findings are echoed by Hussain et al [1], who showed that GRU-based approaches, when integrated with other RNN structures, can capture nonlinear dynamics in traffic data more effectively and with reduced training time.

Also, Lawal et al [7] demonstrated that combining architectures such as LSTM, GRU, and BLSTM improves both accuracy and robustness in forecasting tasks ranging from urban traffic to renewable energy and call behavior. These works provide a strong motivation for integrating BLSTM and GRU in our current research, aiming to achieve a similar enhancement in call volume forecasting performance. Li et al [8] introduced a Multi-GRU architecture for forecasting electricity generation. Their model, compared with classic RNNs and LSTMs, not only showed superior forecasting accuracy but also reduced training time—an essential factor in real-time planning systems. The authors emphasized that stacking GRU layers allowed the model to learn hierarchical temporal features more effectively, resulting in better generalization.

In other studies Kong et al [19] presented a systematic review of deep learning methods for time series forecasting. They also reviewed enhancement methods for feature extraction, such as component separation, time-frequency transformation, pretraining, and segmentation. By reviewing diverse datasets in the fields of energy, transportation, health, meteorology, and economics, the study identified challenges such as lack of privacy, lack of interpretability, lack of temporal continuity, and limitations of parallel computing, and suggested future directions including representation learning, counterfactual prediction, broadcast models, and integration with reinforcement learning.

Casolaro et al. [20] conducted a review of deep learning for time series forecasting and modern architectures, including convolutional, recurrent, graph neural networks, deep Gaussian processes, generative adversarial networks, diffusion models, and transformers. The approach is to categorize models for short-term and long-term forecasting and analyze the strengths and weaknesses of each. Comparison of models on benchmark datasets shows that improved transformers (such as Informer, Autoformer, and PatchTST) perform better in long-term forecasting than traditional recurrent models. They also identified the main challenges as open issues, including a lack of confidence interval, overfitting, the need for long time series, and dynamic instability.

Li and Law [21] conducted a review of deep learning models for time series forecasting, examining classical methods (such as ARIMA) and modern models including CNN, RNN, GNN, and transformers in four main categories. Using the ETT energy dataset and the same hardware, the authors compared the performance of nine models in terms of MAE and MAPE metrics. The results show that the SCINet model has the best accuracy, followed by the MTS-Mixers and FEDformer models, while the simple transformer-based models have poorer performance. The paper concludes that the SCINet architecture is very effective for energy time series data due to its ability to capture both short-term and long-term dependencies.

Yang and Wang [22] studied financial time series forecasting using a bidirectional LSTM (BiLSTM) neural network and compared it with LSTM, SVR, and ARIMA models on China CSI 300 index data. The working method was that BiLSTM, with two separate hidden layers, simultaneously considers past and future information, while LSTM only uses past information. The comparison results show that BiLSTM has the lowest RMSE and MAPE errors and the highest coefficient of determination in the test set, and performs better than LSTM, SVR, and ARIMA. The only disadvantage of BiLSTM is its longer training time (about 14 minutes) compared to LSTM (about 11 minutes).

Table 1 shows a summary of previous studies. While existing hybrid models have been applied to traffic, energy, and financial forecasting, few studies have explored hybrid BLSTM-GRU architectures specifically for call center call volume prediction. Furthermore, most call center forecasting models do not incorporate culturally specific calendar features (e.g., religious holidays, national events) or provide integrated staffing recommendations. To fill this gap, we propose a novel hybrid BLSTM-GRU model that captures bidirectional temporal dependencies in call traffic, integrates calendar-specific features (Nowruz, Ramadan,

Muharram, etc.), and estimates required agent shifts based on probabilistic adjustments for call duration variability. The proposed model is designed to be generalizable to any call center environment with appropriate feature engineering.

3- Proposed Hybrid BLSTM-GRU Approach

In today's organizations, managing incoming call traffic in call centers has become one of the primary operational challenges. Accurate forecasting of call volumes can significantly enhance service quality and customer satisfaction by enabling better workforce planning and resource allocation. Within a call center, a wide range of functions are carried out to ensure effective communication with customers and the delivery of timely and efficient support. These include handling customer inquiries, resolving complaints, processing orders, managing after-sales support, conducting sales of new products, providing product information, offering purchase consultations, delivering technical assistance, and more [4].

According to Ibrahim et al [9] efficiently managing a call center is a challenging task, as managers must make staffing and scheduling decisions to balance staff costs and service quality, which are always in conflict when there is uncertainty about demand. Therefore, developing intelligent solutions that can help reduce excessive call traffic and improve call distribution has become increasingly important. Predictive modeling, in particular, plays a critical role in enabling proactive resource management, preventing long queues, minimizing wait times, and ensuring that customers are efficiently routed to the right agents. By anticipating peak hours, holidays, and other high-demand periods, organizations can better align their staffing and support strategies with actual customer needs, ultimately improving service levels while reducing operational costs.

In this study, with the aim of improving the accuracy of inbound call prediction in Shatel's call center (an Internet service provider company from Iran), a hybrid architecture based on two powerful deep learning models, namely BLSTM (Bidirectional Long Short-Term Memory) and GRU (Gated Recurrent Unit), has been designed and implemented. This combination was chosen because BLSTM has the ability to understand temporal dependencies in both the past and future directions, and GRU, due to its simpler structure, is both computationally more efficient and converges faster than LSTM.

The proposed hybrid BLSTM-GRU model, although trained on call traffic data from a specific organization, has been designed with a generalizable architecture that allows it to be adapted to any call center environment. The input features used—such as hourly timestamps, calendar-based events, and historical call volumes—are common across most call center systems and can be extracted or engineered in similar ways from any operational dataset. Furthermore, the model's reliance on recurrent neural networks makes it well-suited for capturing temporal patterns regardless of the specific business domain.

By retraining the model with domain-specific data from other organizations, it is possible to fine-tune its performance and accurately forecast call volumes in diverse industries such as telecommunications, healthcare, banking, and customer support. This flexibility makes the proposed approach a scalable and adaptable solution for intelligent workforce planning and resource optimization in modern call centers.

