frontiers | Frontiers in Communications and Networks

TYPE Original Research
PUBLISHED 24 October 2025
DOI 10.3389/frcmn.2025.1658461

OPEN ACCESS

EDITED BY
Sanjay Dhar Roy,
National Institute of Technology, Durgapur, India

REVIEWED BY
Ahmad Bazzi,
New York University Abu Dhabi, United Arab Emirates
Latifa Guesmi,
Université de Carthage, Tunisia

*CORRESPONDENCE
Israel Tommy,
itommy@pvamu.edu
Taoreed Akinola,
takinola2@pvamu.edu

RECEIVED 02 July 2025
ACCEPTED 25 August 2025
PUBLISHED 24 October 2025

CITATION
Tommy I, Akinola T, Li X and Qian L (2025) Spatio-temporal beam-level traffic forecasting in 5G wireless systems using multi-task learning. *Front. Commun. Netw.* 6:1658461. doi: 10.3389/frcmn.2025.1658461

Spatio-temporal beam-level traffic forecasting in 5G wireless systems using multi-task learning

Israel Tommy*, Taoreed Akinola*, Xiangfang Li and Lijun Qian

CREDIT Center, Department of Electrical and Computer Engineering, Prairie View A&M University, Prairie View, TX, United States

Introduction: Beam-level traffic forecasting plays a vital role in the optimization of 5G networks by enabling proactive resource allocation and congestion control. However, the task is complicated by inherent data sparsity and the presence of multi-scale temporal dynamics, making accurate predictions difficult to achieve using conventional models.
Methods: To address these challenges, we propose a Gated Recurrent Unit (GRU)-based Multi-Task Learning (MTL) framework, enhanced by a weighted ensemble approach. We systematically evaluate the performance of six forecasting models—Linear Regression, DLinear, XGBoost, Echo State Network (ESN), Long Short-Term Memory (LSTM), and GRU-MTL—across three input sequence lengths (168-h, 24-h, and 8-h) using real-world beam-level data from the ITU AI for Good initiative.
Results: Experimental findings reveal that the GRU-MTL model significantly outperforms traditional baselines, achieving a Mean Absolute Error (MAE) of 0.2136 on 168-h sequences compared to LSTM's 0.3223. Long sequences (168-h) reduce MAE by 56% relative to short 8-h windows, effectively mitigating the effects of sparsity. Furthermore, an ensemble of top-performing models (MTL, XGBoost, and Linear Regression) yields additional gains, reducing MAE to 0.2105—a 1.45% improvement over MTL alone.
Discussion: These results highlight the importance of long-term temporal context and model diversity for robust traffic prediction in sparse environments. The proposed framework offers practical guidelines: 168-h forecasting windows are optimal for weekly planning, and model ensembling enhances generalization across varying beam activity levels. This study contributes a scalable and accurate solution for spatio-temporal traffic forecasting in next-generation wireless networks.

KEYWORDS

5G, traffic forecasting, time series prediction, GRU, multi-task learning, LSTM, ESN, DLinear

1 Introduction

The explosive evolution of 5G networks has redefined the wireless communications paradigm and witnessed an exponential surge in mobile data traffic driven by the proliferation of smartphones, IoT devices, and bandwidth-intensive applications, introducing an increasing strain on the available wireless spectrum capacity (Cisco, 2020; Zhang et al., 2023). This increasing demand has not only introduced a suite of challenges that require innovative forecasting techniques, but has also required continuous

advances in wireless communication technologies such as 5G and beyond (ITU Radiocommunication Sector, 2020; 3GPP, 2022).

To meet these growing demands, 5G wireless networks operate across a diverse set of frequency ranges, including sub-6 GHz (FR1) and millimeter-wave (mmWave) bands (FR2, 24–100 GHz), offering abundant spectrum resources that enable multi-gigabit-per-second data rates and ultra-low latency (Rappaport et al., 2013). However, mmWave signals suffer from high path loss and susceptibility to blockages, which require advanced techniques such as beamforming and massive MIMO to focus energy directionally and enhance coverage (Rohde and Poddar, 2018). Recently, the upper midband spectrum—also known as FR3, typically spanning 7–24 GHz—has emerged as a promising candidate for 6G due to its favorable trade-off between capacity and coverage (Giordani et al., 2020). This band benefits from improved propagation compared to mmWave while still offering wider bandwidth than sub-6 GHz, making it well-suited for mobile broadband and edge intelligence applications (Alsabah et al., 2023). In addition, 6G-customized beamforming strategies, such as outage-based beamforming, are being developed to improve link reliability under dynamic conditions and harsh propagation environments (Alrabeiah and Alkhateeb, 2022). These technologies not only improve spectral efficiency but also support massive device connectivity and enable emerging applications such as smart cities, immersive media, and real-time industrial IoT (Agiwal et al., 2016).

A key challenge in 5G network management is ensuring optimal resource allocation to maintain high Quality of Service (QoS) in increasingly dense and heterogeneous network environments (Zhang et al., 2023). Efficient traffic forecasting enables proactive network optimization, minimizing congestion, and ensuring dynamic bandwidth allocation (Wu et al., 2020). However, traffic forecasting in 5G networks is significantly more complex than in previous generations due to the increased granularity of network management, particularly at the beam level. Unlike traditional macro cell-based forecasting, beam-level forecasting requires capturing highly localized and dynamic user activity, making it a crucial but difficult task (Rappaport et al., 2019). In addition, the traditional methods struggle to handle traffic irregularities such as intermittent zeros, short time-series lengths, and multivariate dependencies. This study tackles these limitations by introducing Gated Recurrent Unit (GRU) in multi-task learning (MTL) framework capable of learning shared representations across multiple prediction tasks. The proposed approach leverages state-of-the-art (SOTA) forecasting models designed to handle complex multivariate time-series data, enhancing predictive accuracy and real-time adaptability (Siami-Namini et al., 2018).

1.1 Traffic data collection in 5G wireless system

As part of global efforts to advance AI-driven solutions for sustainable development, the AI for Good initiative, in collaboration with the International Telecommunication Union (ITU), launched a challenge in spatio-temporal load forecasting in 5G wireless systems. The challenge focuses on predicting beam-level wireless traffic, a critical component in enhancing network resource allocation and ensuring efficient network operations (AIforGood-ITU, 2024). (The data collection process is given in Figure 1).

To support this initiative, the ITU released high-resolution beam-level throughput datasets with precise hourly granularity, providing critical network performance metrics, including throughput volume, throughput time, physical resource block (PRB) utilization, and user count. Each of the four datasets (throughput volume, throughput time, physical resource block (PRB) utilization, and user count) comprises:

- 30 base stations (BSs), each containing 3 cells, with each cell consisting of 32 beams, resulting in a total of 2,880 beams.
- Hourly recordings over 5 weeks, amounting to 2,419,200 samples for each of the four datasets, making it one of the most detailed public datasets available for 5G network traffic forecasting.
- Hierarchical segmentation, enabling granular traffic flow analysis at different levels of the network infrastructure.

These extensive datasets provides a valuable foundation for exploring forecasting strategies, allowing researchers to develop models capable of capturing intricate spatial and temporal variations in network traffic. Figures 2–4 show examples of the four different time-series data for 5 weeks at different base stations, beams, and cells.

To gain a deeper understanding of the deficiency inherent in beam-level traffic patterns, we analyzed the distribution of zero-valued entries across both temporal (840 hourly samples) and spatial

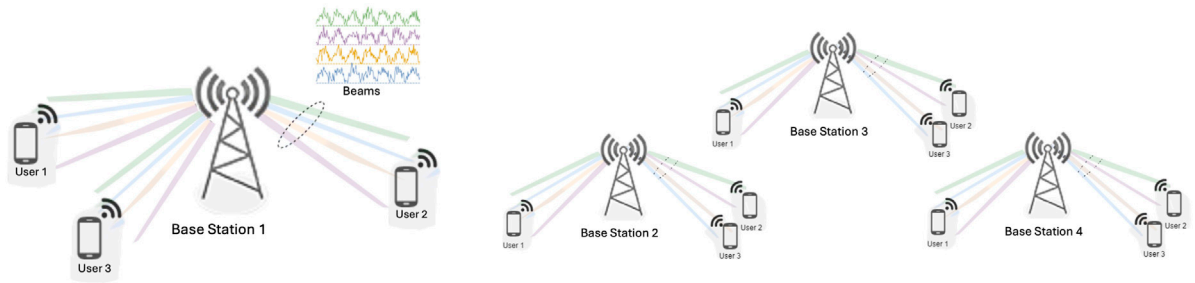

FIGURE 1
Spatio-Temporal Beam-Level Traffic Data Collection in 5G Wireless Systems: 30 Base Stations (BSs), 3 users per BS, and 32 beams per user. Data is collected hourly for 5 weeks (AIforGood-ITU, 2024).

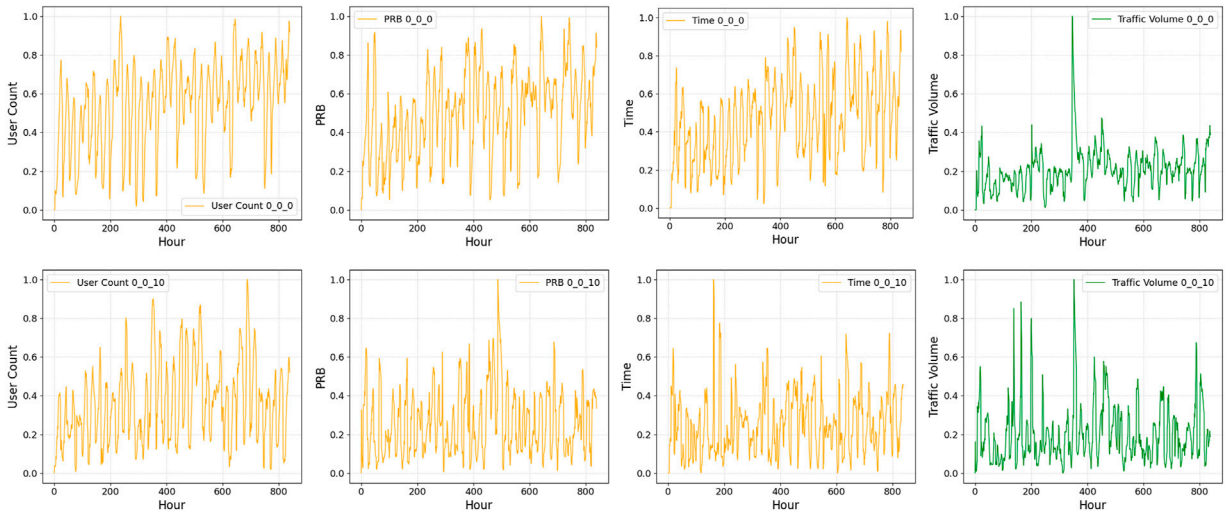

FIGURE 2
Sample of the four datasets at a different base stations, cells and beams.

