Mobile Network Data Synthesis with Generative AI: Challenges and Solutions

Sijing Duan, Ye Zhang, Feng Lyu, Conghao Zhou, and Xuemin (Sherman) Shen

Abstract

Mobile network data plays a vital role in designing intelligent services for cellular, vehicular, and satellite networks, etc. However, restricted data access poses a barrier to conducting effective and open data-driven research. Data synthesis is a promising solution to address the barrier in the era of generative artificial intelligence (GAI). In this article, we comprehensively study mobile network data synthesis via GAI techniques. Specifically, we first discuss the motivation and challenges for implementing mobile network data synthesis. Then, we introduce several key technologies that address challenges specific to typical mobile network traffic, trajectory, and application usage data. Finally, we propose an *AppSyn* method for synthesizing mobile application usage data based on large language models and conduct a case study. Experimental results demonstrate the effectiveness of our proposed method compared to state-of-the-art benchmarks.

Introduction

Mobile networks connect users to the internet and each other through diverse technologies, including cellular, vehicular, and satellite systems [1, 2]. These networks generate vast amounts of data from device-to-network communications, user activity monitoring, and system-level processes. Such data includes such as mobile traffic, base station (BS) associations, user mobility trajectories, signal strength, and application usage statistics. This rich dataset serves as the foundation for intelligent services, i.e., artificial intelligence (AI)-powered applications designed to analyze and leverage mobile network data to optimize network operations and enhance user experiences [3]. These services enable operators to improve traffic prediction and management in vehicular networks and optimize load balancing in cellular systems. Furthermore, mobile network data can enhance smart city management and deliver personalized recommendations, fostering proactive decision-making across various domains.

Despite the vast potential of intelligent services empowered by data, limited data accessibility poses a significant challenge. Specifically, network operators are reluctant to share their data due to strict privacy regulations and the high costs associated with data collection, which often limit access to valuable mobile network data, curbing innovation, and research reproducibility. To address these challenges, synthetic data generation has emerged as a promising solution and has attracted considerable attention. Synthetic data is artificial data created by computers to mimic real-world data without exposing sensitive information. High-quality synthetic data offers three key benefits. First, it improves data availability in scenarios where collecting real-world data is scarce due to cost, missing data, or accessibility constraints. Second, it improves system performance through training dataset augmentation [4, 5]. Third, it preserves privacy (e.g., user identifiers and mobility patterns) by generating data that retains the statistical properties of real-world datasets without directly replicating sensitive details, thus enabling secure data sharing.

Recent advancements in generative artificial intelligence (GAI) models, including generative adversarial networks (GANs), diffusion models, and large language models (LLMs) like generative pre-trained transformers (GPT) and Llama, have provided new opportunities for synthesizing data across various domains, including mobile networks. These models excel at replicating data characteristics, enabling high-quality data synthesis. With the rapid progress in generative models, significant efforts have focused on synthesizing network traffic data, such as Internet of Things (IoT) device traffic [6] and single base station (BS) traffic [7]. However, comprehensive mobile network data synthesis, which spans multiple dimensions such as mobile traffic, BS association trajectories, and mobile user app usage behaviors, has received limited attention. A holistic approach to mobile network data synthesis is essential to capture the interdependencies among these diverse data types, enabling more robust and comprehensive data synthesis.

However, the comprehensive synthesis of mobile network data still presents several significant challenges. First, mobile network data is affected by factors such as user mobility patterns, network topology, and traffic load variations, which pose difficulties for data synthesis. Second, synthetic data may retain sample-level patterns and details from the original datasets. This poses privacy risks for re-identification and attribute disclosure by membership inference or

Sijing Duan is with Tsinghua University, China; Ye Zhang (corresponding author) is with Beijing Information Science and Technology University; Feng Lyu is with Central South University, China; Conghao Zhou is with Xidian University, China; Xuemin (Sherman) Shen is with the University of Waterloo, Canada.