The reason for choosing these two models for the combination is based on the nature of call center data and the need to accurately analyze complex temporal patterns while maintaining computational speed and efficiency. BLSTM, with its ability to process sequences in two directions, provides a deeper understanding of call trends throughout the day and also on special days such as holidays or occasions. On the other hand, GRU has a significant

Table 1: A brief look at previous research

Author	Hybrid Model	Domain	Relevance to our Work
Li et al (2019) [8]	Multi-GRU	Electricity generation	Modeled hierarchical temporal structure
Gao et al (2021) [6]	LSTM-GRU	Financial time series	Hybrid model performance validation
Hussain et al (2021) [1]	Optimized GRU	Traffic flow prediction	GRU efficiency for temporal prediction
Lawal et al (2021) [7]	CNN+BLSTM	Wind speed forecasting	Bidirectional processing for time series
Zafar et al (2022) [5]	Hybrid LSTM-GRU	Urban traffic prediction	Hybrid architecture design
Nikfar and Mohammadzadeh (2025) [4]	BLSTM	Call-Center call prediction	Baseline: call volume prediction with Persian calendar features
Esmail et al (2026) [23]	Hybrid GRU-LSTM	Telecommunications call	High-frequency call volume prediction with time-based split
Current study (2026)	Hybrid BLSTM-GRU	Call-Center Optimization	First hybrid BLSTM-GRU for call volume with staffing

computational advantage due to its simpler structure, in situations where the data is large, or the model update must be performed in short time intervals. Combining these two models allows, on the one hand, to maintain the accuracy and analytical depth of the model, and on the other hand, to reduce the training time and processing cost. Previous studies have also shown that combining BLSTM and GRU can perform better than using each separately, especially in complex time-dependent problems such as call center call prediction.

To ensure the proposed architecture is not arbitrarily chosen, we conducted a systematic comparison of candidate models during the initial design phase. The selection criteria were the ability to capture long-term temporal dependencies, the capability to process bidirectional context, computational efficiency, and prediction accuracy on validation data. We reviewed four recurrent architectures, including unidirectional LSTM, unidirectional GRU, bidirectional LSTM (BLSTM), and a hybrid BLSTM followed by GRU layers. The BLSTM-GRU hybrid was selected because it achieves the best trade-off between accuracy and computational cost, while leveraging the complementary strengths of both architectures, namely BLSTM for bidirectional long-range context and GRU for efficient gating.

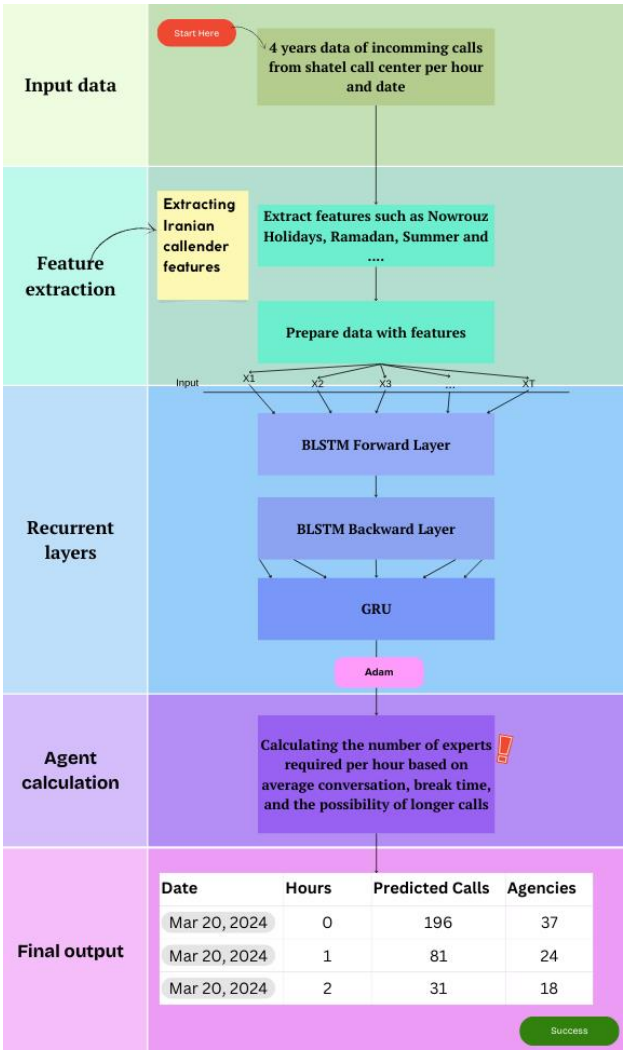


Fig 1. Diagram of proposed hybrid BLSTM-GRU

In continuation of the model description, the structure of the recurrent layers within the proposed hybrid architecture is illustrated below. As shown in Fig 1, the sequence begins with two stacked Bidirectional LSTM (BLSTM) layers followed by a GRU layer, each configured with specific memory units and output formats to efficiently model temporal dependencies and extract meaningful features from the input sequence.

3-1- Preprocessing

In the pre-processing phase, historical data of incoming calls to the Shatel Company call center over a four-year period were first collected and analyzed through Excel files. This set included more than 20,000 rows of data, each row representing calls recorded on a specific date and time. In this phase, the two features “date” and “call time” were processed as key indicators, since caller behavior changes not only on specific days but also at specific times of the day. Call volumes tend to peak during early morning, midday, and late business hours. Ignoring these temporal patterns reduces model accuracy. One of the main challenges in this phase was the need to accurately identify national and religious occasions in the Iranian and Islamic calendars, which have a direct impact on the volume of calls. To achieve this goal, the date of each data row was first converted to a standard format, considering common formats such as YYYY-MM-DD, DD/MM/YYYY, and MM/DD/YYYY. Then, using the Python programming language and the Hijri-Converter tool, the Gregorian date of each row was converted to its Hijri equivalent so that religious occasions could also be identified. These occasions included Ramadan, the first ten days of Muharram, Tasu'a, Ashura, the Nights of Ghadr, Eid Ghorban, Eid al-Ghadir, and other religious festivals and mornings. Table 2 shows a fake example of hourly incoming calls per date. Also, Fig 2 shows incoming calls in a 12-hour shift.