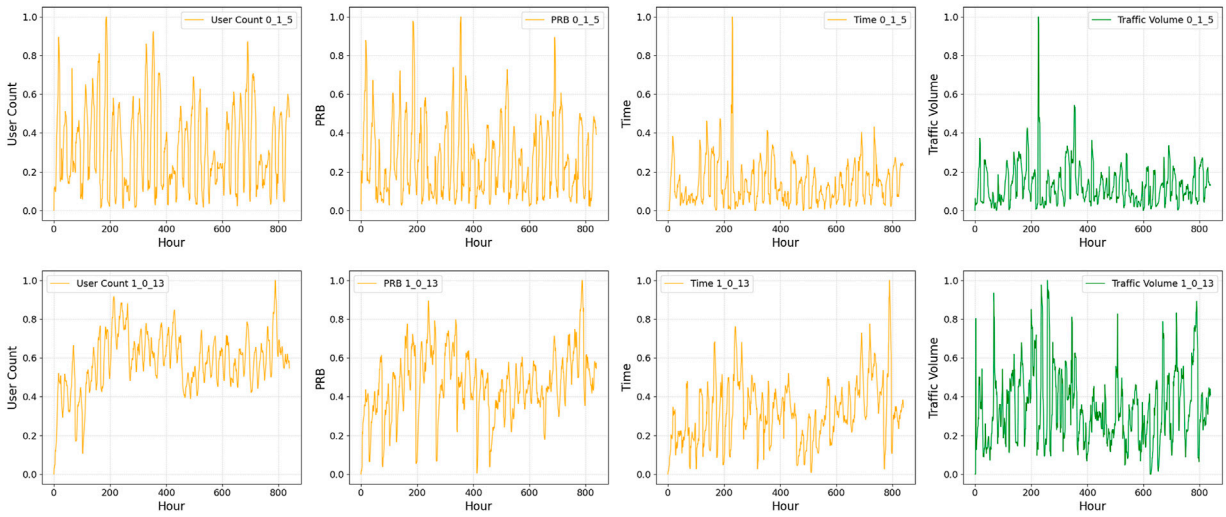

FIGURE 3
Sample of the four datasets at a different base stations, cells and beams.

(2,880 beams) dimensions. These zero entries represent periods or locations of beam inactivity where no user data traffic was recorded. An initial analysis, illustrated in the bar chart of Figure 5a, reveals the average spatial sparsity per time sample. On average, each sample comprises approximately 915 zero-valued and 1965 non-zero-valued beams, indicating a mean sparsity level of approximately 32%. This reflects a consistent pattern of intermittent beam activity at any given time step. Furthermore, we examined the temporal evolution of this deficiency, as depicted in the line plot of Figure 5b. The percentage of inactive beams per sample fluctuates considerably over the 840-h period, with values ranging from approximately 20% to over 40%. This dynamic trend highlights the significant temporal variability in network utilization, likely attributable to factors such as diurnal usage cycles, user mobility patterns, and heterogeneous service demands. These findings underscore a critical modeling challenge: the data is fundamentally sparse and dynamically so. This necessitates the development of forecasting models, such as scarcity-aware or multi-task architectures, that can explicitly account for inactive periods to improve predictive accuracy and avoid biases introduced by zero-inflated data.

1.2 The problem statement

The rapid expansion of 5G wireless networks, driven by unprecedented growth in mobile devices, IoT applications, and bandwidth-demanding services, has placed increasing

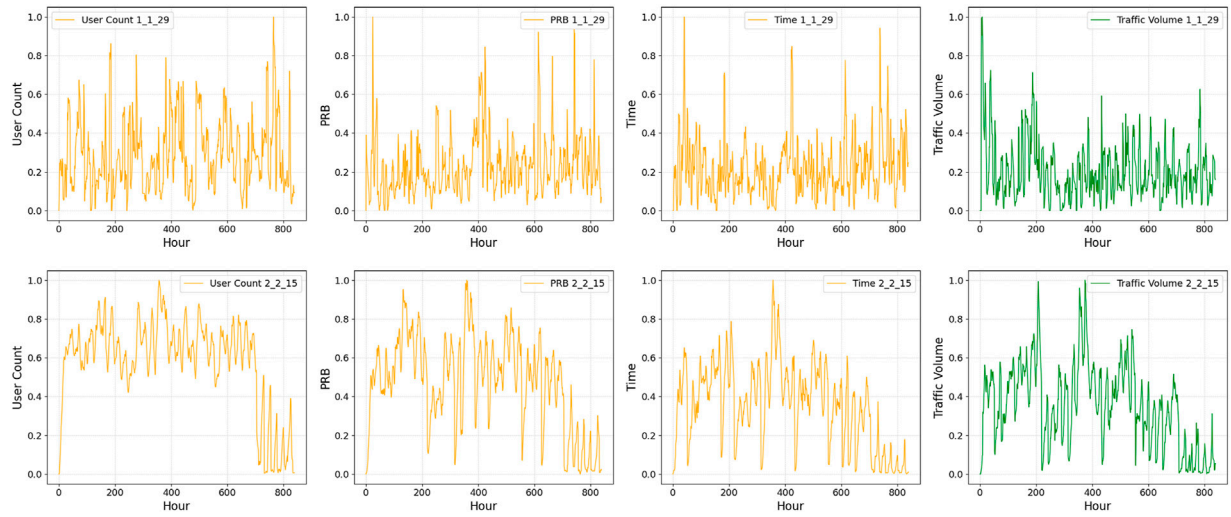

FIGURE 4
Sample of the four datasets at a different base stations, cells and beams.

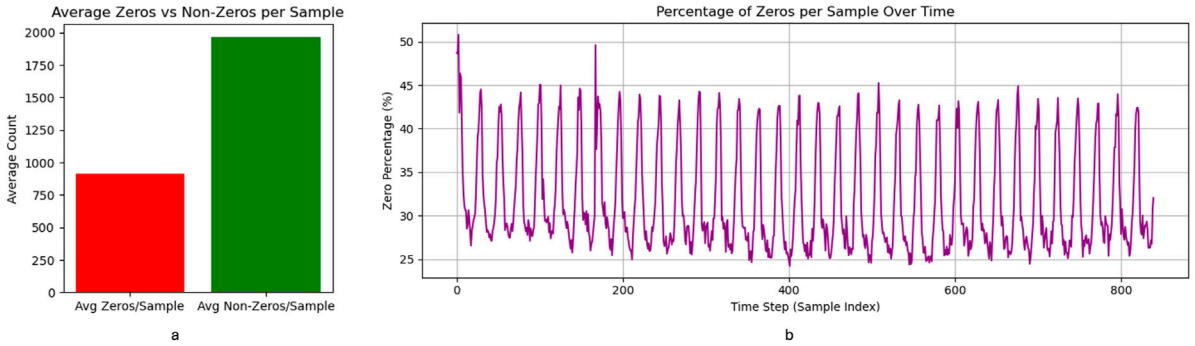

FIGURE 5
Traffic zero and non-zero data distribution. **(a)** Average Zero vs Non-Zero Counts per Sample. **(b)** Temporal Distribution of Zero Entries Across Samples.

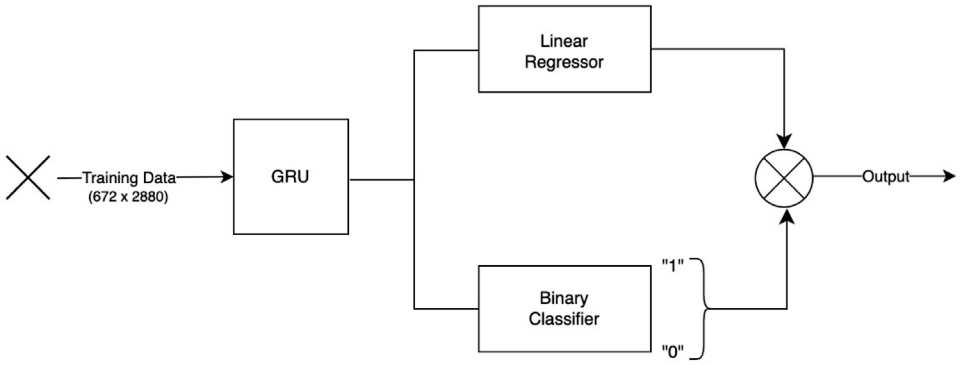

FIGURE 6
Gru integrated multi-task learning framework.

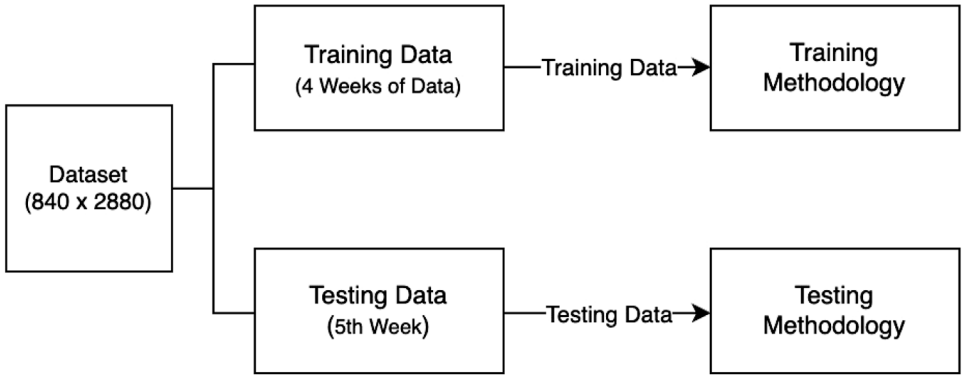

FIGURE 7
Training and testing data block diagram.

