5G Data Generation for Machine Learning Applications

by

Rania FARJALLAH

MANUSCRIPT-BASED THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT OF A MASTER'S DEGREE
WITH THESIS IN ENGINEERING WITH A PERSONALIZED CONCENTRATION
M.A.Sc.

MONTREAL, "OCTOBER 09, 2025"

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mrs. Bassant Selim, Thesis supervisor
Department of Systems Engineering at Ecole de technologie supérieure

Mr. George Kaddoum, Thesis Co-Supervisor
Department of Electrical Engineering at Ecole de technologie superieure

Mrs. Brigitte Jaumard, Thesis Co-Supervisor
Department of Computer Science and Software Engineering at Concordia University

Mr. Mohamed Cheriet, Chair, Board of Examiners
Department of Systems Engineering at Ecole de technologie supérieure

Mr. Rami Langar, Examiner
Department of Software Engineering and IT at Ecole de technologie supérieure

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON "SEPTEMBER 30, 2025"

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to my research supervisor, Professor Selim Bassant, for her invaluable guidance, support, and encouragement throughout the course of my master's research. Her insightful advice, patience, and motivating discussions have greatly influenced the development of this thesis. I am grateful for the many stimulating conversations that contributed to the progress of this work.

I would like to express my sincere and special thanks to my co-supervisor, Professor George Kaddoum, for his constant encouragement, valuable feedback, and support. His expertise and constructive suggestions have significantly enriched my research experience.

My gratitude also goes to my external supervisor, Professor Brigitte Jaumard, for her valuable input and helpful discussions, which have broadened my perspective and added depth to my project.

I would also like to extend my appreciation to Dr. Samr Ali, Dr. Jean-Michel Sellier, and the members of Ericsson with whom I had the opportunity to interact during this research. Their insights, collaboration, and constructive feedback have been invaluable and have contributed significantly to the progress and quality of my work.

Special thanks to the members of the Synchromedia laboratory for providing a collaborative and inspiring research environment. The support and encouragement of the laboratory members have made my journey more enjoyable and intellectually rewarding. Thanks to all the wonderful people that I met at Synchromedia and that made this experience so satisfactory.

I am profoundly grateful to my family for their unwavering love and encouragement. To my parents, thank you for your endless support, sacrifices, and faith in me throughout this journey and my entire life. To my brother, your moral support and belief in me have been a constant source of strength and motivation.

Towards the end, my gratitude goes to all my friends back home, who made me feel like I never left and provided me with long-distance support, as well as to my friends in Canada for their encouragement and friendship during this journey.

This research was supported by Ericsson – Global Artificial Intelligence Accelerator (GAIA) AI-Hub Canada in Montréal and MITACS Accelerate. The author gratefully acknowledges their financial support.

Génération de données 5G pour les applications d'apprentissage automatique

Rania FARJALLAH

RÉSUMÉ

Les réseaux mobiles de cinquième génération offrent des débits sans précédent, une latence extrêmement faible et une connectivité massive, donnant lieu à des schémas de trafic hautement dynamiques et complexes. L'optimisation de ces réseaux reposent de plus en plus sur l'apprentissage automatique, mais des modèles performants exigent des jeux de données à grande échelle, complets et représentatifs, qui sont rarement disponibles. Pour relever ce défi, ce travail s'appuie sur de vastes jeux de données issus de la mobilité urbaine, sélectionnés en raison de leurs schémas de trafic et de leurs dynamiques temporelles, qui reflètent étroitement ceux observés dans les réseaux 5G. Toutefois, ces jeux de données présentent souvent des observations manquantes, ce qui motive le développement d'une approche d'imputation adaptative, conçue pour reconstruire des séries temporelles incomplètes tout en préservant les structures saisonnières et les dépendances temporelles. Afin d'évaluer la qualité de l'imputation en l'absence des valeurs réelles, de nouvelles métriques sont proposées, ffrant une alternative aux métriques traditionnelles et permettant une évaluation plus robuste et plus informative. Après validation, les jeux de données urbains reconstruits sont transformés en traces préliminaires de trafic 5G, constituant une base structurée pour des analyses et modélisations ultérieures. Sur cette base, un cadre de génération de trafic, fondé sur le Principe de Maximum d'Entropie, est proposé ; il encode des contraintes empiriquement observées afin de produire des données synthétiques réalistes qui capturent fidèlement les dynamiques temporelles et les propriétés distributionnelles. Cette approche s'avère efficace dans des scénarios réels où l'accès à des données de trafic 5G de haute qualité est limité en raison de contraintes liées à la confidentialité, de restrictions opérationnelles ou de mesures incomplètes. En générant des données synthétiques représentatives et statistiquement cohérentes, le cadre proposé facilite l'amélioration des prévisions, la détection d'anomalies et l'évaluation des performances du réseau, permettant ainsi aux opérateurs de télécommunications et aux chercheurs de concevoir, tester et optimiser plus efficacement les systèmes mobiles de nouvelle génération. Des expériences approfondies démontrent que le cadre proposé améliore la qualité des données, favorise la modélisation réaliste du trafic 5G et permet des prévisions plus robustes, une meilleure détection des anomalies et une évaluation plus fiable des performances des réseaux de prochaine génération.