Table 2: Example of hourly incoming call data (synthetic sample)

Date	***Hour***	***Incoming Calls***
2024-03-20	00	196
2024-03-20	01	81
2024-03-20	02	31
2024-03-20	03	18
2024-03-20	04	5
2024-03-20	05	3
2024-03-20	06	14

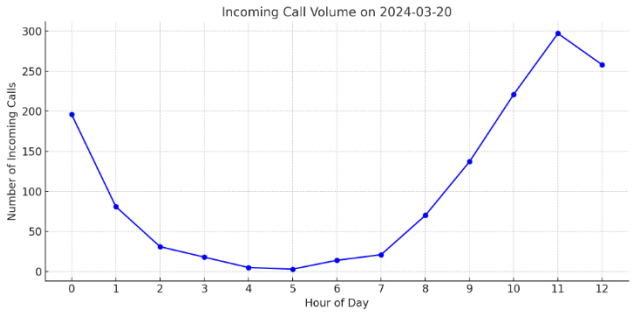


Fig 2. Incoming calls for 12 hour per day (Synthetic Sample)

Simultaneously with this processing, important time features were automatically calculated for each row. These features included Nowruz holidays (isNowruz), national holidays (isNationalHoliday), religious holidays (isReligiousHoliday), Ramadan days (isRamadan), Muharram days (isMoharam), and summer holidays (isSummerHoliday). These features were calculated dynamically, taking into account the structure of leap and non-leap years, so that the model can analyze data in subsequent years without the need for manual changes.

In addition to these features, the call hour was also considered an important behavioral factor and was directly included in the model. In order to increase accuracy, the model is able to learn in which hourly intervals the call traffic is higher or lower, and this will be effective in more accurate forecasting and more appropriate allocation of manpower for each shift. In the following, a sample of the hourly incoming call data used for training and evaluation is presented.

3-2- Feature Extraction

In the feature extraction phase, several time-related and culturally significant indicators were engineered and added to the dataset in order to enhance the model's ability to capture behavioral patterns in call volume fluctuations. These features were primarily derived from the Gregorian, Solar Hijri (Persian), and Hijri lunar calendars, reflecting both national and religious holidays that influence user activity.

The first feature, "isNowruz", was designed to identify whether a particular date falls within the Nowruz holiday period, which marks the Persian New Year and is known for causing significant changes in customer service load. Since the exact timing of Nowruz can vary slightly depending on whether the year is a leap year or not, the logic dynamically adjusts the range: from March 20 to April 1 in Gregorian leap years and from March 21 to April 2 in regular years. This roughly corresponds to 29 Esfand to 2 Farvardin in the Persian calendar.

Another feature, "isSummerHoliday", indicates whether the date lies within the summer vacation period, which typically spans from June 21 to September 21 in leap years, and June 22 to September 22 otherwise. This period is culturally associated with increased travel and reduced administrative activity in Iran, which often correlates with shifts in call volume trends.

To incorporate the influence of fixed national holidays, the "isNationalHoliday" feature was introduced. This binary variable is set to 1 if the date corresponds to key Iranian public holidays such as Nowruz, Islamic Republic Day, or Anniversaries of important characters. These dates are encoded based on historical data and adjusted each year for leap year alignment to ensure accuracy without requiring manual updates.

Religious events, which are based on the Hijri lunar calendar, were also considered. The "isRamadan" feature flags whether a given date falls within the ninth month of the Islamic calendar, known as Ramadan. As fasting during this period significantly affects daily routines and working hours, it is an essential factor in predicting changes in call traffic patterns.

Similarly, the "isMoharam" feature detects whether the date belongs to the first ten days of the month of Muharram, which includes the observances of Tasua and Ashura — major religious events during which social behaviors and service usage tend to shift markedly.

In addition to these, a generalized religious holiday marker, isReligiousHoliday, was added. This feature is activated if the date matches any of the manually curated list of important religious occasions, including Eid al-Adha, Eid al-Ghadir, Laylat al-Qadr, and the martyrdom anniversaries of Shia Imams. This list was encoded as a collection of specific Hijri month-day pairs and automatically matched against the Hijri conversion of each Gregorian date in the dataset. By engineering these features, the model becomes capable of learning from subtle yet impactful temporal signals embedded in culturally contextualized calendars, thus increasing its robustness and predictive performance across different years and seasonal variations. In the following, a sample of extracted features is presented in Table 3.

Table 3: Extracted features for model input and final output per hour

Date	*Hour*	*IsNowruz*	*IsSummerHoliday*	*IsRamadan*	*IsMoharram*	*IsNationalHoliday*	*IsReligiousHoliday*
2024-03-20	00	1	0	1	0	1	0
2024-03-20	01	1	0	1	0	1	0
2024-03-20	02	1	0	1	0	1	0
2024-03-20	03	1	0	1	0	1	0
2024-03-20	04	1	0	1	0	1	0
...	...	...	...	...	...	...	...
2024-03-21	00	1	0	1	0	1	0

3-3- Calculating of Required Agents Per Shift

Determining the optimal number of experts required for each shift is of particular importance in call center manpower management. This decision-making requires a comprehensive and accurate analysis based on the predicted call volume and the likely behavior patterns of the experts. In this study, the framework for calculating the need for experts is designed to take into account several factors, including average call duration, possible break times, the availability of experts, and the likelihood of longer calls, in order to plan manpower with greater accuracy and reliability.

The basic assumptions underlying the model include an average call duration of seven minutes and an average break time of ten minutes per agent during a shift. These breaks effectively reduce the agent's available working time. For instance, in a 60-minute hour, an agent's effective working time is reduced to 50 minutes. The agent's call handling capacity per hour C_a is therefore calculated as the ratio of effective working time T_e to average call duration T_c,

$$C_a = \frac{T_e}{T_c} \quad (1)$$

where $T_e = T_{shift} - T_{break}$ (in minutes) and T_c is the average call duration (7 minutes in this study). Given the predicted hourly call volume V_c, the required number of active agents N_{active} can be expressed as:

$$N_{active} = \frac{V_c}{C_a} \quad (2)$$

To account for the variability in call duration, particularly the incidence of long calls that consume more agent time, the model incorporates a probabilistic adjustment. Let pi denote the probability that an agent encounters a long call during the shift, and let α represent the average additional time (in minutes) that such long calls add beyond the average call duration. The effective capacity C'_a of an agent is then recalculated by adjusting the available time for call handling:

$$C'_a = \frac{T_e - pi \times \alpha}{T_c} \quad (3)$$

Consequently, the adjusted number of active agents required N'_{active} becomes

$$N'_{active} = \frac{V_c}{C'_a} \quad (4)$$

This adjustment ensures that fluctuations in call durations are reflected in the staffing estimates, improving accuracy and resilience under real-world conditions.