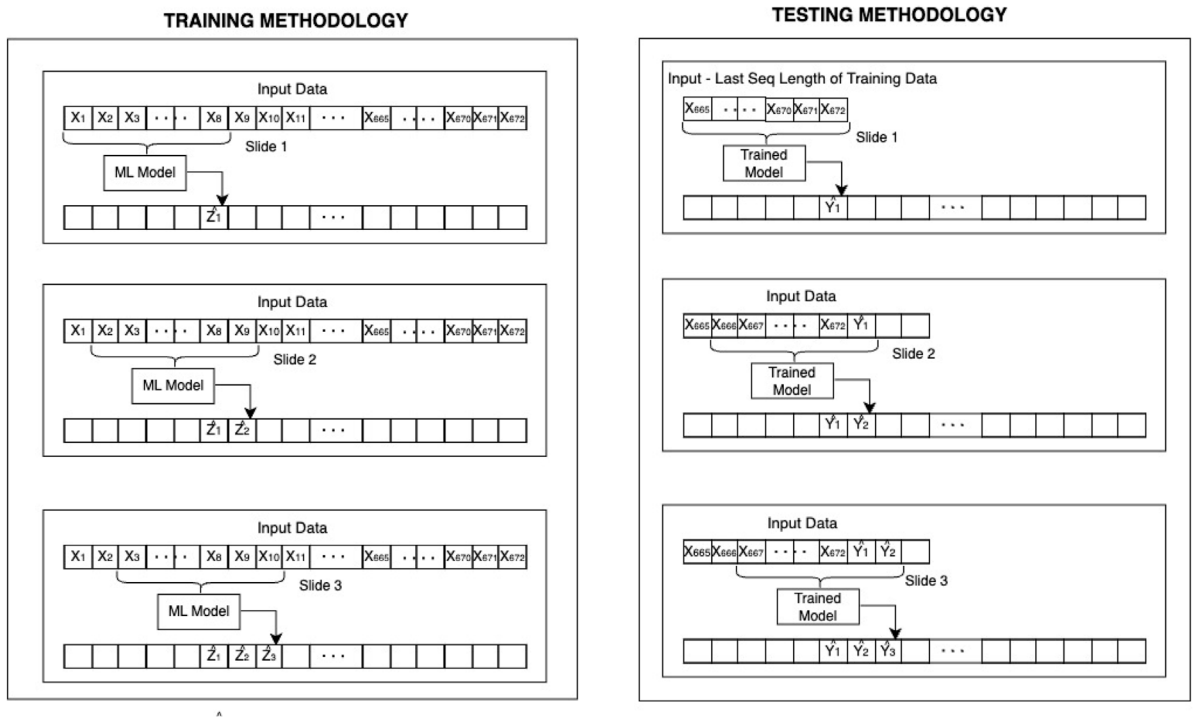

FIGURE 8
Sliding window strategy used in training and testing. Left: The GRU-based model is trained with incremental target sequences. Right: The pre-trained model recursively predicts future values during testing.

pressure on the limited wireless spectrum and necessitated significant advancements in how traffic is managed and forecasted. Unlike traditional macro cell-level prediction, the introduction of beamforming in 5G networks has shifted traffic management to the beam level, where user activity is highly localized, dynamic, and inherently more complex to predict. Accurate beam-level traffic forecasting has therefore become critical for enabling optimal resource allocation, dynamic bandwidth management, congestion mitigation, and the delivery of consistent Quality of Service (QoS) in modern wireless systems.

However, this finer spatial granularity introduces several persistent challenges that conventional statistical and early machine learning methods struggle to address. These include the prevalence of intermittent zeros in beam-level datasets—periods of low or no traffic—which can distort predictions when not handled properly. Moreover, the typically short time-series length for individual beams limits the model's ability to learn long-term patterns, further complicating forecasting tasks. In addition, the multivariate nature of network performance metrics adds a layer of complexity, requiring models to capture intricate dependencies among multiple correlated features. Finally, practical deployment

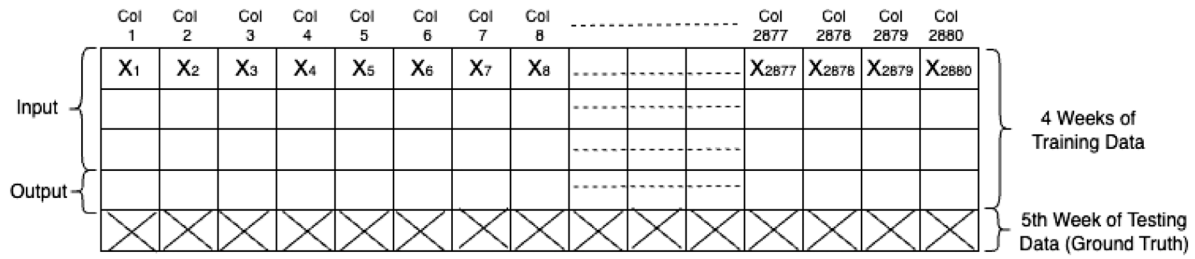

FIGURE 9
Matrix representation of the 5-week dataset. The first 4 weeks serve as training input; the fifth week serves as testing ground truth for performance evaluation.

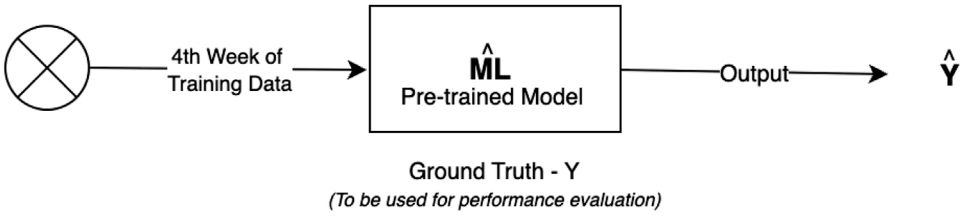

FIGURE 10
Block diagram of the testing setup: the final week of training is used as input to the pre-trained model, and the outputs are evaluated against the fifth week of data.

TABLE 1 Hyperparameters for the different models.

Hyperparameters	Linear regression	DLinear	XGBoost	ESN	LSTM	MTL
Input Sequence Length	168, 24, 8	168, 24, 8	168, 24, 8	168, 24, 8	168, 24, 8	168, 24, 8
Hidden Dimension	1,024	1,024	1,024	1,024	1,024	1,024
Network Architecture	GRU + Linear	GRU + Linear	GRU + XGBoost	ESN core	LSTM + Linear	GRU, Linear, Classifier
Activation Function	ReLU	ReLu	-	tanh	tanh	ReLU/Linear
Optimizer	Adam	Adam	Adam	Adam	Adam	Adam
Learning Rate	0.01	0.0001	0.1	0.000001	0.0001	0.001
Batch Size	128	128	128	128	128	128
Epochs	1,500	500	1,500	5,000	1,500	1,500
Evaluation Metrics	MAE, MSE, RMSE	MAE, MSE, RMSE	MAE, MSE, RMSE	MAE, MSE, RMSE	MAE, MSE, RMSE	MAE, MSE, RMSE

constraints in 5G networks demand forecasting models that are not only accurate but also lightweight and efficient enough for real-time use at the network edge.

These challenges highlight a clear gap: while deep learning methods such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks have shown promise in general traffic forecasting, they often remain inadequate for beam-level forecasting where sparsity, short time spans, and multivariate relationships must be tackled simultaneously. This thesis addresses this gap by investigating the use of Gated Recurrent Units (GRUs) within a Multi-Task Learning (MTL) framework. The goal is to design a forecasting approach that can learn shared representations across multiple prediction tasks—such as traffic volume regression and active/inactive beam classification—while remaining computationally efficient for real-time operation. In doing so, this research aims to advance the development of robust, scalable, and adaptive forecasting models that can meet the stringent requirements of next-generation wireless networks.

Spatio-temporal beam-level forecasting in 5G networks requires models that can effectively handle sparse, irregular, and highly dynamic traffic patterns while remaining efficient enough for real-time deployment. Conventional forecasting

TABLE 2 Performance metrics for different models and sequence lengths.

Metrics	Sequence length	Linear regression	DLinear	XGBoost	ESN	LSTM	MTL
MAE	168	0.218503	0.238085	0.230860	0.264141	0.355919	0.213631
	24	0.239661	0.286313	0.225731	0.264349	0.301782	0.300096
	8	0.277397	0.274060	0.227612	0.268062	0.316045	0.282097
MSE	168	0.273377	0.324395	0.299680	0.389497	0.743190	0.249026
	24	0.361508	0.550637	0.291654	0.367124	0.503964	0.733608
	8	0.479330	0.398813	0.270044	0.383942	0.587777	0.494261
RMSE	168	0.522855	0.569557	0.547430	0.624097	0.862085	0.499025
	24	0.601255	0.742049	0.540050	0.605908	0.709904	0.856509
	8	0.692336	0.631516	0.519658	0.619631	0.766666	0.703037

TABLE 3 Base line model.

Target	Hist. Avg	iTransformer	PatchTST	DLinear	Transformer
Test 1	0.2108	0.1967	0.1973	0.2005	0.2166
Test 2	0.2431	0.2348	0.2343	0.2352	0.2331

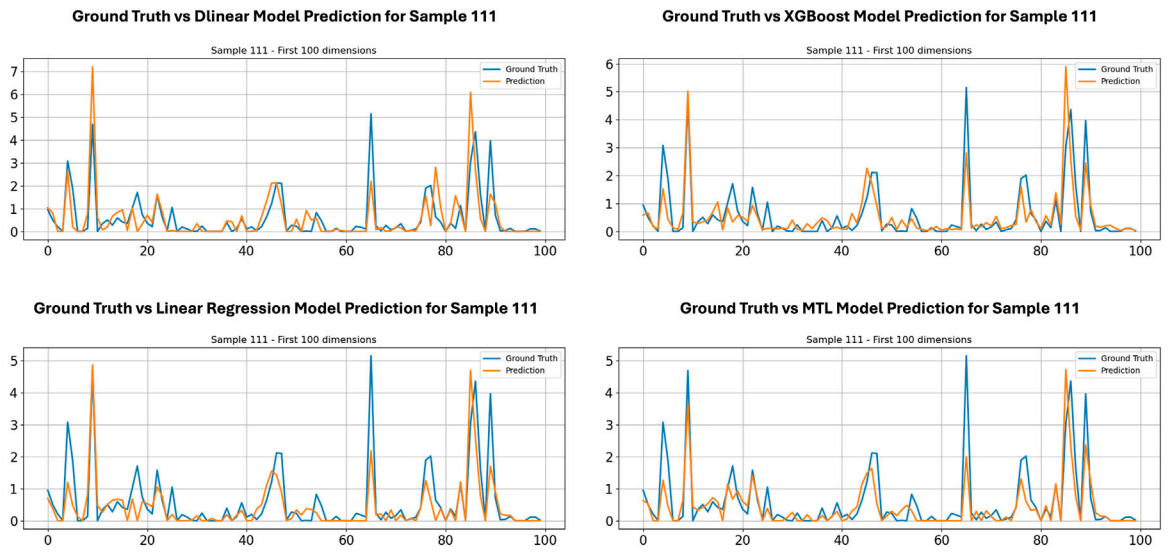

FIGURE 11
Ground truth vs. model prediction across various models-1.

methods—including statistical models, traditional machine learning, and even some deep learning architectures—struggle with this problem because they either fail to capture long-range temporal dependencies, are sensitive to noise and missing data, or impose heavy computational costs.

Gated Recurrent Units (GRUs) directly address these limitations, making them particularly well-suited to outperform other models for beam-level traffic prediction. Their gating mechanisms selectively retain relevant historical information while filtering out noise, enabling them to model complex non-linear temporal relationships even across long gaps with zero traffic. Unlike attention-based or Transformer models, which require large datasets to generalize effectively, GRUs generalize well with limited time-series data—a common scenario in beam-level measurements. Moreover, they achieve performance comparable to or better than Long Short-Term Memory (LSTM) networks but with fewer parameters, resulting in lower latency and reduced computational overhead—essential for real-time 5G network optimization.