Mots-clés: Génération de trafic 5G, Imputation de séries temporelles, Génération de données synthétiques, Principe d'entropie maximale, Divergence de Jensen–Shannon, Distance de Wasserstein, Score discriminatif, TimeGPT

5G Data Generation for Machine Learning Applications

Rania FARJALLAH

ABSTRACT

Fifth-generation mobile networks enable unprecedented throughput, ultra-low latency, and massive device connectivity, giving rise to traffic patterns that are highly dynamic and complex. Optimizing such networks increasingly relies on machine learning, yet effective models require large-scale, complete, and representative datasets. To address this challenge, this work relies on large-scale urban mobility datasets, chosen for their traffic patterns and temporal dynamics, which closely reflect those observed in 5G networks. However, these datasets often suffer from missing observations, which motivates the development of an adaptive imputation approach designed to reconstruct incomplete time series while preserving seasonal structures and temporal dependencies. To evaluate imputation quality where ground truth is unavailable, novel metrics are introduced, providing an alternative to traditional metrics and enabling a more robust and informative evaluation. After validation, the reconstructed urban datasets are transformed into preliminary 5G traffic traces, providing a structured foundation for further modeling and analysis. Building on this foundation, a principled traffic generation framework based on the Maximum Entropy Principle is proposed, which encodes empirically observed constraints to produce realistic synthetic data that accurately capture temporal dynamics and distributional properties. This approach is particularly effective in real-world scenarios where access to high-quality 5G traffic data is limited due to privacy concerns, operational constraints, or incomplete measurements. By generating representative and statistically consistent synthetic data, the framework facilitates improved forecasting, anomaly detection, and network performance evaluation, enabling telecommunication operators and researchers to design, test, and optimize next-generation mobile systems more effectively. Extensive experiments confirm that the proposed framework improves data quality, supports realistic 5G traffic modeling, and enables more robust forecasting, anomaly detection, and network performance evaluation in next-generation mobile systems.

Keywords: 5G traffic generation, time series imputation, synthetic data generation, Maximum Entropy Principle, Jensen-Shannon divergence, Wasserstein distance, Discriminative Score, TimeGPT

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS AND ACRONYMS

5G	Fifth Generation Mobile Communication
AIC	Akaike Information Criterion
AI	Artificial Intelligence
AR	Autoregressive
ARIMA	Auto Regressive Integrated Moving Average
ASC	Agence Spatiale Canadienne
AutoML	Automated Machine Learning
BFGS	Broyden–Fletcher–Goldfarb–Shanno
BIC	Bayesian Information Criterion
CART	Classification and Regression Tree
CDF	Cumulative Distribution Function
CNNs	Convolutional Neural Networks
DL	Deep Learning
eMBB	Enhanced Mobile Broadband
EM	Expectation Maximization
ETS	École de Technologie Supérieure
GASTN	Graph Attention Spatial-Temporal Network
GAN	Generative Adversarial Network
GCN	Graph Convolutional Networks

IoT	Internet of Things
JSD	Jensen-Shannon Divergence
KDE	Kernel Density Estimation
KL	Kullback-Leibler
KNN	K-Nearest Neighbors
LSTM	Long Short-Term Memory
MA	Moving Average
MAE	Mean Absolute Error
MEP	Maximum Entropy Principle
MI	Multiple Imputation
ML	Machine Learning
MLP	Multi-Layer Perceptron
mMTC	Massive Machine-Type Communications
MSE	Mean Squared Error
NLL	Negative Log-Likelihood
POIs	Points of Interest
RMSprop	Root Mean Square Propagation
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SARIMA	Seasonal Auto Regressive Integrated Moving Average