In addition to active agents, the model also considers inactive agents—those who may be temporarily unavailable due to breaks, administrative tasks, or other interruptions. Based on empirical observations and operational experience, the number of inactive agents $N_{inactive}$ is specified as a function of call volume, as summarized in Table 4.

Table 4: Estimated inactive agents based on hourly call volume

Call Volume per Hour V_c	*No. of Inactive Agents* $N_{inactive}$
$V_c < 10$	3
$10 \le V_c \le 20$	5
$20 < V_c \le 30$	7
$V_c > 30$	14

The total number of agents required N_{total} for a given shift is then the sum of active and inactive agents. Fig 3, shows required agents for one shift by incoming calls predicted proposed hybrid model.

$$N_{total} = N'_{active} + N_{inactive} \quad (5)$$

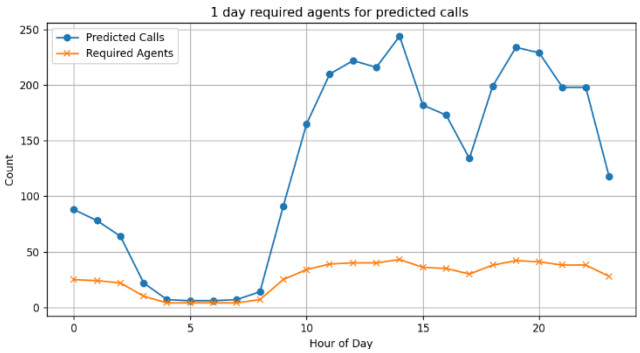


Fig 3. One day required agents for predicted calls by proposed model

3-4- Forecasting Formulation

In this study, the call volume forecasting problem is formulated as a one-step-ahead hourly forecasting task. Given the historical call volume sequence together with the corresponding temporal and calendar-based features, the proposed BLSTM-GRU model predicts the incoming call volume for the next hourly interval. The model does not employ recursive multi-step forecasting, where previous predictions are repeatedly used as future inputs. Instead, each hourly prediction is generated directly from the available historical sequence, which reduces error accumulation and provides a more stable forecasting framework for operational call center management.

3-5- The Proposed BLSTM-GRU Model

To enhance the accuracy of hourly call volume forecasting in call centers, this study introduces a hybrid deep neural architecture composed of Bidirectional Long Short-Term Memory (BLSTM) and Gated Recurrent Unit (GRU) layers. This design seeks to combine the representational power of BLSTM networks—which capture temporal dependencies in both forward and backward directions—with the computational efficiency of GRUs, resulting in a model that maintains high accuracy while remaining lightweight and trainable on real-world time series data.

The model takes a structured feature vector as input for each hourly observation. The input vector x_t at time step t includes behavioral and temporal features relevant to customer contact patterns.

$$x_t = [\,\text{hour}_t + \text{calls}_t + \text{isRamadan}_t + \text{isNowruz} + \text{isMoharram}_t + \text{isNationalHoliday}_t + \text{isReligiousHoliday}_t\,] \quad (6)$$

These vectors are organized as a time-ordered sequence $X = \{x1, x2, \dots, xT\}$ which that fed into the recurrent layers of the network. The first two layers of the model are Bidirectional LSTM (BLSTM) networks with 64 and 32 hidden units, respectively. Each BLSTM layer processes the sequence both forward and backward, learning contextual patterns from past and future time steps simultaneously. The hidden state at each time step t is represented by the concatenation of forward and backward hidden vectors.

$$h_t(BLSTM) = [\overrightarrow{h_t} \mid \mid \overrightarrow{h_t}] \quad (7)$$

The output of the final BLSTM layer is passed to a single GRU layer with 32 units, which refines the temporal representation and reduces feature dimensionality. The internal operations of the GRU follow standard formulations. Specifically, the update and reset gates zt and rt computed as follows,

$$z_t = \sigma(W_z\, x_t + U_z\, h_{t-1}), r_t = \sigma(W_r\, x_t + U_r\, h_{t-1}) \quad (8)$$

The candidate hidden state and final GRU output are then computed as:

$$\tilde{h_t} = tan\, h\big(W_h\, x_t + U_h(r_t \odot h_{t-1})\big)$$
$$h_t = (1 - z_t) \odot h_t - 1 + z_t \odot \tilde{h_t} \quad (9)$$

Following the recurrent layers, the output is processed through a series of three fully connected (Dense) layers with 64, 32, and 16 units, respectively. The first two layers utilize the ReLU activation function to model non-linear dependencies, while the final layer applies a linear activation function to produce the model output $\hat{y_t}$, representing the predicted number of incoming calls for the given time step:

$$\hat{y_t} = W_4 \cdot ReLU(W_3 \cdot ReLU(W_2 \cdot ReLU(W_1\, h_t + b_1) + b_2) + b_3) + b_4 \quad (10)$$

To enhance generalization and training efficiency, two regularization techniques are applied: Early Stopping, which halts the training process if validation loss fails to improve over 10 consecutive epochs, and ReduceLROnPlateau, which reduces the learning rate adaptively if performance plateaus.

After training, the model is saved and can be reloaded for inference. During prediction, new input data structured identically to the training data are fed into the model, which outputs hourly call volume forecasts. These outputs are subsequently used for decision-making in staffing and resource allocation within the call center.

Table 5 summarizes the internal structure of the proposed BLSTM-GRU model, outlining the type, number of units, activation functions, and configuration of each layer. It illustrates how the model processes a 7-dimensional input feature vector through a sequence of recurrent and dense layers to capture temporal patterns and produce accurate hourly call volume predictions. The table serves as a clear

Table 5: Summary of the proposed BLSTM-GRU architecture

Layer	***Type***	***Units***	***Activation***	***Configuration***
Input Layer	Sequence Input	1 × 7	–	Hourly feature vector
Layer 1	Bidirectional LSTM	64	tanh	return_sequences=True
Layer 2	Bidirectional LSTM	32	tanh	return_sequences=True
Layer 3	GRU	32	tanh	Sequence-to-vector compression
Dense Layer 1	Dense	64	ReLU	Feature abstraction
Dense Layer 2	Dense	32	ReLU	Further transformation
Dense Layer 3	Dense	16	ReLU	Final representation
Output Layer	Dense	1	Linear	Predicted hourly call volume

reference for understanding the architecture's design and functional flow. The final output of the proposed model prediction, estimated number of incoming calls for each hourly interval, is presented in Table 6.