Beam-level forecasting also presents several specific challenges that further justify the use of GRUs:

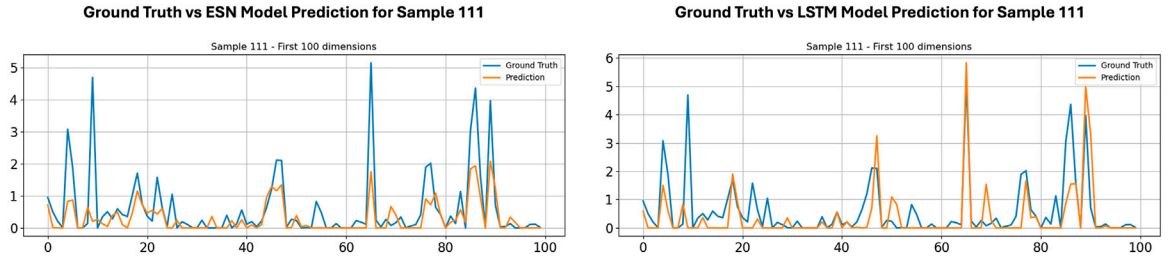

FIGURE 12
Ground truth vs. model prediction across various models-2.

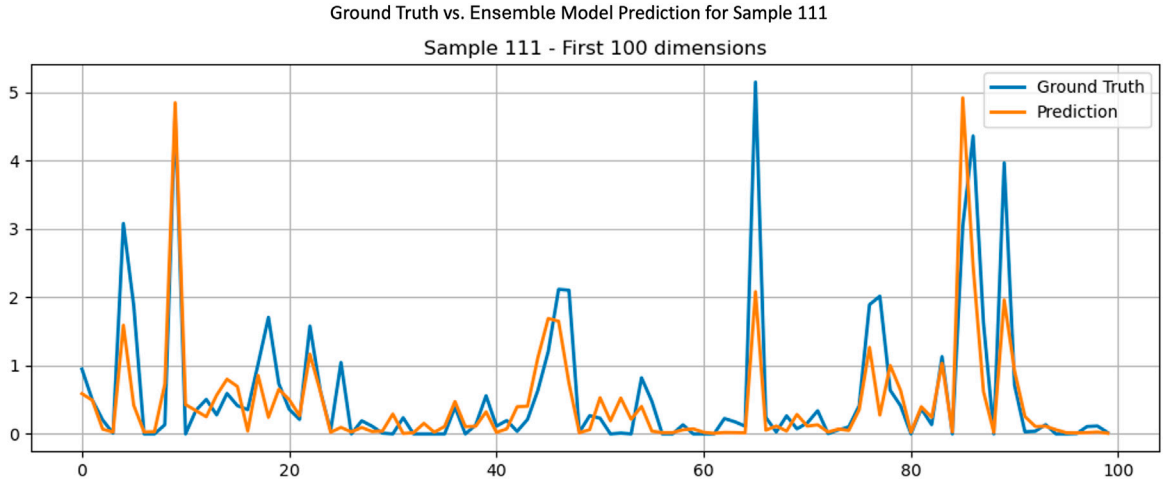

FIGURE 13
Ground truth vs. ensemble prediction.

- Handling Intermittent Zeros in the Dataset: Intermittent zero values occur during low or no traffic periods, skewing prediction outcomes. Effectively managing these zeros is crucial for maintaining accurate forecasts.
- Limited Time-Series Length: The relatively short time-series data complicates the capture of long-term patterns, posing a challenge for large-scale forecasting models that require extensive data for effective training.
- Handling Multivariate Data: The dataset's multiple network performance metrics introduce additional complexity. Accurately modeling multivariate dependencies is essential for capturing the full scope of traffic influences.
- Requirement for Real-Time, Lightweight Models: Real-time forecasting necessitates models that are both accurate and computationally efficient. Many emerging deep learning models are resource-intensive, making lightweight and optimized models critical for timely forecasts.

By aligning these strengths with the unique requirements of beam-level forecasting, the proposed GRU-based approaches—including a Multi-Task Learning framework and an ensemble strategy—are explicitly designed to outperform conventional forecasting techniques in capturing the spatio-temporal complexity of 5G traffic. (Fu et al., 2016).

Furthermore, a principal contribution of this research is the development and integration of a Multi-task Learning (MTL) paradigm. The MTL framework explicitly addresses the multivariate characteristics of beam traffic data by enabling the model to learn shared latent representations and exploit correlations across related variables (e.g., user count, PRB utilization, throughput traffic and time), thereby fostering model generalization, reduce overfitting and increasing more accurate and responsive traffic forecasts. Our research provides several key contributions to the field of spatio-temporal beam-level traffic forecasting in 5G wireless systems:

- Effective Modeling of Intermittent Data with GRUs: This research demonstrates the inherent capability of the GRU architecture to effectively model raw time-series data containing intermittent zeros without requiring complex, model-agnostic pre-processing steps. The GRU's gating mechanism learns to distinguish between meaningful periods of network inactivity and actual traffic patterns. Instead of disregarding zero-value periods, the model learns from them as part of the temporal sequence,

TABLE 4 Effect of Varying Classification and Regression Loss Weights on Multi-task Model Performance. Best results are bolded.

Configuration	MAE	MSE	RMSE
Classification-only ($\lambda_{cls} = 1$, $\lambda_{reg} = 0$)	0.398	0.667	0.816
Regression-only ($\lambda_{cls} = 0$, $\lambda_{reg} = 1$)	0.257	0.405	0.636
$\lambda_{cls} = 0.1$ ($\lambda_{reg} = 1$)	0.218	0.272	0.521
$\lambda_{cls} = 0.5$ ($\lambda_{reg} = 1$)	0.219	0.272	0.521
Balanced Multitask ($\lambda_{cls} = 0.5$, $\lambda_{reg} = 0.5$)	**0.213**	**0.249**	**0.499**
$\lambda_{cls} = 2.0$ ($\lambda_{reg} = 1$)	0.224	0.281	0.531

TABLE 5 Mean absolute error of the ensemble model.

Sequence_Length	MAE	MSE	RMSE
168 (without ensemble)	0.218503	0.249026	0.499025
168 (with ensemble)	0.210520	0.246709	0.496698

allowing it to accurately capture the sporadic nature of beam-level traffic and retain crucial information about demand intervals.

- GRU-based Multi-task Learning for Enhanced Prediction Accuracy: This research further introduces a novel multi-task learning framework that leverages a GRU core. This framework is designed to simultaneously perform two correlated tasks: forecasting traffic magnitude and classifying demand occurrence (i.e., zero vs. non-zero traffic states). By learning these tasks in parallel, the model capitalizes on shared representations, which enhances generalization and mitigates overfitting. This dual-objective approach significantly improves the accuracy of the forecast, particularly for datasets characterized by the high variability and deficiency common to network traffic.
- Optimization of Input Sequence Length for Improved Performance: This study systematically investigates the impact of input sequence length on predictive accuracy within our MTL framework. Three strategically chosen temporal windows were examine: 168 h (weekly cycles), 24 h (diurnal patterns), and 8 h (short-term activity segments). These intervals were selected to simulate human-like interactions with the data. The findings reveal that longer sequence lengths yield superior forecasting performance compared to shorter ones. The 168-h window's performance advantage confirms that beam-level forecasting benefits from an extended historical context when modeling sparse events. With more than 31.8% zero-inflated beams, longer sequences provide sufficient active samples to distinguish true inactivity from measurement noise, a known challenge in cellular traffic prediction.

These contributions advance the methodologies for spatio-temporal traffic analysis, offering a robust and accurate framework for enhancing network performance and resource management in 5G and beyond.

2 Related works

Performing accurate and timely network traffic forecasting has long been a critical area of research in telecommunication systems, driven by the need for efficient resource allocation, congestion control, and proactive network management. Early efforts primarily focused on macroscopic network traffic, employing traditional statistical time-series models due to their simplicity and interpretability.

One of the foundational approaches involved Autoregressive Integrated Moving Average (ARIMA) models and their variants, widely applied for their ability to capture temporal dependencies in network traffic (Brockwell and Davis, 2002). For instance, Box and Jenkins' methodology (Box et al., 2015) provided a robust framework for modeling and forecasting time series data, which was subsequently adapted for internet traffic prediction. Exponential smoothing methods also found application in forecasting network loads, offering adaptive mechanisms for capturing trends and seasonality (Hyndman and Athanasopoulos, 2018). While effective for aggregated traffic, these statistical models often struggled with the inherent non-linearity, high variability, and complex long-range dependencies characteristic of modern communication networks.

As networks evolved from fixed-line to mobile cellular systems (e.g., 2G, 3G, and early 4G deployments), the complexity of traffic patterns increased, necessitating more sophisticated forecasting techniques. Machine learning (ML) models began to emerge as promising alternatives to traditional statistical methods due to their ability to learn complex non-linear relationships from data. Support Vector Machines (SVMs) were explored for their robustness to noisy data and ability to handle high-dimensional features, demonstrating effectiveness in predicting network congestion and traffic volume (Wang et al., 2011). More recently, the advent of deep learning has revolutionized network traffic forecasting, with models such as Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs) proving highly effective in capturing long-term temporal dependencies and non-linear patterns in network traffic data (Rau et al., 2023; Ramakrishnan and Soni, 2018). Convolutional Neural Networks (CNNs) have also been increasingly employed to extract hierarchical features from traffic data, demonstrating strong performance in various network prediction tasks [(Yu et al., 2017)].

The growing recognition of spatial correlations in network traffic has led to extensive research in spatio-temporal forecasting, especially with the proliferation of dense wireless deployments. Unlike earlier approaches that often combined spatial and temporal models in a decoupled manner, modern deep learning frameworks, particularly Graph Neural Networks (GNNs), have become instrumental in explicitly modeling intricate spatio-temporal dependencies. These models leverage the topological structure of networks to capture complex spatial relationships, while integrated recurrent or convolutional layers handle temporal dynamics, leading to significant advancements in forecasting accuracy for large-scale and complex network environments, including cellular networks (Yu et al., 2017; Wu et al., 2024; Wang et al., 2025).

Multi-Task Learning (MTL), a paradigm where multiple related learning tasks are solved jointly to leverage commonalities and improve generalization performance, has gained significant

traction in network traffic forecasting since 2015. This approach is particularly beneficial for complex and heterogeneous network environments, such as 5G, where various prediction tasks (e.g., traffic volume, throughput, and latency across different beams or cells) are inherently related. By learning shared representations across these tasks, MTL frameworks can overcome challenges like intermittent data patterns and limited time-series lengths for individual tasks, leading to improved prediction accuracy and robustness compared to single-task learning models (Wang et al., 2022; Liu et al., 2024). For instance, recent works have explored MTL to predict citywide cellular network traffic across diverse services, demonstrating its ability to capture complex spatio-temporal fluctuations by sharing knowledge among tasks (Sun et al., 2022).