SGD	Stochastic Gradient Descent
SOM	Self Organizing Maps
TKCM	Top-K Case Matching
VAE	Variational Autoencoder
WD	Wasserstein Distance

INTRODUCTION

The 5G wireless network introduces a transformative era for digital communication, characterized by unprecedented data rates, low latency, and support for a vast ecosystem of Internet of Things (IoT) devices. This evolution in communication technology enables new and complex applications, including real-time analytics, smart city management, autonomous vehicles, and remote healthcare services (Li *et al.* (2022b); Li, Zheng, Xiao & Wang (2023b); Chen, Wang, Wang, Li & Ren (2023)). The infrastructure of 5G networks is designed to accommodate these high demands, but it also introduces significant challenges related to data complexity, volume, and heterogeneity. Managing this data efficiently is critical, particularly with respect to quality, continuity, and accuracy, since reliable information is essential for optimizing network performance and resource allocation. Given the complexity and heterogeneity of 5G traffic, machine learning (ML) has become an indispensable tool for understanding traffic behavior, predicting network demand, detecting anomalies, and optimizing resource allocation (Fourati, Maaloul & Chaari (2021)). However, ML models require large-scale, complete, and representative datasets for training and evaluation. In practice, real-world 5G datasets are scarce due to privacy regulations, proprietary restrictions, and infrastructure limitations. To overcome this limitation, this work leverages urban mobility datasets as a proxy for 5G traffic, motivated by the fact that their temporal dynamics, periodicity, and statistical properties closely resemble those observed in real 5G network traffic. These datasets allow us to simulate realistic network dynamics and evaluate models where direct access to extensive 5G traces is not feasible.

When working with urban mobility datasets, the challenge of missing data emerges. These datasets often suffer from missing observations caused by sensor failures, device malfunctions, intermittent connectivity, or collection errors (Miao, Wu, Chen, Gao & Yin (2023)). For example, in Montreal urban traffic dataset[1], more than 50% of the collected data is missing due

[1] Traffic and Pedestrian Counting Dataset of Montreal City : https://donnees.montreal.ca/dataset/comptage-vehicules-pietons.

to sensor outages and communication failures. These gaps present a critical issue for time-series analysis, where each observation typically depends on previous values to capture meaningful trends and seasonal patterns. Missing data disrupts these dependencies, making it difficult for predictive models to deliver accurate results. For example, in network management and traffic forecasting, missing data can lead to inaccurate predictions of network load, inefficient resource allocation, and degraded service quality, especially under high-demand conditions.

Traditionally, missing data in time series has been handled using techniques such as listwise deletion or simple imputation methods like forward and backward filling (Saad, Chaudhary, Karray & Gaudet (2020)). While these approaches are straightforward and computationally efficient, they often fail to capture the complex temporal structures, periodicity, and burstiness that characterize modern network and mobility datasets. More advanced imputation strategies (Bandara, Bergmeir & Hewamalage (2021)), including statistical models and machine learning algorithms, provide improvements by leveraging hidden patterns within the data to generate better estimates. Nevertheless, these methods also have limitations, especially when dealing with long or sequential gaps where retaining the underlying seasonal and trend-based patterns is essential for accurate reconstruction.

In addition to the problem of missing data, another significant challenge in 5G traffic analysis is the generation of realistic synthetic data. Since access to large-scale, high-quality 5G datasets is limited, synthetic traffic generation seeks to produce artificial pattern that maintain the statistical, temporal, and structural properties of real-world traffic while bypassing privacy and accessibility issues. However, existing generative approaches such as GANs, VAEs (Yin, Lin, Jin, Fanti & Sekar (2022); Xu *et al.* (2022)), and diffusion-based models, while widely used, tend to be less effective when applied to 5G traffic modeling. A key limitation of these models is their dependence on extensive volumes of training data to learn complex traffic patterns reliably. This requirement is often incompatible with the nature of 5G datasets, which are

typically scarce, noisy, or restricted due to privacy and regulatory constraints. As a result, there is growing interest in exploring alternative methods that can better accommodate the limitations of real-world 5G datasets while still capturing essential traffic characteristics.