Digital Object Identifier: 10.1109/MCOM.001.2500052

0163-6804/25/$25.00 © 2025 IEEE

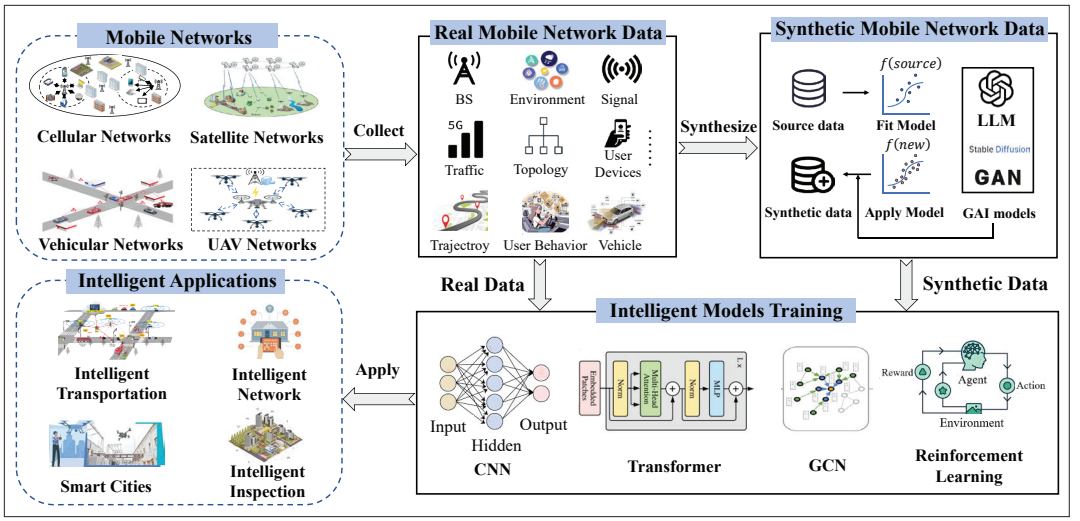

FIGURE 1. Overview of mobile network data collection, synthesis, and application with GAI.

attribute linkage, especially with sensitive information. Developing data synthesis methods that retain the statistical properties of real-world data while ensuring privacy remains challenging.

In this article, we present a holistic overview of mobile network data synthesis by using GAI models. We first introduce motivation and challenges for diverse attributes of mobile network data synthesis. Then, we present several key technologies such as GANs, diffusion models, and LLM for addressing these challenges. Finally, we carry out a case study of mobile application data usage synthesis and propose an *AppSyn* framework, experimental results demonstrate the efficacy of the proposed framework.

Mobile Network Data Synthesis: Motivation and Challenges

Motivation

Enhancing Data Availability: Data synthesis techniques can significantly reduce reliance on costly real-world data collection while expanding data availability for diverse applications, with practical adoption in areas such as network optimization, intelligent transportation, personalized services, academic research, and privacy-preserving data sharing. Particularly, GANs effectively capture multi-source information, such as user attributes and network traffic by modeling complex data distributions. Diffusion models excel in simulating spatiotemporal dynamics for the synthesis of mobility patterns and temporal dependencies. LLMs can generate natural language representations of network activities, enhancing interpretability and contextual richness. Together, these techniques produce augmented datasets that support intelligent network services, ultimately enhancing system performance.

Preserving Data Privacy: Data synthesis techniques facilitate secure data sharing by creating datasets that preserve the statistical characteristics of real-world data while ensuring sensitive details from the original data are not breached. To further enhance privacy performance, data synthesis methods can be combined with privacy-preserving techniques like differential privacy. By introducing controlled noise into the synthetic data generation process, differential privacy ensures that no individual user's information can be inferred from the dataset while preserving the overall statistical properties.

Framework Overview

Figure 1 presents an overview of mobile network data collection, synthesis, and application with GAI. Diverse mobile networks give rise to a vast amount of real-world data. These data serve as the foundation for training and fine-tuning GAI models, thereby generating synthetic data. Following Gartner's report about future AI trends, it is anticipated that synthetic data will entirely supplant real data in AI models by 2030 [8]. Both real and synthetic data can be employed for training intelligent models such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Graph Neural Networks (GCN), Transformer networks, and reinforcement learning networks. With a plentiful of data, deep learning models can be efficiently trained to support a diverse range of intelligent services and applications, including intelligent transportation, network infrastructure, smart cities, and UAV-assisted inspections.

Overall Challenges

Modeling Complex and Dynamic Characteristics: The intricate and dynamic nature of mobile network data stems from user mobility patterns, such as varying speeds and movement trajectories, combined with real-time network conditions fluctuations like signal strength, congestion levels, and handovers between BSs. These complexities pose significant challenges to GAI models in synthesizing data, as models must accurately capture temporal dependencies, spatial correlations, and the interplay between user behavior and network dynamics to ensure reliable data modeling and synthesis.

Balancing Data Utility and Preserving Privacy: A significant challenge in data synthesis is achieving a balance between maintaining the data utility and preserving the statistical properties of the original dataset. The synthesized data should ensure essential patterns, correlations, and distributions that allow AI models to learn and perform effectively on real-world tasks. At the same time,

With a plentiful of data, deep learning models can be efficiently trained to support a diverse range of intelligent services and applications, including intelligent transportation, network infrastructure, smart cities, and UAV-assisted inspections.