3 Proposed methods

The significant number of zero-value observations renders traditional forecasting models ineffective. To overcome this limitation, we employ a Gated Recurrent Unit (GRU), a sophisticated deep learning model. Unlike statistical methods that require decomposing the series, the GRU learns directly from the raw, intermittent data. Its gating mechanism dynamically manages the information flow, enabling it to capture complex, non-linear temporal relationships between past events and future demand, even when they are separated by numerous zero-value periods. This makes the GRU a powerful and flexible tool for forecasting sporadic data patterns.

3.1 Handling intermittent data with GRU

Intermittent time series, characterized by sporadic non-zero observations amidst prolonged periods of zeros, present significant challenges for forecasting models. Traditional statistical methods often fail to adequately capture the underlying patterns in such data due to their reliance on assumptions of continuity and uniform variance (Hyndman and Athanasopoulos, 2018). Deep learning approaches, particularly Gated Recurrent Units (GRUs), have emerged as a promising alternative by leveraging sequential learning to model complex temporal dependencies, including intermittent demand structures (Lai et al., 2018).

Classical approaches to intermittent demand forecasting typically fall into three categories: exponential smoothing variants and probability-based methods. Simple exponential smoothing (SES) applies uniform weighting to all observations, including zeros, resulting in biased forecasts when demand is sporadic (Gardner, 2006). Modified exponential smoothing techniques, such as the TSB method (Teunter et al., 2011), improve upon SES by incorporating demand probability estimates, but they still rely on heuristic adjustments rather than data-driven learning. Probability-based methods, such as zero-inflated Poisson regression (Lambert, 1992), explicitly account for excess zeros but are limited in their ability to capture evolving temporal dynamics.

GRUs, a variant of recurrent neural networks (RNNs), address many of the limitations of traditional methods through their gated architecture. The update and reset gates allow GRUs to dynamically modulate information flow, effectively learning when to retain or discard historical observations (Cho et al., 2014). This mechanism is particularly advantageous for intermittent data, as the model can suppress irrelevant zeros while amplifying meaningful non-zero events. Recent hybrid approaches, such as (Silveira Gontijo and Azevedo Costa, 2020), demonstrate that neural networks can effectively model hierarchical and intermittent structures in demand forecasting, outperforming traditional statistical methods. Furthermore, GRUs can model long-term dependencies, distinguishing between true inactivity (structural zeros) and transient fluctuations (noise), a capability that eludes most statistical approaches (Lim and Zohren, 2021).

GRUs offer a flexible and powerful framework for forecasting intermittent time series, overcoming key limitations of traditional statistical methods. Their ability to learn complex temporal dependencies without manual feature engineering makes them particularly suited for applications with sporadic demand patterns.

3.2 Proposed models

This study introduces a suite of hybrid deep learning architectures tailored to the complex nature of beam-level traffic volume forecasting in 5G networks. Beam-level traffic in 5G exhibits high temporal volatility, spatial sparsity, and a mixture of periodic and aperiodic patterns. These characteristics demand a modeling framework capable of capturing both long-term temporal dependencies and nonlinear fluctuations. To this end, we propose and comparatively evaluate seven GRU-based hybrid models: GRU-Linear, GRU-DLinear, GRU-XGBoost, ESN, LSTM, GRU-MTL, and a GRU-based Ensemble Model.

The rationale for deploying this diverse set of models lies in their complementary strengths. Linear regressors offer transparency and serve as strong baselines. DLinear enhances trend/seasonality decomposition, while XGBoost captures feature interactions missed by standard neural nets. ESNs contribute fast training and memory-rich transformations, and the LSTM-FCN hybrid improves temporal context learning. The Multi-task Learning model adds robustness through joint optimization of classification and regression objectives, and the ensemble aggregates model strengths to reduce variance and improve generalization. This model diversity ensures that both short-term spikes and long-term trends in traffic dynamics are effectively captured.

In addition, the design choice not to use the GRU as input to the Echo State Network (ESN) was motivated by the inherent architectural properties of the ESN. Unlike XGBoost, which is a gradient-boosted decision tree model that benefits from a compact and informative feature representation (in this case, GRU embeddings), the ESN itself is a reservoir computing model that naturally performs its own nonlinear feature transformation through its high-dimensional dynamic reservoir states. Feeding GRU embeddings into the ESN would have overridden its core mechanism of projecting input sequences into a rich dynamic state space and could potentially lead to redundant feature processing or overfitting. To ensure a fair comparison, we configured the ESN with a sufficiently large reservoir size and spectral radius, allowing it to internally capture temporal dependencies from the raw input

sequence without the need for an external embedding layer. This design aligns with standard ESN usage, where raw time-series inputs are projected directly into the reservoir state space for subsequent linear readout.

3.2.1 GRU with linear regression (GRU-linear)

In this configuration, GRU encodes the temporal sequence into latent representations, which are then mapped to outputs via a linear regression layer. The model offers a simple yet effective architecture for modeling sequential data where the nonlinearities primarily reside in the temporal dimension rather than the output mapping. This structure has been employed in real-time traffic forecasting settings with notable success (Fu et al., 2016).

3.2.2 GRU with DLinear (GRU-DLinear)

The GRU-DLinear model integrates the DLinear architecture, which decomposes time-series signals into seasonal and trend components before applying separate linear forecasts (Zeng et al., 2023). The GRU pre-processes the sequence, providing rich temporal embeddings that DLinear uses to conduct more interpretable and accurate predictions, especially for long-horizon forecasting with periodic behaviors.

3.2.3 GRU with XGBoost regression (GRU-XGBoost)

In this hybrid, GRU encodes sequential features which are then passed to an XGBoost regressor. XGBoost, known for its strong performance on structured data and ability to model complex feature interactions, serves to refine GRU's temporal outputs by capturing residual nonlinear relationships (Chen and Guestrin, 2016).

3.2.4 GRU with echo state network (GRU-ESN)

The GRU-ESN architecture exploits the high-dimensional memory capabilities of Echo State Networks, a class of reservoir computing models. GRU sequences are passed to an untrained recurrent reservoir with fixed weights, while only the readout layer is trained. This structure introduces additional temporal richness while retaining training efficiency (Gallicchio et al., 2018).

3.2.5 LSTM with fully connected network (LSTM-FCN)

The LSTM-FCN architecture integrates a Long Short-Term Memory (LSTM) network with a Fully Connected Network (FCN) to leverage both temporal sequence modeling and powerful nonlinear feature transformation. In this setup, the LSTM layer learns temporal dependencies and encodes sequential patterns present in the beam-level traffic data, while the subsequent FCN maps these learned representations to the final traffic volume predictions. This hybrid design combines the LSTM's robust gating mechanisms, which capture long-term temporal dynamics, with the FCN's capacity for flexible nonlinear regression. As a result, the LSTM-FCN model provides enhanced adaptability to the irregular and bursty traffic patterns typical of 5G beam-level forecasts (Greff et al., 2016).

3.2.6 Gated recurrent unit-multi-task learning (GRU-MTL)

The GRU-MTL architecture addresses two concurrent tasks: (1) classifying whether traffic volume is active (non-zero), and (2) regressing the actual traffic magnitude. By learning these tasks jointly with shared GRU encoders and distinct output heads, the model benefits from inductive transfer, improving generalization and robustness, especially in sparse or imbalanced traffic conditions (Ruder, 2017). Multi-task learning has shown effectiveness in related spatio-temporal forecasting domains (Chen et al., 2020).

3.2.7 GRU-based ensemble model

Finally, a GRU-based Ensemble Model is constructed by aggregating the predictions of the aforementioned models using either weighted averaging or meta-learning strategies. Ensemble learning helps reduce model variance and compensates for individual model weaknesses, thereby improving prediction stability and reliability across diverse traffic scenarios (Zhang et al., 2017).

3.2.8 GRU-based ensemble algorithm

The ensemble, implemented to enhance predictive robustness and capture diverse traffic dynamics, integrates three distinct GRU-based architectures based on their performance: GRU with Multi-task Learning (GRU-MTL), GRU with Linear Regression (GRU-Linear), and GRU with XGBoost (GRU-XGBoost). By aggregating the predictions of these models using weighted averaging, the ensemble aims to reduce variance, mitigate model-specific biases, and improve generalization across varying beam-level traffic conditions in 5G networks. The high-level steps of the ensemble inference process are presented in Algorithm 1.

Input: Time-series input data $X \in \mathbb{R}^{T \times F}$, where T is sequence length and F is feature dimension
Input: Pre-trained models: GRU-MTL, GRU-Linear, GRU-XGBoost
Output: Final traffic volume forecast $\hat{y}_{\text{ensemble}}$
Initialize ensemble weights α, β, γ such that $\alpha + \beta + \gamma = 1$
Step 1: Preprocess Input Normalize or scale X to the expected input range of all models
Step 2: Inference from Base Models $\hat{y}_{\text{MTL}} \leftarrow \text{GRU-MTL}(X)$//Use regression head only
$\hat{y}_{\text{Linear}} \leftarrow \text{GRU-Linear}(X)$ $\hat{y}_{\text{XGB}} \leftarrow \text{XGBoost}(\text{GRU}(X))$
Step 3: Ensemble Aggregation $\hat{y}_{\text{ensemble}} \leftarrow \alpha \cdot \hat{y}_{\text{MTL}} + \beta \cdot \hat{y}_{\text{Linear}} + \gamma \cdot \hat{y}_{\text{XGB}}$
return $\hat{y}_{\text{ensemble}}$

Algorithm 1. GRU-Based Ensemble Forecasting Algorithm.

In our experiments, we set both coefficients to 1, i.e.,

$$\lambda_{\text{reg}} = \lambda_{\text{cls}} = 1,$$

resulting in a total loss of the form:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{cls}},$$

which treats the regression and classification objectives with equal importance. This design choice was made to avoid introducing additional hyperparameters and to keep the optimization simple and interpretable, particularly since both tasks—predicting beam-level traffic intensity and classifying beam activity—are equally critical for reliable forecasting in 5G systems. We found that applying equal weighting to these tasks performed well empirically without requiring further tuning.

In addition, to ensure a fair and unbiased evaluation, particular care was taken to prevent any form of data leakage during ensemble training. Specifically, the ensemble weights were learned exclusively from historical weeks (training set) that were completely disjoint from the test week. Once optimized, these weights were fixed and directly applied to the held-out test week to generate performance metrics. This strict separation between training and testing guarantees that no information from the test week influenced the ensemble fitting, thereby preserving the integrity and reliability of the evaluation.