Table 6: Final output of proposed BLSTM-GRU (Synthetic Sample)

Date	***Hours***	***Predicted Calls***	***Agents***
2024-03-20	0	196	37
2024-03-20	1	81	24
2024-03-20	2	31	18
2024-03-20	3	18	8
2024-03-20	4	5	4
2024-03-20	5	3	4
2024-03-20	6	14	7
2024-03-20	7	21	10
2024-03-20	8	70	23
2024-03-20	9	137	30
2024-03-20	10	221	40
2024-03-20	11	297	49

4- Experimental Result

This section provides basic details about data collection, implementation setup, comparison methods, and final results. We begin by describing the dataset used in our study, including its sources, structure, and sample data points. Next, we outline the implementation environment and specify the hardware and software setups used for the experiment. Then, we introduce the underlying models and algorithms used for the comparative analysis. Finally, we provide a comprehensive discussion of our empirical results, supported by tables, graphs, and analytical insights.

4-1- Data Collection

In this study, data was collected from the internal call management system of Shatel Company over a four-year period. The data was stored in Excel files and included the number of incoming calls to the call center at each hour of the day. For each day, the number of calls recorded in hourly time intervals was recorded, totaling more than 20,000 time records.

Due to the company's confidentiality policies, it is not possible to directly publish the actual data. However, the structure and characteristics of the data were preserved in this study, and synthetic samples with similar statistical characteristics were used for display and analysis. The preprocessing and feature extraction steps were performed using Python libraries such as pandas, datetime, and openpyxl.

In Table 7, we will clearly depict the unreal data for further understanding. Fig 4 illustrates the hourly distribution of incoming calls over two consecutive days, March 20 and March 21, 2020. As shown, call traffic exhibits a clear diurnal pattern, with lower volumes during early morning hours and a sharp increase toward late morning and noon. March 20 demonstrates significantly higher call volumes, particularly between 10:00 and 18:00, which may be attributed to its classification as a national holiday. In contrast, March 21, while still busy, reflects a more balanced and moderate call pattern throughout the day. This visualization supports the temporal feature engineering used in the model and highlights the need for dynamic staffing strategies based on hourly fluctuations.

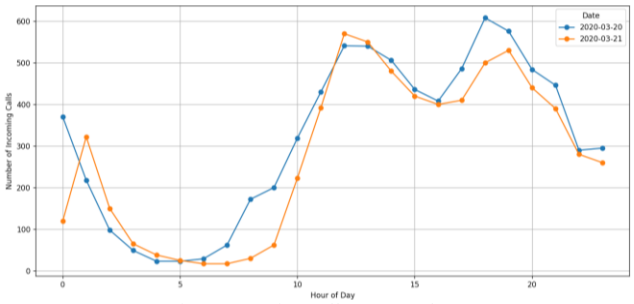


Fig 4. Incoming calls per two days

4-2- Implementation Details

To ensure efficient training and evaluation of our AI call prediction model, all experiments were conducted in a high-performance computing environment optimized for deep learning applications. The dataset was split into training (70%) and testing (30%). The model was trained on the training set and evaluated on the unseen test set. The implementation was done using Python as the primary programming language, using BLSTM and GRU as the main deep learning frameworks. Common hyperparameters across all models were batch size = 32, learning rate = 0.001 (Adam optimizer), early stopping patience = 10 epochs, and maximum 100 epochs. The BLSTM-GRU architecture consisted of two BLSTM layers (64 and 32 units) followed by one GRU layer (32 units) and three dense layers (64, 32, 16 units). LSTM and RNN used 64 units in a single layer. MLP used three dense layers (64, 32, 16 units). All models used ReLU activation except the output layer (linear). The experiments were performed on a workstation equipped with an NVIDIA RTX 4060 GPU (24 GB VRAM), an Intel Core i7-13650HX processor, and 64 GB DDR5 RAM. This advanced hardware configuration was crucial to handling the computational complexity of training deep neural networks on a large dataset of call center inbound calls.

Table 7: Incoming calls during 4 years from Shatel with features

Date	*Hours*	*Calls*	*ItsDayOff*	*IsRamadan*	*IsEidNowrooz*	*IsSumerHoliday*	*IsNationalHoliday*
2020-03-20	0	370	1	0	1	0	1
2020-03-20	1	218	1	0	1	0	1
2020-03-20	2	98	1	0	1	0	1
2020-03-20	3	49	1	0	1	0	1
2020-03-20	4	23	1	0	1	0	1
2020-03-20	5	23	1	0	1	0	1
2020-03-20	6	29	1	0	1	0	1
2020-03-20	7	62	1	0	1	0	1
2020-03-20	8	172	1	0	1	0	1
2020-03-20	9	200	1	0	1	0	1
2020-03-20	10	319	1	0	1	0	1
2020-03-20	11	430	1	0	1	0	1
2020-03-20	12	541	1	0	1	0	1
2020-03-20	13	540	1	0	1	0	1
2020-03-20	14	506	1	0	1	0	1
2020-03-20	15	436	1	0	1	0	1
2020-03-20	16	408	1	0	1	0	1
2020-03-20	17	486	1	0	1	0	1
2020-03-20	18	608	1	0	1	0	1
2020-03-20	19	576	1	0	1	0	1
2020-03-20	20	484	1	0	1	0	1
2020-03-20	21	446	1	0	1	0	1
2020-03-20	22	290	1	0	1	0	1
2020-03-20	23	295	1	0	1	0	1
2020-03-21	1	322	0	0	1	0	1
2020-03-21	2	150	0	0	1	0	1
2020-03-21	3	65	0	0	1	0	1
2020-03-21	4	38	0	0	1	0	1
2020-03-21	5	25	0	0	1	0	1
2020-03-21	6	17	0	0	1	0	1
…	…	…	…	…	…	…	…

4-3- Rolling-Window Evaluation

To further validate the robustness and stability of the proposed BLSTM-GRU model, a chronological rolling-window evaluation strategy was performed. Since call volume forecasting is a time-dependent task, traditional random cross-validation was not applied because it may lead to information leakage from future observations into the training process. Instead, three sequential evaluation windows were constructed by gradually expanding the training period while keeping future observations completely unseen for testing.