GAN models have been widely proven to be capable of flexibly synthesizing various types of data, with the ability to model complex distributions and multivariate dependencies.

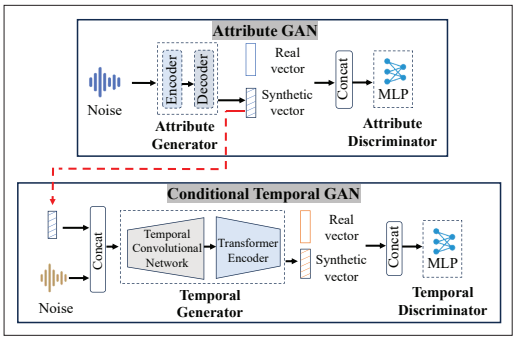

FIGURE 2. Design overview of mobile traffic data synthesis framework.

privacy safeguards must be in place to prevent sensitive information from revealing, such as individual user trajectories.

Key Technologies for Mobile Network Data Synthesis

This section presents several key GAI-based data synthesis technologies on different data categories in mobile network systems: mobile traffic, mobile network trajectory, and mobile application usage. The novelty of the core technologies lies in adaptations and integrations to generate synthetic mobile network data, which extend beyond standard applications of GANs, diffusion models, and LLMs.

Mobile Traffic Data Synthesis

Mobile traffic data specifically refers to the amount of traffic consumption when a user connects to the BS through mobile apps. Each entry in the dataset includes an anonymous user identifier (ID), uplink and downlink traffic measurement values, and start and end timestamps. In addition, the dataset also includes user attributes such as age and gender since user traffic usage is significantly related to their attributes, which is important for downstream applications such as personalized recommendations. Synthesizing mobile network traffic data requires considering two aspects. Firstly, traffic data display multi-scale temporal patterns, consisting of daily periodicity, weekly fluctuations, and abrupt bursts. It necessitates that the data synthesis model can capture these temporal dynamics and correlations. Secondly, different data attributes, such as user age, gender, and application usage frequency are highly correlated. Understanding interrelations among various impacting factors, such as the relations between age and application usage preference, could improve data synthesis's effectiveness. Thus, the mobile traffic data synthesis model needs to comprehensively capture the relationship between network traffic and user attributes.

GAN models have been widely proven to be capable of flexibly synthesizing various types of data, with the ability to model complex distributions and multivariate dependencies. Furthermore, GAN models can preserve the statistical properties of the original data when synthesizing data. We can design a user attribute-aware network traffic data synthesis model based on GAN, which simultaneously models temporal traffic and categorical attributes. As shown in Fig. 2, two components can be integrated into GAN to generate mobile network data.

1. *Attribute GAN*: This component captures and generates diverse user profiles. The generator uses an encoder-decoder structure, where the encoder built with stacked linear layers, extracts latent representations, and the decoder maps them back to an attribute vector. The real attribute vector and synthesis vector are concatenated and fed to the discriminator. The discriminator is a multilayer perceptron (MLP), discerning real from generated attribute vectors.
2. *Conditional Temporal GAN*: This component is designed to generate temporal traffic sequences based on user attributes, including a generator consisting of a Temporal Convolutional Networks (TCNs) for short-term patterns (e.g., daily and weekly trends) and a Transformer encoder for long-term dependencies. By combining TCNs and Transformers, our hybrid approach integrates the local, short-term modeling prowess of TCNs with the long-range, global modeling capabilities of Transformers. This synergy enables the model to generate synthetic traffic data that mirrors the multi-scale temporal structure of real-world mobile networks. The real traffic series vector and synthesis vector are concatenated finally. The discriminator uses an MLP to evaluate sequence realism. The time-granularity (e.g., minute or hourly based) of generated samples can be adapted to different requirements by adjusting the input sequence segmentation and model parameters accordingly.

Mobile Network Trajectory Data Synthesis

Mobile network trajectory data is associated with mobile devices. In cellular networks, trajectories represent the BS association of individuals using mobile devices, influenced by signal strength, coverage areas, and mobility patterns. Each BS association entry includes information about the ID, geographic location coordinates, and signal coverage radius. Therefore, cellular association trajectory synthesis should generate both BS association ID sequence and handover trajectory. However, accurately mapping user movements from geographic locations to BS handover-based cellular association trajectory is not straightforward for two reasons.