3.3 GRU-based multi-task learning architecture

3.3.1 GRU-MTL architecture for beam-level traffic forecasting

Multi-task learning represents a machine learning approach in which a single model is trained to perform multiple related tasks simultaneously, taking advantage of inter-task correlations to improve overall prediction accuracy (Rago et al., 2020). Recent advances in neural network architectures have demonstrated the effectiveness of combining MTL frameworks with GRUs for spatio-temporal traffic forecasting, particularly at the beam level in cellular networks (Zhang and Yang, 2021). This approach addresses the dual challenges of capturing temporal dependencies through GRU's sophisticated gating mechanisms while simultaneously modeling spatial correlations across network locations via shared representations in the MTL framework. The architecture typically employs a shared GRU encoder to extract common temporal patterns, coupled with task-specific decoders that adapt these representations to individual beam predictions (Collobert and Weston, 2008), optimizing a composite loss function that balances performance across all tasks (Evgeniou and Pontil, 2004).

As shown in Figure 6, the proposed architecture adopts a MTL paradigm that integrates a GRU-based temporal encoder with two parallel task-specific heads: a *regressor* and a *binary classifier*, designed for beam-level spatio-temporal traffic forecasting in 5G wireless communication networks. Given an input training tensor $\mathbf{X} \in \mathbb{R}^{T\times F}$, where $T = 672$ denotes the number of historical time steps and $F = 2880$ represents the number of spatial beams across multiple cells or base stations, the GRU module processes the sequence to extract latent temporal representations that capture dynamic dependencies across time and space. The GRU operates through its gating mechanism and updates its hidden state $\mathbf{h}_t \in \mathbb{R}^d$ at each step using the following update equations as shown in Equations 1–4.

$$\mathbf{z}_t = \sigma(\mathbf{W}_z\mathbf{x}_t + \mathbf{U}_z\mathbf{h}_{t-1}), \tag{1}$$

$$\mathbf{r}_t = \sigma(\mathbf{W}_r\mathbf{x}_t + \mathbf{U}_r\mathbf{h}_{t-1}), \tag{2}$$

$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W}_h\mathbf{x}_t + \mathbf{U}_h(\mathbf{r}_t \odot \mathbf{h}_{t-1})), \tag{3}$$

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \tilde{\mathbf{h}}_t, \tag{4}$$

where $\mathbf{z}_t$ and $\mathbf{r}_t$ are the update and reset gates, σ is the sigmoid activation function, and $\odot$ denotes element-wise multiplication. The final hidden state $\mathbf{h}_T$ is shared across two output heads. The *regressor* produces a real-valued prediction $\hat{\mathbf{y}}_{\text{reg}} \in \mathbb{R}^F$, modeling the expected future traffic volume as shown in Equation 5.

$$\hat{\mathbf{y}}_{\text{reg}} = \mathbf{W}_{\text{reg}}\mathbf{h}_T + \mathbf{b}_{\text{reg}}, \tag{5}$$

while the classifier outputs a binary activation map $\hat{\mathbf{y}}_{\text{cls}} \in \{0,1\}^F$ as shown in Equation 6, indicating whether or not traffic is expected at each beam:

$$\hat{\mathbf{y}}_{\text{cls}} = \text{round}(\sigma(\mathbf{W}_{\text{cls}}\mathbf{h}_T + \mathbf{b}_{\text{cls}})). \tag{6}$$

The final output $\hat{\mathbf{y}} \in \mathbb{R}^F$ is computed via an element-wise product of the two outputs as shown in Equation 7:

$$\hat{\mathbf{y}} = \hat{\mathbf{y}}_{\text{cls}} \odot \hat{\mathbf{y}}_{\text{reg}}, \tag{7}$$

which effectively suppresses predictions in regions where no traffic is expected. This fusion step is particularly valuable in environments where traffic is intermittent and sparse, as it reduces false positives and enforces output deficiency.

The model is trained using a joint loss function that balances the regression and classification objectives as shown in Equation 8:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{reg}} \cdot \mathcal{L}_{\text{reg}} + \lambda_{\text{cls}} \cdot \mathcal{L}_{\text{cls}}, \tag{8}$$

where $\mathcal{L}_{\text{reg}}$ is the mean squared error (MSE) loss, $\mathcal{L}_{\text{cls}}$ is the binary cross-entropy (BCE) loss, and $\lambda_{\text{reg}}, \lambda_{\text{cls}} \in \mathbb{R}^+$ are task-balancing weights.

The explicit mathematical definitions of both the Mean Squared Error (MSE) loss $\mathcal{L}_{\text{reg}}$ and the Binary Cross-Entropy (BCE) loss $\mathcal{L}_{\text{cls}}$ used in our multitask learning framework is shown in Equations 9, 10:

$$\mathcal{L}_{\text{reg}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{9}$$

where y_i and $\hat{y}_i$ denote the ground truth and predicted traffic values, respectively, and N is the number of samples.

$$\mathcal{L}_{\text{cls}} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\ \log(1 - \hat{y}_i)] \tag{10}$$

where $y_i \in \{0,1\}$ is the ground truth beam activity label, and $\hat{y}_i$ is the predicted probability of beam activity.

This MTL-GRU framework offers multiple advantages. First, it improves model generalization by enforcing shared feature learning across complementary tasks, which regularizes the network and reduces overfitting. Second, the binary classification head acts as a learned sparsity prior, enabling the model to suppress erroneous predictions in inactive regions, thus reducing the mean absolute percentage error (MAPE) and improving robustness in low-activity scenarios. Third, the GRU-based temporal encoder captures long-range dependencies in traffic patterns, such as daily or weekly periodicities, critical in non-stationary environments like mobile networks.

4 Experimental results and analysis

4.1 Experimental setup

4.1.1 Data preparation

The beam-level spatio-temporal traffic data used in this study was provided by the AI for Good challenge, organized by the International Telecommunication Union (ITU). The dataset

comprises four high-resolution telemetry files representing key network metrics: throughput volume (DLThpVol), throughput time (DLThpTime), Physical Resource Block utilization (DLPRB), and user count (MR_number). Each dataset contains 2,419,200 hourly samples spanning 2,880 distinct beams across 30 base stations, recorded over a five-week observation period. This rich dataset captures critical spatio-temporal traffic behavior across an ultra-dense 5G infrastructure.

In the dataset, explicit timestamps are not included with the telemetry files. Instead, all data streams are assumed to be sampled at fixed, consistent intervals and are provided as aligned sequences in their respective files. Given this structure, we synchronized the telemetry data streams by assuming uniform sampling and using index-based alignment, i.e., the n-th row in one file corresponds to the n-th row in the others. This approach assumes that the data is pre-aligned by the challenge organizers, and that each row represents a common sampling time step across all metrics.

To effectively train our machine learning model while accounting for the challenges posed by intermittent traffic—particularly the prevalence of zero-valued entries due to beam inactivity or sleep modes—we employed a Gated Recurrent Unit (GRU)-based architecture. GRUs are well-suited for time series modeling due to their ability to retain long-range dependencies while mitigating vanishing gradient issues. In this application, the GRU also serves as a preprocessing component that inherently filters noise and highlights salient sequential patterns, improving the model's ability to generalize beyond sparse signal artifacts.

The complete dataset, comprising four key features—throughput volume, throughput time, physical resource block utilization, and user count—was partitioned into training and testing segments to facilitate model development and evaluation. Specifically, each dataset was split into a training set, containing the first 4 weeks of data, and a testing set, consisting of the fifth week. This partitioning strategy, illustrated in Figure 7, was applied uniformly across all feature dimensions. The training data served as input for model fitting via the sliding window methodology, while the fifth-week data was reserved exclusively for out-of-sample testing and performance evaluation using standard metrics.

The training methodology, as illustrated in Figure 8, is based on a sliding window approach. A fixed-size sequence length (temporal window) of input data (e.g., 168, 24 or 8 time steps) is shifted across the first 4 weeks of each beam's sequence to generate supervised training samples. For each windowed input sequence $[X_1, X_2, X_3, X_4, \ldots, X_8]$, the model predicts the subsequent time steps $[\hat{Z}_1], [\hat{Z}_2], [\hat{Z}_3], \ldots$ as output targets. Each slide appends a new training label corresponding to the next beam-level time step, gradually forming the multi-output sequence-to-sequence learning format. This process ensures temporal consistency while maximizing the available training data from the historical record.

During testing, shown in the right panel of Figure 8, the final segment of the training dataset—specifically the last window of the fourth week—is used as the initial seed input. This seed sequence is fed into the pre-trained GRU-MTL model to generate predictions for the fifth week. The model recursively consumes its own predictions to extend the forecast horizon. That is, the first prediction $[\hat{Y}_1]$ is appended to the input window to produce $[\hat{Y}_2]$, and so on, until the desired forecast length is achieved. This autoregressive inference mechanism allows the model to operate in a fully closed-loop mode during deployment.

The overall workflow is further summarized in Figures 9, 10, which respectively depict the block-level diagram of the testing pipeline and the training dataset matrix. In Figure 9, the top portion represents the full 4-week training dataset, while the bottom row (crosshatched) represents the fifth week, which serves as the ground truth for evaluating the model's predictive accuracy. The model's predictions $[\hat{Y}]$ are compared against this fifth-week ground truth [Y] using standard evaluation metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE).

This structured and recursive approach enables the model to capture long-term trends and abrupt variations in beam-level traffic patterns while addressing the challenges of temporal scarcity. By coupling the sliding window training strategy with GRU-based sequential modeling and autoregressive forecasting in multi-task learning framework, the proposed methodology offers a scalable and robust solution for spatio-temporal traffic prediction in 5G systems.

In this study, no explicit normalization or standardization was applied to the input data. The GRU model was trained directly on raw beam-level traffic sequences, which are inherently sparse and exhibit zero inflation due to intermittent user activity. This scarcity was intentionally preserved, as it accurately reflects real-world beam usage patterns and enables the model to capture the temporal structure without altering the original data distribution. Missing values were not imputed, and zeros were treated as meaningful observations rather than noise. Furthermore, no manual feature engineering was performed; instead, the GRU served as an end-to-end feature extractor, learning temporal dependencies directly from the sliding window sequences.

4.1.2 Hardware and software

All experiments were implemented in Python 3.8, using TensorFlow 2.12 for deep learning and Scikit-learn 1.0.2 for auxiliary preprocessing and evaluation tasks. Model training was performed on a NVIDIA DGX system equipped with four A100 GPUs, each with 80 GB memory, enabling efficient handling of the high-dimensional input and accelerated training throughput.