This thesis addresses these challenges by employing an effective consecutive filling strategy to handle missing data, ensuring temporal consistency in reconstructed time series. Furthermore, alternative evaluation metrics are introduced to evaluate the quality of imputed data when ground truth is unavailable, which is often the case in real-world datasets. As a result, conventional metrics such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) are insufficient for a comprehensive evaluation. To overcome this limitation, this work explores statistical measures, enabling a more robust evaluation of imputation performance. After validation, the reconstructed urban datasets are transformed into preliminary 5G traffic traces, providing a structured foundation for further modeling and analysis. Building on this foundation, the thesis presents a Maximum Entropy Principle (MEP)-based traffic generation framework that encodes empirically observed constraints to generate realistic synthetic 5G traffic while accurately capturing temporal dynamics and distributional properties of real-world observations.

This manuscript-based thesis is structured into four chapters. Most chapters consist of published articles or papers in refereed scientific journals or conferences. The contents of each chapter is almost the same as that of the published paper, with minor modifications for consistency of notation throughout the thesis. Therefore, an overlap could not be avoided. Although each chapter may be read independently, it is recommended to read the thesis sequentially.

The first chapter provides an introduction to traffic generation and an overview of existing imputation techniques, along with the motivation, problem statement, and contributions of this work. Chapter 2 presents the design and validation of two no–ground-truth evaluation metrics—the Wasserstein distance and the Jensen–Shannon divergence—for assessing time-series imputation, together with a comparative study on two real world datasets. Chapter 3

describes an iterative gap-filling approach and a broader benchmark that includes TimeGPT in addition to ARIMA, SARIMA, XGBoost, and LSTM, and introduces three validation metrics. It also reports a real-world case study on the Montreal dataset. Chapter 4 presents a Maximum-Entropy–based traffic generation framework that encodes empirical constraints and details its optimization strategy.

CHAPTER 1

GENERAL CONTEXT OF THE PROJECT

1.1 Introduction

This chapter provides the general context of the project and introduces the key concepts related to 5G traffic analysis. It begins with an overview of 5G traffic generation, highlighting the different service categories and discussing how they contribute to heterogeneous and dynamic traffic patterns. It then examines the issue of missing data in time series, exploring common causes, traditional imputation techniques, and recent deep learning-based approaches, along with their limitations. Next, the chapter presents the motivation and problem statement, focusing on the challenges posed by incomplete datasets, traffic variability, and the growing need for realistic synthetic data in modern 5G environments. Finally, the objectives and contributions of this work are introduced, summarizing the solutions proposed to address these challenges.

1.2 Overview of 5G Traffic Generation

The evolution of 5G networks introduces a new generation of mobile infrastructure that supports a broad spectrum of service categories. These include enhanced Mobile Broadband (eMBB), massive Machine-Type Communications (mMTC), and Ultra-Reliable Low-Latency Communications (URLLC), each exhibiting unique traffic profiles and usage dynamics. Unlike earlier generations, where traffic behavior was relatively uniform, modern mobile networks must now support highly heterogeneous traffic flows (Li *et al.* (2022b); Chen *et al.* (2023)). This multifaceted nature of 5G has introduced new challenges in modeling, simulating, and managing network traffic.

Traditional traffic models, which typically relied on simplified statistical assumptions or average-case approximations, fall short in capturing the temporal granularity and spatial heterogeneity now characteristic of 5G-enabled applications (Ding, Schober & Poor (2021); Jin *et al.* (2022)). These approaches often assumed stationarity and homogeneity across users and services, thereby

limiting their effectiveness in reproducing the dynamic, bursty, and highly context-dependent traffic patterns seen in modern networks. As network conditions become more variable and application demands more diverse, there is an urgent need for advanced modeling techniques capable of representing realistic user behaviors and service-specific requirements.

To address these complexities, recent research has shifted towards data-driven traffic generation that leverages large-scale empirical datasets collected from operational mobile networks (Lu, Zhang, Li, Jiang & Abbas (2021)). However, a major limitation of these approaches is the scarcity of publicly available large-scale mobile network datasets. In many cases, real-world traces are proprietary or restricted due to privacy and regulatory constraints, which makes comprehensive data difficult to obtain. As a result, existing works are often trained on limited datasets that fail to capture long-term temporal patterns and the full variability of real network dynamics. These real-world datasets, when available, do offer valuable spatiotemporal insights into mobile usage behavior and network load variations. By analyzing these patterns, researchers are able to construct generative models that produce synthetic traffic reflecting the nuanced statistical structure of actual usage. Recent efforts employ deep learning methods such as Generative Adversarial Network (GAN), Variational Autoencoder (VAE), and diffusion models to learn these pattern, enabling the simulation of traffic traces that exhibit realistic variations in time, volume, and location (Yin *et al.* (2022); Xu *et al.* (2022)). In a related direction, (Ziazet, Jaumard, Duong, Khoshabi & Janulewicz (2022)) proposed a dynamic traffic generator that utilizes open urban mobility datasets from the City of Montreal to construct synthetic 5G traffic traces. By transforming vehicular and pedestrian counts into multiple classes of 5G network slices, their work demonstrated the potential of leveraging large-scale urban data to capture realistic temporal dynamics and peak-hour variations observed in 5G usage.