1. *BS association sequence exhibits heterogeneity*. Users traverse the communication coverage areas of various BSs, resulting in user-specific BS association ID sequences representing distinct mobility patterns. Besides differences, BS ID association sequences corresponding to each user may exhibit similarity due to regular users' daily routines, such as staying at workplaces during the day and returning home at night.
2. *The disparity between road and BS spaces*. The disparity is primarily characterized by heterogeneous distribution and representation mismatch of road segments and BSs. Road segments exhibit a continuous and interconnected distribution, whereas BSs are scattered and irregularly distributed. In addition, the complexity of handover dynamics is influenced by signal strength, user speed, and load-balancing algorithms, making it

difficult to generate cellular association sequences that mimic real-world user mobility behavior. A potential solution is to use a two-stage trajectory generation method, as shown in Fig. 3.

In the first stage, we generate spatial BS ID association sequences using a GAN structure. The generator uses an LSTM-based component to create the sequence of BS IDs in the order a user might connect to them (e.g., BS1 → BS2 → BS3). Additionally, a personalized linear layer is designed to ensure that the generated BS IDs align with the specific BS set associated with each user. The discriminator checks if the sequence looks realistic by comparing it to real data.

In the second stage, we adopt a two-step method to generate cellular association trajectories. First, we synthesize coarse-grained cellular association trajectories for a given road segment-based trajectory, where three modules are involved.

1. *GPS reference point generation*. Roads and BSs don't naturally align, so we convert both into GPS coordinates. This creates a unified map where we can track a user's movement more accurately.
2. *Arrival time generation*. We estimate when a user reaches each GPS point using historical average travel speeds (e.g., from APIs like Amap). For instance, if it takes 5 minutes to drive between two points, we assign timestamps accordingly.
3. *Bayesian decision*. Using the GPS points, timestamps, and BS locations, a Bayesian decision model figures out the most likely BS a user connects to at each moment. It considers factors like distance or signal strength to make an estimation. The output is a coarse-grained trajectory.

Then, we refine the synthesized coarse-grained trajectory with a diffusion-based model. This includes two steps, i.e., forward diffusion and reverse denoising. Specifically, we start with the coarse trajectory and add Gaussian noise. This mimics real-world unpredictability such as traffic fluctuations and signal dynamics. Then, a trained diffusion model removes the noise step-by-step, reconstructing a smooth and realistic trajectory. It uses a decoder to understand spatial relationships (e.g., how BSs are laid out geographically) and Transformer layers to track time patterns (e.g., rush hour vs. late night). Finally, we pair each BS with its timestamp to complete the trajectory. The output is a fine-grained trajectory that contains detailed GPS points, accurate BS connections, and natural variations.

Mobile Application Usage Data Synthesis

Mobile application (app) usage data synthesis poses several specific challenges. App usage data is usually mixed with discrete and continuous attributes with different distributions, such as app labels, traffic consumption, usage time, and usage frequency. The relationship among these attributes cannot be intuitively captured and unknown in advance. Furthermore, the app usage data usually contains user-level sensitive information, so avoiding revealing identifiable information and retaining sample-level patterns from original datasets is challenging.

Large language models (LLMs) offer a flexible way to generate synthetic mobile app usage data under given feature conditions for two key reasons. First, LLMs use a fully text-based tokenization process that avoids the need to predefine data types, reducing preprocessing loss and preventing dimensionality issues caused by one-hot encoding in high-dimensional data. Second, by including both feature names and values in the text encoding, LLMs preserve more contextual information, which is beneficial for modeling heterogeneous data. To leverage this, we propose a template-based text coding strategy to represent features like user behavior patterns and traffic sequences as compact text. An autoregressive fine-tuning approach using a pre-trained LLM (e.g., Llama 3) is then used to learn the format and temporal patterns of traffic data, enabling dynamic data synthesis under any conditions. A detailed model design will be presented in the case study.

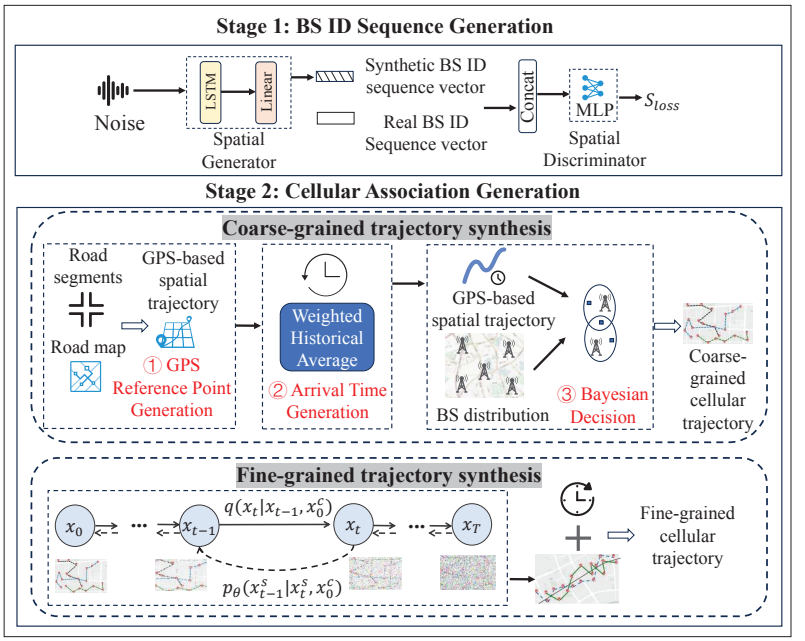

FIGURE 3. Overview of mobile network trajectory data synthesis framework.