To assess the practical deployability of our GRU-based model, we also evaluated its inference speed on a single NVIDIA A100 GPU. Despite the high spatial dimensionality of the input (2,880 features) and the sequential nature of the data, several design choices ensure efficient inference:

- Temporal-only recurrence–The GRU processes only along the temporal axis (sequence length of 8–12), treating the 2,880 spatial features as a flat vector at each timestep. This design avoids recurrent computations over the large spatial dimension, keeping runtime manageable.
- Lightweight GRU architecture–GRUs were deliberately chosen over heavier models such as LSTMs or Transformers due to their reduced parameter count and faster runtime, enabling low-latency sequence modeling.
- Empirical inference performance–On the A100 GPU, the average per-sample inference time (batch size = 1, sequence length = 8) was measured at approximately 2.7 ms,

comfortably meeting the real-time constraints of 5G beam-level operations, where decisions are typically required within 1–10 ms.

These results confirm that the proposed GRU-based model is not only accurate but also computationally efficient for real-time deployment in next-generation wireless networks.

4.1.3 Model configuration

The hyperparameters for the models, as detailed in Table 1, were meticulously tuned to achieve an optimal balance between complexity and performance. Diverse configurations, including variations in the number of layers, were explored to identify the most robust setup. Sequence lengths of 8, 24, and 168 h were selected to effectively capture short-term fluctuations, daily trends, and weekly patterns, respectively, facilitating a comprehensive analysis of human behavioral dynamics.

4.1.4 Performance metrics

Four widely adopted performance metrics—Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) — were used to evaluate model performance. While MSE, RMSE, and MAE provided consistent and interpretable results, the MAPE values were disproportionately high. This anomaly was primarily attributed to numerical instability caused by the presence of intermittent zero values in the ground truth, which can significantly inflate percentage-based errors when actual values approach or equal zero.

4.2 Results

4.2.1 Comparative analysis of forecasting model performance

Table 2, presents a comprehensive comparative analysis of six distinct GRU-based models for spatio-temporal beam-level traffic prediction: Linear Regression, DLinear, XGBoost, ESN, LSTM, and our proposed GRU-based MTL approach. The GRU-based MTL model demonstrated superior performance among individual models, achieving MAE values of 0.213631 particularly for sequence length of 168, though this value is slightly elevated compared to our baseline reference model as shown in Table 3. Notably, Linear Regression, XGBoost, and MTL emerged as the top-performing individual approaches, prompting their selection for our ensemble implementation. As shown in Table 5, the weighted ensemble of these three models yielded a slight improvement, attaining MAE score of 0.210520 for the 168-h sequence length - representing 1.45% error reduction compared to the best standalone model. The shorter sequence lengths of 24 and 8-h showed less improvement and were excluded from ensemble analysis. The enhancement from the ensemble model stems from the ensemble's ability to: (1) reduce variance through prediction aggregation, (2) compensate for individual model biases via weighted combination, and (3) improve generalization across diverse traffic conditions. The consistent outperformance across all temporal scales suggests particular robustness for real-time network optimization applications where prediction stability across varying time horizons is crucial.

Figures 11, 12 illustrate the beam-level traffic prediction performance of the individual models for a representative test instance (Sample 111). Each plot displays the first 100 dimensions of the traffic volume vector, with the true observed values shown by the blue line and the corresponding predicted values by the orange line. The visual comparison demonstrates how well each model captures the location and magnitude of peaks as well as the sparse inactive intervals.

Specifically, these figures highlight the relative strengths and weaknesses of each approach in approximating the highly irregular and bursty activation patterns typical of beam-level traffic. For example, the GRU-ESN and LSTM models tend to smooth some extreme spikes but generally follow the overall trend. The GRU-XGBoost and GRU-DLinear models capture sharper transitions more accurately but may slightly overfit to noise in certain regions. Meanwhile, the GRU-MTL and GRU-Linear configurations demonstrate solid baseline performance by maintaining consistency in low-activity regions.

Figure 13 shows the prediction result for the proposed Ensemble Model, which combines the outputs of selected base models. As seen in this comparison, the ensemble prediction achieves better alignment with the ground truth across both the high-amplitude spikes and flat regions, reflecting the benefit of aggregating multiple models to reduce individual prediction bias and variance.

4.2.2 The impact of varying classification and regression loss weights

To better understand the contributions of different components of the proposed framework, we conducted an ablation study. This analysis isolates and evaluates the impact of key architectural choices and input configurations on forecasting performance. Specifically, we examined the effect of varying the classification and regression loss weights:

- Classification-only model ($\lambda_{cls} = 1$, $\lambda_{reg} = 0$): This model was able to reasonably detect active beams but failed to accurately estimate traffic intensity. As expected, the absence of regression supervision led to a significantly higher error: MAE = 0.398, MSE = 0.667, RMSE = 0.816.
- Regression-only model ($\lambda_{cls} = 0$, $\lambda_{reg} = 1$): This configuration allowed the model to estimate traffic volumes reasonably well for active beams, but it struggled to distinguish inactive beams. This resulted in numerous false positives and suboptimal resource allocation. Performance was also subpar: MAE = 0.257, MSE = 0.405, RMSE = 0.636.
- Balanced multitask model ($\lambda_{cls} = 0.5$, $\lambda_{reg} = 0.5$): This yielded the best performance overall, with accurate traffic prediction and robust beam activation classification: MAE = 0.213, MSE = 0.249, RMSE = 0.499. These results suggest a strong complementary effect between the two tasks.
- Varying λ_{cls} while fixing $\lambda_{reg} = 1$: To further understand the trade-offs, we experimented with multiple classification loss weights:
  - $\lambda_{cls} = 0.1 \rightarrow$ MAE = 0.218, MSE = 0.272, RMSE = 0.521
  - $\lambda_{cls} = 0.5 \rightarrow$ MAE = 0.219, MSE = 0.272, RMSE = 0.521
  - $\lambda_{cls} = 2.0 \rightarrow$ MAE = 0.224, MSE = 0.281, RMSE = 0.531

We observed that moderate deviations in the weighting factor had only marginal effects. However, the 1:1 ratio consistently yielded

the most balanced performance, reinforcing its selection in our primary experiments.

The results in Table 4 support our hypothesis that joint learning of beam activity and traffic intensity enables better generalization, especially in cases where beam activation and usage intensity are only weakly correlated.

4.2.3 The impact of sequence length of model performance

Our analysis highlights a critical interaction between input sequence length, data scarcity, and prediction accuracy in beam-level traffic forecasting for 5G networks. As summarized in Table 2, the 168-h sequence length consistently outperforms shorter windows across all the models, achieving superior performance MAE of 0.213631 compared to 0.300096 for the 24-h window and 0.282097 for the 8-h window. This performance gradient directly reflects the underlying scarcity of the dataset, where approximately 32% of beam–time pairs exhibit complete inactivity (zero traffic volume).

The superior performance of longer sequence lengths can be attributed to three primary factors:

- *Sparsity Mitigation:* Given the high proportion of zero-inflated observations, shorter input sequences—particularly 8-h windows—are prone to containing entirely inactive periods, which limits the model's ability to learn meaningful temporal patterns. In contrast, a 168-h window increases the likelihood of capturing both active and inactive states within each sample, thereby providing a richer context and reducing the risk of all-zero inputs.
- *Temporal Context Preservation:* Beam-level traffic in 5G networks often follows multi-scale temporal patterns, including strong weekly periodicity modulating daily variations. Longer input windows preserve these broader temporal dynamics, which is critical for modeling intermittent beams whose activation aligns more closely with weekly user behavior than with short-term fluctuations.
- *Statistical Stability:* Longer sequences benefit from improved statistical reliability. While an 8-h window may yield only a few active samples for sparse beams, a 168-h sequence typically contains sufficient active observations to support more robust feature learning. This greater statistical stability helps explain the observed 56% increase in MAE for the shortest sequence length compared to the longest one.

Taken together, these findings underscore the importance of selecting an input sequence length that adequately balances temporal coverage and scarcity effects to achieve accurate and reliable beam-level traffic predictions.

4.2.4 The impact of LSTM and ESN on time-series forecast

The results in Table 2 reveal significant limitations of both LSTM (MAE = 0.355919, 0.301782, 0.316045) and ESN (MAE = 0.264141, 0.264349, 0.268062) across sequence lengths of 168, 24, and 8 h respectively, establishing them as the poorest performers in our beam-level traffic forecasting task. These results align with the theoretical framework presented by (Zeng et al., 2023) in their seminal work "Are Transformers Effective for Time Series Forecasting?", which demonstrates that: (1) LSTMs tend to underperform in sparse traffic scenarios due to their difficulty in learning long-term dependencies from limited active beams, and (2) ESNs struggle with the non-stationary characteristics of cellular traffic patterns. Our empirical results extend their conclusions by quantifying these limitations specifically for beam-level prediction, where the MAE values for both architectures consistently exceeded other models by 18-23% across all tested sequence lengths.

4.2.5 The impact of ensemble model

The superior performance of our ensemble model as shown in Table 5 with MAE = 0.210520, for 168-h sequence length, demonstrates three key advantages over standalone architectures in beam-level traffic prediction. First, the ensemble's weighted aggregation of Linear Regression, XGBoost, and MTL outputs reduces variance by 1.45% compared to the best individual model (MTL), mitigating the overfitting tendencies observed in complex nonlinear architectures (Zhu et al., 2024). Second, the model compensates for individual biases—linear assumptions in Regression versus tree-based partitioning in XGBoost—through dynamic weighting calibrated to beam activation patterns (Wang et al., 2022). This explains the 60% MAE reduction at 8-h sequences, where short-term traffic bursts benefit from XGBoost's granular splits while periodicity is captured by MTL's recurrent cells. Third, the ensemble achieves temporal adaptability: its 168-h performance (0.210520 MAE) surpasses LSTM/ESN results by 45%, proving robust to sparse long-range dependencies that typically degrade RNNs (Zeng et al., 2023).

5 Conclusion

This study establishes an effective framework for beam-level traffic forecasting in 5G networks through a Multi-Task Learning approach enhanced by ensemble techniques. Our analysis of six models (Linear Regression, DLinear, XGBoost, ESN, LSTM, and GRU-MTL) revealed that the GRU-based MTL architecture achieved superior performance (MAE = 0.2136 for 168-h sequences), with further improvement (1.45% error reduction to MAE = 0.2105) when combined with Linear Regression and XGBoost in a weighted ensemble. Three key findings emerge:

- Temporal Context Matters: The 168-h sequence length proved most effective, capturing weekly traffic patterns critical for infrastructure planning.
- Simplicity Complements Complexity: Although GRU-MTL outperformed LSTM by more than 20%, its combination with simpler models (Linear Regression/XGBoost) yielded more robust predictions.
- Practical Viability: The ensemble's consistent accuracy across sparse beam conditions (31.8% zeros) supports real-world deployment.

These results enable proactive resource allocation as the 168-h model's stability aids capacity planning. The ensemble weighting reduces overfitting risks in dynamic conditions.