Nevertheless, such generative models still face limitations. For instance, capturing the multi-scale periodicity of real-world traffic (e.g., hourly surges, daily cycles, or weekend shifts) remains a persistent challenge (Zhang (2023); Li, Braud, Li & Hui (2021)). Moreover, the spatial structure of urban environments shaped by factors such transportation networks, and base station placement, often leads to regional correlations in traffic that are inadequately represented

in purely neural architectures (Gong, Jia & Li (2022)). Beyond these technical limitations, accessibility to high-quality traffic data is constrained by privacy regulations and infrastructure variability, further complicating model training and validation.

1.3 Data Filling in Time Series

Real-world datasets often suffer from substantial amounts of missing data caused by various factors, including sensor malfunctions, GPS inaccuracies, device failures, and data collection errors. These gaps are particularly prevalent in measurements collected through distributed sensing infrastructures, where data acquisition relies on multiple heterogeneous sources with varying levels of reliability. In large-scale monitoring systems, such as those used for traffic analysis and mobility tracking, missing data can occur intermittently or persist over extended periods due to network outages, hardware limitations, or communication delays. The presence of missing values compromises the integrity of time series data, distorts temporal patterns, and introduces substantial challenges for downstream tasks such as traffic forecasting, anomaly detection, and synthetic data generation (Pratama, Permanasari, Ardiyanto & Indrayani (2016); Song & Szafir (2019)).

Although traditional imputation techniques such as forward/backward filling, linear interpolation, and statistical models like ARIMA and SARIMA offer computational simplicity, they struggle to recover the complex, non-stationary patterns found in 5G traffic (Saad *et al.* (2020)). These methods often fail to model bursty or multi-modal behaviors and tend to oversimplify underlying temporal dependencies. As a result, imputed values may introduce systematic biases, particularly in scenarios with high variability or dense service-layer activity.

To improve imputation accuracy, several deep learning-based models have been proposed, including LSTMs, MLPs, and Transformers (Che, Purushotham, Cho, Sontag & Liu (2018)). These models can learn intricate time-dependent structures and offer more accurate estimations under challenging conditions. However, they also come with limitations: they are typically

data-hungry, sensitive to hyperparameters, and often operate as black-box systems, which reduces their interpretability.

A particularly difficult aspect of real-world time series imputation lies in the absence of ground truth. Since the true values corresponding to missing intervals are rarely available, evaluating the performance of imputation models in practice is particularly challenging. Most existing studies simulate missing data by artificially removing known values to construct benchmark scenarios; however, this strategy fails because, in real-world datasets, the actual missing values are unknown and cannot be used to validate the imputation. Even if artificially removed gaps are made irregular, this does not replicate the real challenge, which lies in evaluating imputation quality when no ground truth exists. As a result, such benchmarks cannot fully represent the structural dependencies and uncertainty associated with real-world missing data (Saad *et al.* (2020)).

1.4 Motivation and Problem Statement

One of the major challenges in 5G traffic prediction is the lack of realistic, large-scale datasets spanning long time periods. Due to privacy concerns, proprietary restrictions, and infrastructure limitations, publicly available datasets that fully capture the dynamics of mobile networks are extremely scarce. To address this limitation, this work relies on urban mobility datasets as a proxy for 5G traffic, motivated by their similar temporal dynamics and statistical properties. However, these datasets introduce a critical challenge: they often contain significant amounts of missing data caused by sensor failures, GPS inaccuracies, connectivity issues, and data collection errors.