Privacy-Enhanced Data Synthesis

To further enhance the privacy performance of data synthesis, differential privacy can be used in the training process. For GAN-based mobile traffic data synthesis, we can train a Wasserstein GAN (WGAN) with differential privacy. This is achieved by sanitizing the gradients during training to limit the influence of any single data point, tuning hyperparameters for balance between privacy and utility, and employing a decaying clipping bound to improve stability. During the training process, multiple models are saved at different epochs. We can select the best private model utilizing a quality measure (such as the L1-distance between histograms of real and synthetic data) and perturb the scores with noise. For example, the top-K models are selected based on their noisy scores. Finally, we can conduct a privacy analysis to ensure differential privacy.

For LLM-based mobile app usage data synthesis, fine-tuning can be performed using differential privacy stochastic gradient descent. To enhance training performance, two novel loss functions are introduced: weighted cross-entropy loss (WCEL) and numerical cross loss (NCL). WCEL helps the

The primary objective of the LLM fine-tuning training process is to maximize the probability of generating the correct sequence of tokens, thereby ensuring that the model accurately learns the patterns and statistical relationships within the data.

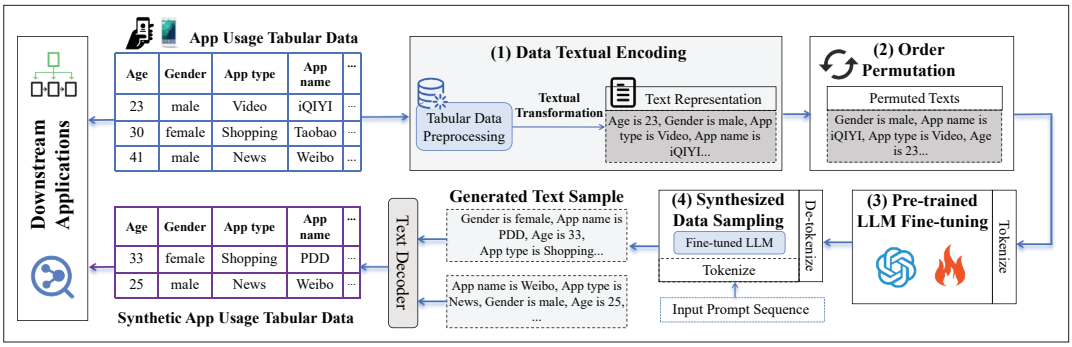

FIGURE 4. Design overview of app usage synthesis *AppSyn* framework.

model distinguish between data format terms and data feature terms, guiding it to focus more on feature distribution learning. In the differential privacy setting, this supports a balance between data utility and privacy by adjusting attention across tokens. NCL, on the other hand, captures numerical differences between predicted and true values in the serialized text, allowing the model to learn numerical features more precisely while avoiding overfitting and enhancing privacy protection.

Case Study

Methodology

As shown in Fig. 4, we propose an LLM-based framework *AppSyn* to generate mobile app usage synthetic data. Each app usage data sample includes seven attributes: user identifier, age, gender, app label, app traffic consumption, days, and app usage frequency. To model these attributes, *AppSyn* consists of four primary components: data textual encoding, feature order permutation, pre-trained LLM fine-tuning, and synthesized data sampling.

Data Textual Encoding: The data used for fine-tuning a pre-trained LLM are typically sentences. To achieve the transformation from tabular app usage data to textual sentences, we first encode the original app usage dataset into natural language text with a textual encoding method. Specifically, we use a subject-predicate-object template approach that can allow us to convert each data sample into a well-formed sentence by mapping the structured data fields into textual elements. Each record in the app usage dataset is represented as a "subject," a "predicate," and an "object." For example, if the dataset contains a record of a user using TikTok 30 times, this could be encoded as *"APP is TikTok, frequency is 30 times."* This textual encoding method bridges the gap between structured data and natural language, allowing the LLM to better capture the relationships between different data points.