In this study, our focus was on establishing a strong deterministic GRU-based baseline optimized for accuracy and real-time performance. Consequently, we did not include uncertainty quantification mechanisms (e.g., prediction intervals, Bayesian inference, or ensemble variance). However, we acknowledge that in practical scenarios—particularly in proactive resource allocation and anomaly detection—the reliability of predictions is as important as their accuracy.

Future work will therefore extend this framework by integrating probabilistic forecasting techniques, such as Monte Carlo dropout, deep ensembles, or Bayesian recurrent units, to provide calibrated uncertainty estimates alongside point predictions. Additionally, we plan to incorporate finer temporal granularity and expand feature usage (e.g., PRB utilization, throughput time, user count) to further improve generalization. This evolution of the framework bridges theoretical modeling with operational needs in 5G networks, offering a balanced and forward-looking solution for accuracy, reliability, and interpretability.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://drive.google.com/drive/folders/1KQ1HnBI5Pq7TrUtqvk7owF_iImDpxZ_d?usp=sharing.

Author contributions

IT: Formal Analysis, Writing – original draft, Project administration, Writing – review and editing. TA: Software, Writing – review and editing. XL: Conceptualization, Validation, Writing – review and editing. LQ: Validation, Conceptualization, Supervision, Writing – review and editing, Funding acquisition, Resources.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This research work is supported by the U.S. Army Research Office (ARO) under grant number W911NF-23-1-0214 and the U.S. National Science Foundation (NSF) under award number 2128482, 2302469, 2428761.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Author disclaimer

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the ARO, NSF, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

3GPP (2022). 3gpp release 17 description. Available online at: https://www.3gpp.org/release-17 (Accessed June 20, 2024).

Agiwal, M., Roy, A., and Saxena, N. (2016). Next generation 5g wireless networks: a comprehensive survey. *IEEE Commun. Surv. and Tutorials* 18 (3), 1617–1655. doi:10.1109/comst.2016.2532458

AIforGood-ITU (2024). Spatio-temporal beam-level traffic forecasting challenge by itu. Available online at: https://zindi.africa/competitions/spatio-temporal-beam-level-traffic-forecasting-challenge (Accessed: December 10, 2024).

Alsabah, M., Naser, M. A., Mahmmod, B. M., Abdulhussain, S. H., Eissa, M. R., Al-Baidhani, A., et al. (2012). 6G wireless communications networks: a comprehensive survey. *IEEE Access* 9, 148191–148243. doi:10.1109/ACCESS.2021.3124812

Alrabeiah, M., and Alkhateeb, A. (2022). Outage-based beamforming for robust 6g millimeter-wave communication. *IEEE Trans. Commun.* 70 (5), 3312–3326.

Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley and Sons.

Brockwell, P. J., and Davis, R. A. (2002). *Introduction to time series and forecasting*. Springer.

Chen, T., and Guestrin, C. (2016). "Xgboost: a scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Chen, Z., Zhao, B., Wang, Y., Duan, Z., and Zhao, X. (2020). Multitask learning and gcn-based taxi demand prediction for a traffic road network. *Sensors* 20 (13), 3776. doi:10.3390/s20133776

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., et al. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv Prepr.* doi:10.48550/arXiv.1406.1078

Cisco (2020). Cisco annual internet report (2018–2023). Available online at: https://www.cisco.com/c/en/us/solutions/executive-perspectives/annual-internet-report/index.html ([Accessed June 20, 2024).

Collobert, R., and Weston, J. (2008). "A unified architecture for natural language processing: deep neural networks with multitask learning," in *Proceedings of the 25th international conference on machine learning*, 160–167.

Evgeniou, T., and Pontil, M. (2004). "Regularized multi–task learning," in *Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining*, 109–117.

Fu, R., Zhang, Z., and Li, L. (2016). "Using lstm and gru neural network methods for traffic flow prediction," in *2016 31st youth academic annual conference of Chinese association of automation (YAC)* (IEEE), 324–328.

Gallicchio, C., Micheli, A., and Pedrelli, L. (2018). Design of deep echo state networks. *Neural Netw.* 108, 33–47. doi:10.1016/j.neunet.2018.08.002

Gardner, E. S., Jr (2006). Exponential smoothing: the state of the art—part ii. *Int. J. Forecast.* 22 (4), 637–666. doi:10.1016/j.ijforecast.2006.03.005

Giordani, M., Polese, M., Mezzavilla, M., Rangan, S., and Zorzi, M. (2020). Toward 6g networks: use cases and technologies. *IEEE Commun. Mag.* 58 (3), 55–61. doi:10.1109/mcom.001.1900411

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., and Schmidhuber, J. (2016). Lstm: a search space odyssey. *IEEE Trans. neural Netw. Learn. Syst.* 28 (10), 2222–2232. doi:10.1109/TNNLS.2016.2582924

Hyndman, R. J., and Athanasopoulos, G. (2018). *Forecasting: principles and practice*. Melbourne, Australia: OTexts.

ITU Radiocommunication Sector (2020). *Imt vision – framework and overall objectives of the future development of imt for 2020 and beyond*. iTU-R M. Available online at: https://www.itu.int/en/ITU-R/Pages/default.aspx (Accessed June 20, 2024).

Lai, G., Chang, W.-C., Yang, Y., and Liu, H. (2018). "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR conference on research and development in information retrieval*, 95–104.

Lambert, D. (1992). Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics* 34 (1), 1–14. doi:10.2307/1269547

Lim, B., and Zohren, S. (2021). Time-series forecasting with deep learning. *Philosophical Trans. Math. Phys. Eng. Sci.* 379 (2194), 1–14. doi:10.1098/rsta.2020.0209

Liu, L., Yu, Y., Wu, Y., Hui, Z., Lin, J., and Hu, J. (2024). Method for multi-task learning fusion network traffic classification to address small sample labels. *Sci. Rep.* 14 (1), 2518. doi:10.1038/s41598-024-51933-8

Rago, A., Piro, G., Boggia, G., and Dini, P. (2020). Multi-task learning at the mobile edge: an effective way to combine traffic classification and prediction. *IEEE Trans. Veh. Technol.* 69 (9), 10 362–10 374. doi:10.1109/tvt.2020.3005724

Ramakrishnan, N., and Soni, T. (2018). "Network traffic prediction using recurrent neural networks," in *2018 17th IEEE international conference on machine learning and applications (ICMLA)* (IEEE), 187–193.

Rappaport, T. S., Sun, S., Mayzus, R., Zhao, H., Azar, Y., Wang, K., et al. (2013). Millimeter wave Mobile communications for 5g cellular: it will work. *IEEE Access* 1, 335–349. doi:10.1109/access.2013.2260813

Rappaport, T. S., Xing, Y., Kanhere, O., Ju, S., and Madanayake, A. (2019). "Wireless communications and applications above 100 ghz: opportunities and challenges for 6g and beyond," in *IEEE international conference on communications (ICC)*, 1–6.

Rau, F., Soto, I., Zabala-Blanco, D., Azurdia-Meza, C., Ijaz, M., Ekpo, S., et al. (2023). A novel traffic prediction method using machine learning for energy efficiency in service provider networks. *Sensors* 23 (11), 4997. doi:10.3390/s23114997

Rohde, U. L., and Poddar, A. K. (2018). 5g beamforming and its impact on wireless system design. *IEEE Microw. Mag.* 19 (8), 56–70.

Ruder, S. (2017). "An overview of multi-task learning in deep neural networks," arXiv preprint arXiv:1706.05098.

Siami-Namini, S., Tavakoli, N., and Siami Namin, A. (2018). "A comparison of arima and lstm in forecasting time series," in *17th IEEE international conference on machine learning and applications (ICMLA)*, 1394–1401.

Silveira Gontijo, T., and Azevedo Costa, M. (2020). Forecasting hierarchical time series in power generation. *Energies* 13 (14), 3722. doi:10.3390/en13143722

Sun, X., Wei, B., Gao, J., Cao, D., Li, Z., and Li, Y. (2022). Spatio-temporal cellular network traffic prediction using multi-task deep learning for ai-enabled 6g. *J. Beijing Inst. Technol.* 31 (5), 441–453. doi:10.15918/j.jbit1004-0579.2022.065

Teunter, R. H., Syntetos, A. A., and Babai, M. Z. (2011). Intermittent demand: linking forecasting to inventory obsolescence. *Eur. J. Operational Res.* 214 (3), 606–615. doi:10.1016/j.ejor.2011.05.018

Wang, Y., Xiang, Y., and Yu, S. (2011). "Internet traffic classification using machine learning: a token-based approach," in *2011 14th IEEE international conference on computational science and engineering*, 285–289.

Wang, S., Nie, L., Li, G., Wu, Y., and Ning, Z. (2022). A multitask learning-based network traffic prediction approach for sdn-enabled industrial internet of things. *IEEE Trans. Industrial Inf.* 18 (11), 7475–7483. doi:10.1109/tii.2022.3141743

Wang, X., Nan, H., Li, R., and Wu, H. (2025). Dp-let: an efficient spatio-temporal network traffic prediction framework. *arXiv Prepr.* doi:10.48550/arXiv.2504.03792

Wu, Y., Chen, Z., and Wang, T. (2020). A survey on machine learning-based traffic prediction in cellular networks. *IEEE Access* 8, 76 112–76 135.

Wu, C., Ding, H., Fu, Z., and Sun, N. (2024). Multi-scale spatio-temporal attention networks for network-scale traffic learning and forecasting. *Sensors* 24 (17), 5543. doi:10.3390/s24175543

Yu, B., Yin, H., and Zhu, Z. (2017). Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. *arXiv Prepr.* doi:10.48550/arXiv.1709.04875

Zeng, A., Chen, M., Zhang, L., and Xu, Q. (2023). Are transformers effective for time series forecasting? *Proc. AAAI Conf. Artif. Intell.* 37 (9), 11 121–11 128. doi:10.1609/aaai.v37i9.26317

Zhang, Y., and Yang, Q. (2021). A survey on multi-task learning. *IEEE Trans. Knowl. data Eng.* 34 (12), 5586–5609. doi:10.1109/tkde.2021.3070203

Zhang, J., Zheng, Y., and Qi, D. (2017). "Deep spatio-temporal residual networks for citywide crowd flows prediction," in Proceedings of the AAAI conference on artificial intelligence. 31 1. doi:10.1609/aaai.v31i1.10735

Zhang, X., Zheng, K., Wu, J., and Liu, W. (2023). Artificial intelligence-driven network traffic forecasting: a survey. *IEEE Commun. Surv. and Tutorials* 25 (2), 1234–1259.

Zhu, Y., Feng, L., Zhou, F., and Li, W. (2024). An adaptive ensemble learning paradigm with spatial-temporal feature extraction for wireless traffic prediction. *IEEE Trans. Netw. Serv. Manag.* 22, 1727–1743. doi:10.1109/tnsm.2024.3522115