Handling these missing values is essential before generating realistic datasets for 5G traffic modeling. Gaps in the data distort temporal dependencies, bias statistical distributions, and compromise the performance of downstream tasks such as traffic forecasting, anomaly detection, and network optimization. Existing imputation techniques remain impractical, as they are often evaluated using ground truth data and fail to capture the non-stationary, highly variable,

and periodic nature of traffic data. On the other hand, deep learning-based approaches, while potentially more accurate, require extensive amounts of training data and often lack interpretability, making them unsuitable in scenarios where datasets are sparse or fragmented. The challenge becomes even greater when the missing intervals are long or irregular, making it difficult to reconstruct the underlying patterns faithfully. Once the missing data is addressed, the goal is to generate a realistic and statistically consistent 5G traffic dataset. Since no large-scale real-world traces are publicly available, producing such datasets is essential for developing and evaluating forecasting models, anomaly detection techniques, and network optimization strategies.

In summary, this study focuses on two interconnected research problems in 5G traffic generation. The first concerns the challenge of reconstructing incomplete urban datasets in the presence of irregular and long-term gaps, where existing imputation methods are inadequate and reliable evaluation is difficult due to the absence of ground truth. The second concerns the difficulty of generating realistic synthetic datasets when no comprehensive 5G traffic traces exist and current generative techniques fail to capture the temporal dynamics and distributional structure of real traffic. These challenges must be addressed to enable more accurate traffic analysis, support effective forecasting, and improve the evaluation of next-generation mobile network performance.

1.5 Objectives and Contributions

The primary objective of this study is to generate large-scale and realistic 5G traffic datasets that can support advanced analytics, machine learning models, and network performance evaluation. Achieving this goal requires addressing two subobjectives.

The first subobjective is to handle missing data in urban mobility datasets used as a proxy for 5G traffic. Since these datasets often contain incomplete observations due to sensor failures, GPS inaccuracies, or data collection errors, it is essential to reconstruct reliable and continuous time series while preserving temporal dependencies and structural patterns. To enable a reliable

assessment of imputation quality, we also propose three novel evaluation metrics that measure the consistency between imputed values and observed data distributions. These metrics provide a robust alternative to traditional error-based measures, such as RMSE and MAE, and allow for accurate evaluation even when ground truth values are unavailable.

The second subobjective is to propose a data-driven approach for generating realistic synthetic traffic based on the Maximum Entropy Principle (MEP). The aim is to create synthetic datasets that reproduce the temporal dynamics and statistical properties of real-world traffic, enabling more accurate forecasting, anomaly detection, and resource optimization in mobile networks.

Together, these contributions establish a comprehensive foundation for improving traffic analysis, forecasting, and performance evaluation in next-generation mobile networks.

1.6 Conclusion

This chapter provided an overview of the general context surrounding 5G traffic generation and highlighted the growing complexity introduced by modern mobile environments. We examined the challenges posed by heterogeneous traffic patterns, high data variability, and incomplete measurements, which significantly affect forecasting accuracy, anomaly detection, and network performance evaluation. By establishing this context, the chapter sets the foundation for the subsequent sections, where we explore noval methodologies for addressing these challenges. The following chapters will introduce data-driven techniques for handling missing values and generating realistic synthetic traffic, aiming to improve the accuracy, reliability, and adaptability of 5G traffic modeling.

CHAPTER 2

EVALUATION OF MISSING DATA IMPUTATION FOR TIME SERIES WITHOUT GROUND TRUTH

Rania Farjallah[1] , Bassant Selim[1] , Brigitte Jaumard[2] , Samr Ali[3] , Georges Kaddoum[4,5]

[1] Department of Systems Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[2] Department of Computer Science and Software Engineering (CSSE), Concordia University,
1455 De Maisonneuve Blvd. W. Montreal, Québec, Canada H3G 1M8
[3] GAIA, AI Hub Canada, Ericsson,
8275 Trans Canada Route, Saint-Laurent, Quebec, Canada H4S 0B6
[4] Department of Electrical Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[5] Artificial Intelligence & Cyber Systems Research Center, Lebanese American University,
Beirut 03797751, Lebanon

ARTICLE PRESENTED IN IEEE International Conference on Communications 2025 (ICC 2025), June 2025

2.1 Introduction

As data traffic and application demands surge with the deployment of 5G networks, communication systems face unprecedented challenges in complexity. Applications like real-time video streaming, autonomous vehicles, and IoT devices require high data throughput, ultra-low latency, and reliable, adaptive resource management. To meet these demands, machine learning (machine-learning (ML)) and deep learning (DL) techniques have become indispensable for optimizing network performance. These models enable adaptive traffic forecasting, real-time bandwidth allocation, efficient energy management, and proactive congestion control, which identifies and mitigates network bottlenecks to prevent performance degradation Jaumard & Ziazet (2023); Sarkar & Debnath (2021). However, the accuracy of these ML-driven optimizations depends heavily on high-quality, complete data—an ideal that is often unattainable, as real-world datasets are typically riddled with missing values or inconsistencies. Consequently, effective data filling strategies are essential to ensure that ML models function reliably and unlock the full potential of 5G network capabilities. While imputation might seem similar to prediction, it differs in the

availability of ground truth. Prediction forecasts future values using observed data, whereas imputation reconstructs missing values without access to ground truth.