Feature Order Permutation: Since pseudo-location information of features is artificially introduced after the structured data is transformed into textual data, feature order permutation is designed to randomly disrupt the order of feature vectors for feature independence. The permutation operation enables the LLM to be generated controllably under any feature attribute conditions.

LLM Fine-Tuning: We fine-tune a pre-trained LLM (i.e., LLama-2 7B model), on the textually encoded dataset. The first step is tokenization, which preprocesses the permuted textual corpus into a sequence of tokens, such as words, subwords, or even individual characters. Next, we use the LLM to generate each token in the sequence auto-regressively, where each token is predicted based on the previously generated tokens. Specifically, the next token is sampled by weighted choice sampling with a temperature parameter from the LLM's output. The temperature parameter can be used to control the randomness level of token selection. The lower temperatures can lead to a more deterministic LMM model. Higher temperatures encourage diversity by allowing the model to explore less probable tokens. The primary objective of the LLM fine-tuning training process is to maximize the probability of generating the correct sequence of tokens, thereby ensuring that the model accurately learns the patterns and statistical relationships within the data.

Synthesized Data Generation: With a fine-tuned LLM, we can generate a final synthesized app usage dataset with a prompt-based sampling strategy. The prompt provides a feature name and generates the corresponding feature value from a joint distribution. Finally, a pattern-matching algorithm decodes the generated text features into structured data form through regular expressions. The synthetic data can be used for a variety of downstream applications.

Experiment Evaluation

Setup: We perform experiments on a server equipped with two NVIDIA Tian V GPUs, each with 12 GB of memory, and a CPU with 60 GB of memory. The implementation of *AppSyn* and baselines are based on Python 3.8 and Pytorch 1.10. During the training of *AppSyn*, we set the batch size, epoch, learning rate, and gradient clipping threshold to 16, 25, 5e-4, and 1, respectively. The optimizer is AdamW.

Dataset: We utilize a dataset from a large-scale network operator consisting of 6,102 mobile users and more than 0.26 million raw app usage records within one month.

Metrics: We use several widely-used metrics to evaluate the data fidelity and utility:

- **Data Fidelity**: The Earth Mover Distance (EMD) and Jensen-Shannon divergence (JSD) are used to quantitatively assess the similarity for continuous and categorical attributes between synthetic data and original data. Smaller EMD and JSD values suggest the model can generate synthetic data with higher fidelity.
- **Data Utility**: We use two downstream applications to evaluate data utility, i.e., Top-K app set generation and app preference-based user classification. Jaccard simi-

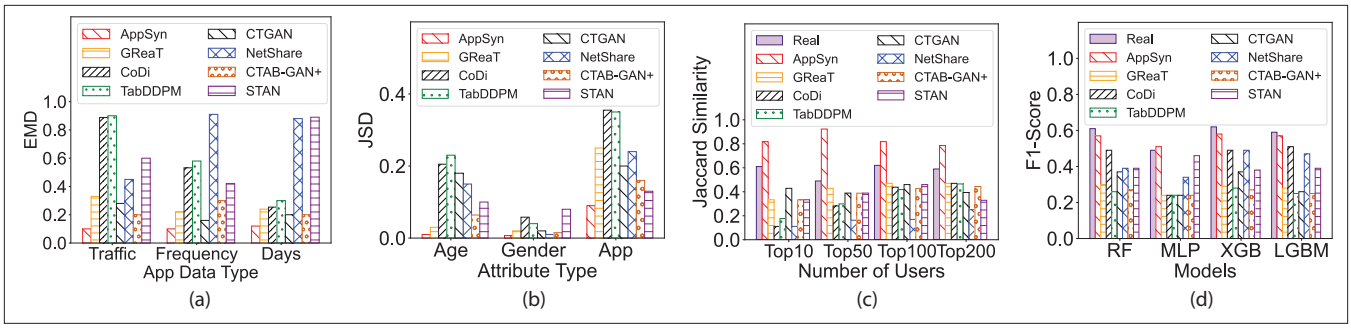

FIGURE 5. (a) and (b) are the overall performance comparison: a) EMD values of continuous attributes; b) JSD values of categorical attributes; (c) and (d) are downstream application performance: c) Jaccard similarity of Top-K application set for advertisers; d) App preference-based user classification.

larity and F1-score are used to evaluate the performance, respectively.

- **Data Privacy**: We use the distance-to-closest record (DCR) to evaluate the distance-based privacy analysis. For each synthetic sample, we compute the minimum Euclidean distance to the nearest real record, and the mean DCR calculates the average distance across all generated samples. A larger DCR typically indicates lower data quality, while a smaller DCR may suggest a potential risk of revealing sensitive information. The privacy evaluation is relative and depends on achieving a balanced approach.