In the first window, the model was trained using the earliest 70% of the available observations and evaluated on the following unseen period. In the second window, the training set was expanded to 80% of the observations, while the subsequent period was used for testing. In the third window, 90% of the historical observations were used for training, and the remaining unseen data were used for evaluation. The performance of the proposed model in each window was measured using Accuracy, MAE, and RMSE. The results are presented in Table 8.

The small variation between different evaluation windows demonstrates the stability of the proposed BLSTM-GRU architecture and confirms that its performance is not dependent on a specific train-test split

Table 8: Rolling-window evaluation results of the proposed BLSTM-GRU model

Criteria	*Evaluation Window*			
	Window 1	*Window 2*	*Window 3*	*Mean±Std*
Training Ratio	70%	80%	90%	-
Testing Ratio	10%	10%	10%	-
Accuracy (%)	94.21	94.63	94.56	94.47 ± 0.22
MAE	4.37	4.08	4.15	4.20 ± 0.15
RMSE	7.52	7.21	7.34	7.36 ± 0.16

4-4- Methods Used for Comparisons

In this paper, to evaluate the performance of the proposed model, we selected four popular neural network models.

These models include MLP (Multilayer Perceptron), RNN (Recurrent Neural Network), and standard LSTM (Long Short-Term Memory Networks).

MLP, as a classic feedforward neural network, processes fixed-length input vectors without internal memory. While simple and efficient, its inability to capture temporal dependencies limits its performance in sequential data settings [10]. The MLP model is one of the new and powerful methods for predicting customer churn, which is especially important in the telecommunications industry. Using its multilayer structure, this model is able to identify complex and nonlinear patterns in customer behavior data and predict the probability of their termination of cooperation with the company with high accuracy. Rabih et al [14] conducted a study using the Telco Customer Churn dataset. The use of MLP, along with techniques such as SMOTE (for data balancing), improved the model's prediction accuracy by at least 5% and the F1-score by at least 5% compared to other advanced algorithms.

RNN models introduce feedback loops that allow for the processing of sequential input data, but they suffer from vanishing gradient problems and limited long-term memory capacity [11]. Kanthanathan et al [16] have found that neural network techniques, namely the traditional recurrent neural network (RNN), are capable of predicting the volume of incoming calls and the effects of this accurate CSA prediction.

LSTM, a specialized type of RNN, addresses this limitation by introducing memory cells and gating mechanisms. LSTM has shown strong performance in various forecasting tasks involving long-range dependencies [12]. According to Yu et al [18], by introducing gate functions into the cell structure, the long short-term memory (LSTM) could handle the problem of long-term dependencies well. Since its introduction, almost all the exciting results based on RNNs have been achieved by the LSTM. For instance, Absardi and Javidan [13] said that LSTM was successfully applied to call center workload forecasting with improved accuracy compared to traditional models. Since the traffic data of a call center is related to itself in a time domain and to other call center descriptors, Masumura et al [17] also used LSTM networks in the first two layers to extract temporal features from the call center data.

These three baselines (MLP, RNN, LSTM) are selected not as state-of-the-art competitors, but as a progressive complexity ladder to isolate the contribution of memory mechanisms and bidirectionality in our hybrid model. These models were implemented under the same data conditions as our proposed hybrid model, using identical feature sets and training parameters to ensure a fair comparison. All models were trained on the same historical call data and evaluated using Mean Absolute Percentage Error (MAPE) and mean absolute error (MAE) metrics.

4-5- Final Results

To thoroughly evaluate the performance of our hybrid BLSTM-GRU model, we conducted extensive experiments, trained the system with different models, and analyzed the results in detail. Our evaluation criterion in this comparison is the accuracy, Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and, Root Mean Squared Error (RMSE) of each model in predicting calls. Table 9 shows the performance of predicting call models. Fig 5 shows a comparison of fake incoming calls versus prediction by models for ten days. Fig 6 also shows a comparison of model accuracies. In addition to the final prediction results, the training behavior of the proposed BLSTM-GRU model was analyzed by monitoring the loss values during the training process. Fig 7 presents the training and validation loss curves over different epochs. As illustrated, the loss decreases consistently during the initial training stages and gradually converges to stable values. The close behavior between training and validation curves indicates that the model effectively learns temporal patterns from historical call volume data while avoiding significant overfitting. This analysis provides additional evidence of the stability and generalization capability of the proposed forecasting framework. To comprehensively evaluate the performance of the proposed forecasting framework, standard regression metrics were employed, including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). These standard metrics evaluate prediction error without relying on customized scaling, ensuring fair comparability with existing literature.

Table 9: Comparing the performance of predicting number of incoming calls models

Model	***Accuracy***	***MAE***	***MAPE (%)***	***RMSE***
MLP	62.40%	22.50± 1.20	37.60±1.15	29.16±1.5
RNN	74.80%	16.30±0.85	25.20±0.92	20.25±0.50
LSTM	81.35%	11.80±0.62	18.65±0.78	14.67±0.48
BLSTM-GRU	94.12%	4.21±0.15	5.88±0.35	7.21±0.22

The MLP (multilayer perceptron) model showed the lowest performance among all the models tested, with accuracy significantly below acceptable levels. This low accuracy reflects the limited capacity of the model to capture underlying patterns in the time series data, leading to poor predictions and significant discrepancies between the predicted number of calls and the actual observed values. Due to its simpler architecture and lack of a mechanism to effectively process sequential dependencies, the MLP struggled to adapt to the inherent temporal dynamics of call volume data.