In practice, 5G network datasets, such as performance measurement counters, frequently contain missing values due to factors like system failures, storage constraints, and synchronization issues within the network's data collection infrastructure Miao *et al.* (2023). These data gaps can undermine the accuracy of ML models used for network management tasks, such as traffic forecasting and resource allocation, thereby impacting overall network performance. Traditional imputation methods, such as deletion or simple filling techniques, often fall short because they do not account for the dependencies and correlations among metrics in 5G network data. Advanced ML-based imputation methods offer more accurate results by modeling these relationships; however, they introduce a new challenge: the need for reliable validation mechanisms to ensure that imputed values faithfully represent the missing data. Existing imputation methods typically assume access to ground truth data to validate their performance, but in data-filling tasks, this original data is inherently missing. Without ground truth, assessing the accuracy of imputed values becomes challenging, highlighting the need for alternative metrics to evaluate model performance effectively.

This chapter explores the use of statistical tools as validation metrics for evaluating imputation performance. These tools measure differences or distances between distributions, a common approach in synthetic data evaluation. By comparing distributional differences between imputed data and the original data, they provide a practical method for assessing imputation accuracy without relying on ground truth. To validate the proposed metrics, we apply them to complete datasets that share characteristics with incomplete 5G network data.

The remainder of this chapter is organized as follows. Section 2.2 describes related works. Section 2.3 presents the proposed metrics. Section 2.4 illustrates the datasets and the validation methodology. Section 2.5 discusses the experimental settings and results, and Section 2.6 concludes the chapter and provides future research directions.

2.2 Related Works

The quality and completeness of datasets are essential for effective analysis, especially in time series applications where missing values can obscure key patterns, such as seasonality, and reduce predictive accuracy. Seasonality, characterized by periodic fluctuations over time (e.g., daily or weekly patterns), can critically impact model performance by introducing systematic variations that imputation methods must account for. However, many imputation methods do not incorporate seasonality, relying only on remaining observed data, which simplifies modeling but reduces accuracy.

For small gaps, basic interpolation methods are commonly used, fitting smooth curves between known data points to estimate missing values. While simple, these methods fail to capture temporal dependencies and can lead to biased results. Other approaches include single-imputation techniques, such as Hot Deck, Cold Deck, and expectation maximization (EM), which replace each missing value with a single estimate but may not reduce bias effectively Donders, van der Heijden, Stijnen & Moons (2006). In contrast, multiple imputation (MI) techniques offer advantages by providing information on how missing data impacts parameter estimates Donders *et al.* (2006). Advanced methods, such as regression-based imputation, self organizing maps (SOM) Junninen, Niska, Tuppurainen, Ruuskanen & Kolehmainen (2004), and K-nearest neighbors (KNN), have proven more effective, particularly in datasets where temporal relationships are significant. KNN, for instance, fills missing values by identifying the k-closest patterns around the missing data point, utilizing local similarity to improve imputation accuracy Tarsitano & Falcone (2011).

Incorporating seasonality into imputation methods can markedly improve accuracy. Techniques such as seasonal adjustment with Kalman filters and linear interpolation on seasonally decomposed data, available in tools like the forecast and zoo R-packages, have proven useful Moritz, Sardá-Espinosa, Bartz-Beielstein, Zaefferer & Stork (2015). Additionally, seasonal auto regressive integrated moving average (SARIMA) has been applied to seasonal time series, although it struggles with consecutive missing values Sutiene, Vilutis & Sandonavicius

(2011). More advanced approaches include neural network-based methods, such as multi-layer perceptron (MLP) Wijesekara & Liyanage (2023) and long short-term memory (LSTM) networks, as well as hybrid neural models Bandara *et al.* (2021). Pattern-based methods, like the top-k case matching (TKCM) algorithm, have also been used to handle missing data Wellenzohn, Böhlen, Dignös, Gamper & Mitterer (2017). However, many of these methods are limited to single-seasonal patterns and are not well-suited for multiple seasonalities.