Baselines

We compare *AppSyn* with several state-of-the-art methods to demonstrate the effectiveness.

1. **STAN** [9] is a synthetic network traffic generation method based on auto-regressive model.
2. **TabDDPM** [10] is a denoising diffusion probabilistic-based model for generating synthetic tabular datasets.
3. **NetShare** [11] is a GAN-based model for generating synthetic packet header traces.
4. **GReaT** [12] is an auto-regressive generative LLM to sample synthetic tabular data.
5. **CTGAN** [13] is a conditional GAN for synthetic data generation.
6. **CoDi** [14] is a co-evolving contrastive diffusion model for mixed-type tabular synthesis.
7. **CTAB-GAN+** [15] is a novel conditional tabular GAN.

Overall Performance Evaluation

We compare the data fidelity of *AppSyn* with some state-of-the-art methods. In Fig. 5a and 5b, we show the average EMD and JSD for different attributes. *AppSyn* outperforms baselines regarding EMD and JSD on all features. For example, *AppSyn* achieves an EMD score of 0.1 for traffic consumption, improving EMD by 69.6%, 88.8%, 64.2%, 77.7%, 83.3%, 88.7%, and 50% compared to GReaT, TabDDPM, CTGAN, NetShare, and STAN, CoDi, and CTAB-GAN+. For the app label, *AppSyn* achieves a score of 0.09, outperforming other methods. This superiority is due to *AppSyn*'s specific design, which leverages textual encoding and contextual learning of LLM to accurately model both the statistical distributions and interdependencies of app-related attributes.

Downstream Application Performance

Top-K Application Set for Advertiser: In the advertising domain, the Top-K app set is particularly valuable for advertisers, as it highlights the most popular and frequently used apps among a target audience. By identifying these applications, advertisers can better understand user preferences and behaviors, allowing them to develop more effective and personalized advertising strategies and maximize user engagement. In Fig. 5c, we present the Jaccard similarity of Top-K app sets between real data and the generated data at K values of 10, 50, 100, and 200, respectively. Across all values of K, our proposed method *AppSyn* consistently outperforms baselines, demonstrating a higher overlap with the real Top-K app set.

App Preference-based User Classification: In this application, the classification model classifies the user category based on user and app attributes to support the prediction of user preferences. Since application preference-based user classification is important for personalized recommendation services, we use four classification models to evaluate the performance, including random forest (RF), multi-layer perception (MLP), XGBoost (XGB), and light gradient-boosting Machine (LGBM). In Fig. 5d, in terms of F1-score, real data achieves the highest values, and the synthetic data generated by *AppSyn* outperforms all baselines on classifiers, which closely match real data. These observations underscore the effectiveness of *AppSyn* in accurately capturing user app preferences.

Model	STAN	NetShare	TabDDDPM	CTGAN	GReaT	AppSyn
DCR	3.96e+7	4.69e+7	16497.68	8.93	0.20	74.63

TABLE 1. Privacy analysis.

Privacy Analysis

Table 1 presents the DCR-based privacy analysis. STAN, NetShare, and TabDDPM achieve significantly high values, suggesting that they generate novel records rather than near-duplicates of the real data. In contrast, CTGAN and GReaT exhibit low DCR values, indicating that their synthetic samples closely mimic real data points, posing a higher privacy risk. However, *AppSyn* strikes a balance with an intermediate DCR value, maintaining both data fidelity and privacy.

Discussion

Model Generalizability. Our framework and case study primarily focus on cellular network data synthesis. However, the modular design and underlying principles of our framework are designed to be flexible and extensible, making

it applicable to other mobile network systems with appropriate modifications by incorporating corresponding domain-specific features and constraints. For vehicular networks, the GAN-based trajectory synthesis can be adapted to incorporate road constraints and traffic dynamics. For satellite networks, the diffusion model-based synthesis can be adjusted to model orbital paths and communication delays.

Data Generalizability. The current experiment is limited to one network operator and may introduce certain biases such as user demographics, network configurations, and operational practices, our framework's modular and adaptable design can mitigate these biases. To address the generalizability concern more comprehensively, we plan to pursue the following directions in future work to remain robust and generalizable across various mobile network environments. These include incorporating multi-network operator datasets, cross-operator learning with transfer learning or domain adaptation, and conducting bias analyses and detection.