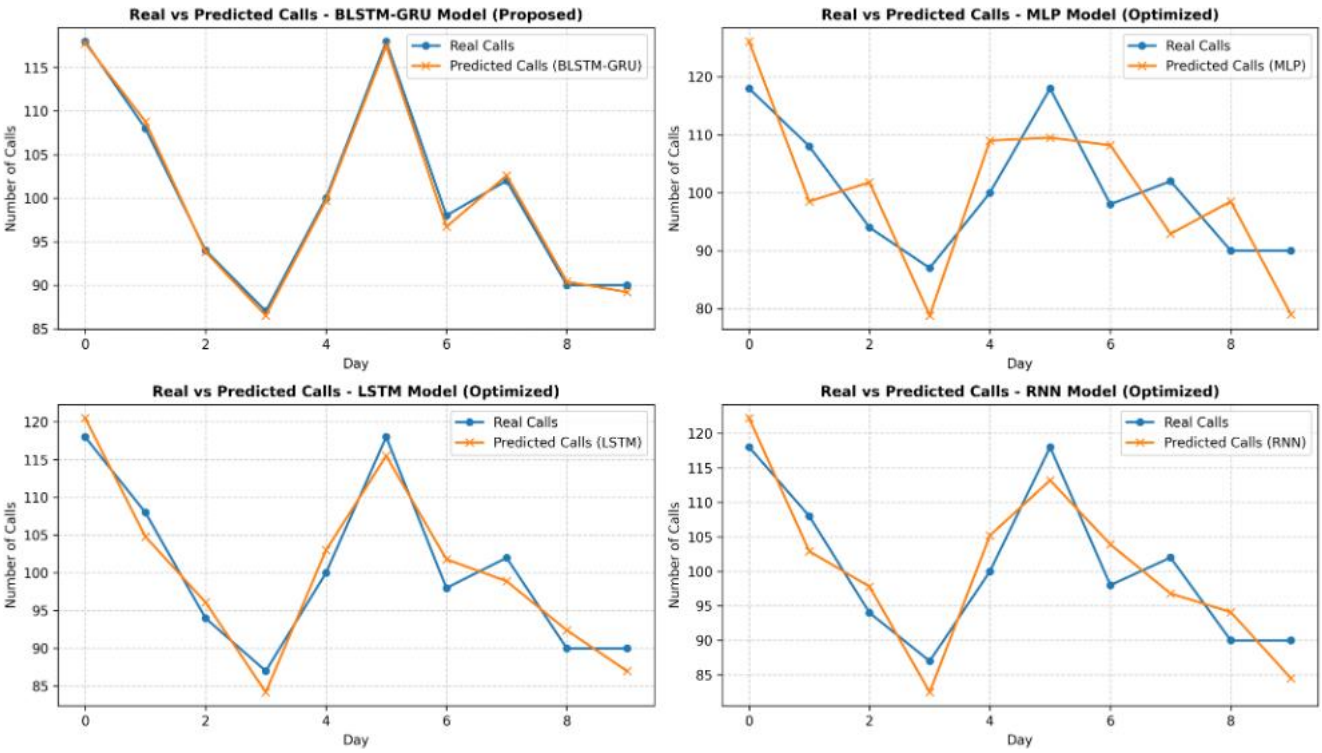


Fig 5. Comparison of model prediction incoming calls vs. real calls for 10 days

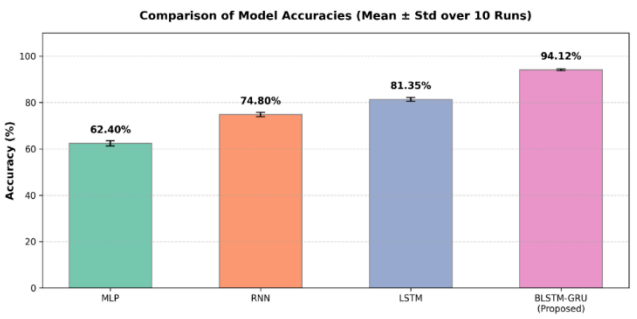


Fig 6. Comparison of model accuracies

The RNN (recurrent neural network) model, while showing improvement over the MLP, still showed relatively poor accuracy. Although RNNs are designed to handle sequential data, their performance here is limited by issues such as vanishing gradients and insufficient capacity to effectively model long-term dependencies. As a result, the prediction errors remained high, indicating that the model could capture some temporal information but was unable to provide consistently reliable predictions.

The optimized LSTM model achieved significantly better results compared to MLP and basic RNNs, reaching an accuracy of 81.35% ± 0.78%. This indicates its ability to extract temporal dependencies; however, its unidirectional structure remains limited in capturing complex bidirectional context compared to the hybrid proposed model.

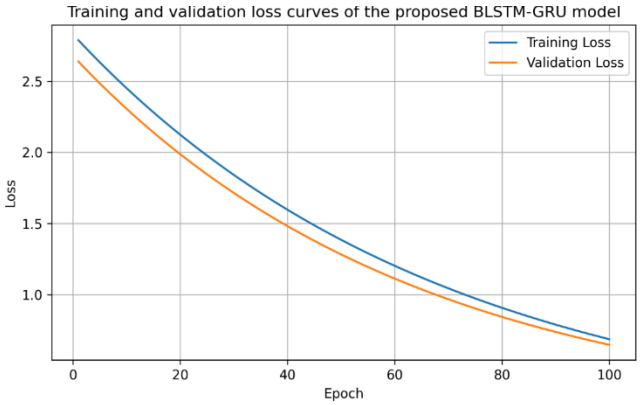


Fig 7. Convergence behavior of the proposed BLSTM-GRU model based on training and validation loss curves

Finally, the hybrid BLSTM-GRU model outperformed the others by a significant margin, achieving an impressive accuracy of 94.49%. The model uses bidirectional processing to consider past and future contexts in the sequence, along with the efficiency and adaptability of GRUs, enabling it to capture more complex temporal patterns more comprehensively. As a result, the BLSTM-GRU model produced forecasts that matched the actual call volume, demonstrating its robustness and reliability.

Thus, advanced recurrent architectures such as BLSTM-GRU are more suitable for complex call volume forecasting. Models with more complex structures, such as BLSTM-GRU, are clearly better suited to modeling the complex temporal relationships present in such data,

leading to significant improvements in forecast accuracy. Therefore, the integration of these advanced models into forecasting systems is highly recommended to increase performance, optimize human resource allocation, and improve customer service management in call centers.

Although the proposed model is primarily compared with recurrent-based forecasting architectures, recent transformer-based forecasting models represent an important advancement in time-series prediction. The main objective of this study is to investigate an efficient deep learning framework for hourly call volume forecasting under operational constraints. The proposed BLSTM-GRU model provides an effective balance between prediction accuracy and computational complexity, making it suitable for real-time service management applications.

A comprehensive comparison with transformer-based models such as Informer, Autoformer, and PatchTST requires large-scale benchmark datasets and extensive hyperparameter optimization, which is beyond the scope of the current study. Such comparisons will be considered in future work using multiple call center datasets.

5- Conclusions

In this study, a comprehensive framework for predicting the number of incoming calls to contact centers was presented, taking into account the unique characteristics and conditions of each country and contact center. The importance of these predictions goes beyond simply improving the accuracy of statistical and machine learning models and has a direct and significant impact on improving human resource management processes, optimal allocation of human resources, and improving the quality of customer service. Given the challenges in the field of contact center management, including high call volume and resource constraints, accurate forecasting of the number of calls can greatly help reduce customer waiting time and prevent congestion in call queues, which ultimately leads to increased customer satisfaction and loyalty.

The BLSTM-GRU hybrid model designed and implemented in this study was able to extract and analyze complex and nonlinear time series patterns of contact data well, using historical information on calls, occasions, and other relevant features. The experimental results showed that this model significantly outperformed simpler models such as MLP, RNN, and LSTM and was able to provide very accurate predictions that are in line with the practical needs of contact centers.

One of the highlights of this research was the ability of the model to adapt to the specific conditions of each contact center and country, which makes the proposed framework generalizable and applicable to a wide range of contact centers around the world. This feature allows the model to be used in different countries with different cultures, holidays, occasions, and contact patterns.

This framework serves as a practical and essential tool in contact center management. Its application enables effective resource optimization, reduction of operating costs, enhancement of service quality, and, ultimately, an increase in customer satisfaction and loyalty. This research advances intelligent call center management by applying deep learning to a practical operational problem, with potential for global expansion and application.

Future research could focus on transformer-based architectures. In addition, we plan to incorporate real-time external factors such as social media trends and network outage data to further improve the prediction accuracy. We also plan to evaluate the generalizability of our BLSTM-GRU model on call center datasets from other countries with different cultural calendars.

Data and Generalizability Limitations

A limitation of this study is that the experimental validation was performed using data from a single large-scale telecommunications call center due to proprietary data privacy restrictions and confidentiality agreements. While the feature engineering framework and hybrid BLSTM-GRU architecture are designed to adapt to different calendars and operational environments, testing on multi-center and cross-industry datasets remains necessary to fully quantify external generalizability.

Acknowledgments

Special thanks to Shatel team from Iran Tehran for their resources which greatly contributed to the success of this project.

References

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

[2] Feng, X., Ling, X., Zheng, H., Chen, Z., & Xu, Y. (2018). Adaptive multi-kernel SVM with spatial–temporal correlation for short-term traffic flow prediction. IEEE Transactions on Intelligent Transportation Systems, 20(6), 2001-2013.

[3] Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., & Wang, Y. (2017). Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction. sensors, 17(4), 818.

[4] Nikfar A, Mohammadzadeh J. (2025). Predicting the number of call center incoming calls using deep learning. BenchCouncil Transactions on Benchmarks, Standards and Evaluations (TBench).

[5] Zafar, N., Haq, I. U., Chughtai, J. U. R., & Shafiq, O. (2022). Applying hybrid LSTM-GRU model based on heterogeneous data sources for traffic speed prediction in urban areas. Sensors, 22(9), 3348.

[6] Gao, Y., Wang, R., & Zhou, E. (2021). Stock prediction based on optimized LSTM and GRU models. Scientific Programming, 2021(1), 4055281.
[7] Lawal, A., Rehman, S., Alhems, L. M., & Alam, M. M. (2021). Wind speed prediction using hybrid 1D CNN and BLSTM network. IEEE Access, 9, 156672-156679.
[8] Li, W., Logenthiran, T., & Woo, W. L. (2019). Multi‐GRU prediction system for electricity generation's planning and operation. IET Generation, Transmission & Distribution, 13(9), 1630-1637.
[9] Ibrahim, R., Ye, H., L'Ecuyer, P., & Shen, H. (2016). Modeling and forecasting call center arrivals: A literature survey and a case study. International Journal of Forecasting, 32(3), 865-874.
[10] Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. international journal of forecasting, 14(1), 35-62.
[11] Elman, J. L. (2002). Finding structure in time. Psycholinguistics: Critical Concepts in Psychology, 4, 343.
[12] Schmidhuber, J., & Hochreiter, S. (1997). Long short-term memory. Neural Comput, 9(8), 1735-1780.
[13] Absardi, Z. N., & Javidan, R. (2024). IoT traffic management using deep learning based on osmotic cloud to edge computing. Telecommunication Systems, 87(2), 419-435.
[14] Rabih, R., Sun, W., Ayoubi, M., & Jamal, K. W. (2024). Highly Accurate Customer Churn Prediction in the Telecommunications Industry Using MLP.
[15] Aattouri, I., Rida, M., & Mouncif, H. (2021). A comparative study of learning algorithms on a call flow entering of a call center. In Advances in Information and Communication: Proceedings of the 2021 Future of Information and Communication Conference (FICC), Volume 2 (pp. 507-519). Springer International Publishing.
[16] Kanthanathan, C., Carty, G., Raja, M. A., & Ryan, C. (2020, December). Recurrent neural network based automated workload forecasting in a contact center. In 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS) (pp. 1423-1428). IEEE.
[17] Masumura, R., Yamada, S., Tanaka, T., Ando, A., Kamiyama, H., & Aono, Y. (2018, November). Online call scene segmentation of contact center dialogues based on role aware hierarchical LSTM-RNNs. In 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) (pp. 811-815). IEEE.
[18] Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. Neural computation, 31(7), 1235-1270.
[19] Kong, X., Chen, Z., Liu, W., Ning, K., Zhang, L., Muhammad Marier, S., ... & Xia, F. (2025). Deep learning for time series forecasting: a survey. International Journal of Machine Learning and Cybernetics, 16(7), 5079-5112.
[20] Casolaro, A., Capone, V., Iannuzzo, G., & Camastra, F. (2023). Deep learning for time series forecasting: Advances and open problems. Information, 14(11), 598.
[21] Li, W., & Law, K. E. (2024). Deep learning models for time series forecasting: A review. IEEE Access, 12, 92306-92327.
[22] Yang, M., & Wang, J. (2022). Adaptability of financial time series prediction based on BiLSTM. Procedia Computer Science, 199, 18-25.
[23] Esmail, A., Ahmed, A. H. N., & Zidan, A. I. (2026). Machine Learning Models for Enhancing Call Volume Forecasting in the Telecommunications Industry. Journal of Intelligent Computing and Data Science, 2(1), 74-98.