Conclusion

In this article, we have investigated the challenges of mobile network data synthesis via GAI models. To address these challenges, we have proposed a novel framework consisting of several effective solutions for different mobile network data types, including GAN-based network traffic synthesis, GAN-based cellular association trajectory data synthesis, diffusion model-based vehicular trajectory synthesis, and LLM-based mobile application usage data synthesis. The proposed framework provides a foundation for implementing GAI in effective data synthesis customized for various types of mobile networks. Additionally, we have presented a case study of the LLM-based data synthesis method for mobile application usage data generation. Data-driven evaluations have demonstrated the effectiveness of our method in synthesizing mobile app usage data. Our future work will focus on enhancing generalizability by incorporating diverse datasets, strengthening privacy through advanced techniques, optimizing model efficiency with smaller or compressed models, and expanding downstream applications with real-world validation.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant 62402279, Grant 62422216, and Grant 62320106006, in part by the Central South University Innovation-Driven Research Program under Grant 2023CXQD029, and in part by the 111 Project under Grant B18059.

References

[1] A. Maatouk *et al.*, "Large Language Models for Telecom: Forthcoming Impact on the Industry," *IEEE Commun. Mag.*, 2024.
[2] S. Duan *et al.*, "MoCo: Urban User Mobile Contact Detection Based on Cellular Signaling Trace," *IEEE Trans. Mobile Computing*, 2025.
[3] N. Sehad *et al.*, "Generative AI for Immersive Communication: the Next Frontier in Internet-of-Senses Through 6G," *IEEE Commun. Mag.*, 2024.
[4] H. Lu *et al.*, "FLAMM: Federated Learning Augmented Map Matching with Heterogeneous Cellular Moving Trajectories," *IEEE JSAC*, 2023.
[5] M. Xu *et al.*, "Unleashing the Power of Edgecloud Generative AI in Mobile Networks: A Survey of AIGC Services," *IEEE Commun. Surveys & Tutorials*, 2024.
[6] S. Hui *et al.*, "Knowledge Enhanced GAN for IoT Traffic Generation," *Proc. ACM Web Conf. 2022*, 2022, pp. 3336–46.
[7] H. Chai, T. Jiang, and L. Yu, "Diffusion Model-Based Mobile Traffic Generation with Open Data for Network Planning and Optimization," *Proc. 30th ACM SIGKDD Conf. Knowledge Discovery and Data Mining*, 2024, pp. 4828–38.
[8] R. Toews. Synthetic Data is About to Transform Artificial Intelligence; https://www.forbes.com/sites/robtoews/2022/06/12/synthetic-data-is-about-to-transform-artificial-intelligence.
[9] S. Xu *et al.*, "STAN: Synthetic Network Traffic Generation with Generative Neural Models," *Deployable Machine Learning for Security Defense: 2nd Int'l. Wksp.*, Springer, 2021, pp. 3–29.
[10] A. Kotelnikov *et al.*, "TabDDPM: Modelling Tabular Data with Diffusion Models," *Int'l. Conf. Machine Learning*, PMLR, 2023, pp. 17,564–79.
[11] Y. Yin *et al.*, "Practical GAN-Based Synthetic IP Header Trace Generation Using Netshare," *Proc. ACM SIGCOMM 2022 Conf.*, 2022, pp. 458–72.
[12] V. Borisov *et al.*, "Language Models are Realistic Tabular Data Generators," arXiv preprint arXiv:2210.06280, 2022.
[13] L. Xu *et al.*, "Modeling Tabular Data Using Conditional GAN," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
[14] C. Lee, J. Kim, and N. Park, "CoDi: Co-Evolving Contrastive Diffusion Models for Mixed-Type Tabular Synthesis," *Int'l. Conf. Machine Learning*, PMLR, 2023, pp. 18,940–56.
[15] Z. Zhao *et al.*, "CTABGAN+: Enhancing Tabular Data Synthesis," *Frontiers in big Data*, vol. 6, 2024, p. 1296508.

Biographies

Sijing Duan [M] (duansj@tsinghua.edu.cn) is a postdoctoral fellow with the Department of Computer Science and Technology, Tsinghua University, China. Her research interests include mobile computing, edge computing, and data analysis.

Ye Zhang [M] (yezhang@bistu.edu.cn) is an assistant professor with the Beijing Information Science and Technology University, China. Her research includes edge computing and natural language processing.

Feng Lyu [SM] (fenglyu@csu.edu.cn) is a professor with the School of Computer Science and Engineering, Central South University, China. His research interests include Internet of Things, big data, and edge computing.

Conghao Zhou [M] (zhouconghao@xidian.edu.cn) is a professor with the School of Telecommunications Engineering, Xidian University, China. His research interests include space-air-ground integrated networks, AI for networking, and immersive communications.

Xuemin (Sherman) Shen [F] (sshen@uwaterloo.ca) is a professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research focuses on network resource management, wireless network security, the Internet of Things, 5G and beyond, and vehicular networks. He was the president of IEEE Communication Society.