All imputation validation approaches rely on ground truth data to evaluate accuracy. Typically, studies create artificial gaps in complete datasets, then fill these gaps and compare the imputed values to the known originals. This approach enables traditional metrics like RMSE and MAE to measure a model's ability to accurately reconstruct missing data. However, this validation strategy assumes that ground truth data is available—an assumption that often does not hold in real-world applications, especially in dynamic environments such as 5G networks.

This chapter addresses this gap by introducing two statistical metrics to evaluate imputation methods without relying on ground truth. These metrics assess how closely the distributions of imputed data align with those of the original data, providing a way to evaluate imputation effectiveness based on internal structure and data consistency. By adapting these metrics, we offer a set of tools that complements existing validation approaches and extends evaluation capabilities to real-world applications where complete datasets are not available.

2.3 Proposed Data Filling Validation Metrics

To validate the performance of data imputation methods in the absence of ground truth, we propose two statistical metrics, namely the wasserstein distance (WD) and the jensen-shannon divergence (JSD), as described in this section, to measure distributional similarity. These metrics, originally used for synthetic data validation Stenger, Leppich, Foster, Kounev & Bauer (2024), allow us to evaluate how closely the imputed data match the pre-gap distribution. Table 2.1 provides a summary of the no ground truth and traditional ground truth metrics considered in this work.

Table 2.1 Comparison of no ground truth and ground truth metrics

Metric type	**No ground truth metrics**	**Ground truth metrics**
Metrics	Wasserstein distance (WD), Jensen-Shannon divergence (JSD)	Root mean squared error (RMSE), Mean absolute error (MAE)
Ground truth requirement	Not required, uses pre-gap data as reference	Requires actual ground truth values for accuracy assessment
Evaluation principle	Evaluates distributional and structural alignment between imputed and pre-gap data	Measures direct error by comparing imputed values to true values
Interpretation	Lower values indicate better alignment with original data distribution	Lower values indicate more accurate gap filling compared to true values

2.3.1 Wasserstein Distance

The WD, also known as the earth mover's distance, measures the dissimilarity between two probability distributions by calculating the minimum effort required to transform one distribution into another Villani (2009). Formally, the $W(P, Q)$ between two distributions P and Q over a metric space $\mathcal{X}$ is defined as:

$$W(P, Q) = \inf_{\gamma \in \Gamma(P,Q)} \int_{\mathcal{X} \times \mathcal{X}} \|x - y\| \, d\gamma(x, y), \tag{2.1}$$

where $\Gamma(P, Q)$ represents the set of all possible joint distributions (or couplings) γ with marginals P and Q. In this context, $\|x - y\|$ quantifies the distance between points x and y in the space $\mathcal{X}$. The WD gives the minimum "cost" required to "move" probability mass from the distribution P to match distribution Q, with "cost" referring to the product between the probability mass and the distance it must be moved. This metric is particularly suitable for evaluating imputed data without ground truth because it directly compares the distribution of pre-gap values with that of the imputed values. A smaller WD means that the imputed data closely aligns with the

patterns of the pre-gap values, showing that the imputation method maintains the original data characteristics well Stenger *et al.* (2024).

2.3.2 Jensen-Shannon Divergence

The Jensen-Shannon Divergence is a statistical measure for quantifying the similarity between two probability distributions Stenger *et al.* (2024). It is based on the kullback-leibler (KL) divergence, which measures the divergence of one probability distribution P from a reference distribution Q. However, KL divergence presents two limitations that make it unsuitable for evaluating imputation techniques. Firstly, KL divergence is asymmetric, meaning that $D_{KL}(P\|Q) \neq D_{KL}(Q\|P)$, which can lead to biased comparisons depending on the order in which the distributions are considered. Second, KL divergence can yield infinite values if there are points in the support of P that have zero probability in Q. This sensitivity to non-overlapping supports can lead to instability, producing values that are disproportionately large or undefined, even when the distributions are similar in other regions. Such behavior makes KL divergence unreliable for assessing the similarity between pre-gap and imputed distributions. Unlike the KL divergence, JSD is symmetric, making it more robust for comparing distributions. The JSD between distributions P and Q is defined as:

$$JS(P\|Q) = \frac{1}{2}D_{KL}(P\|M) + \frac{1}{2}D_{KL}(Q\|M), \tag{2.2}$$

where $M = \frac{1}{2}(P + Q)$ represents the average distribution, and D_{KL} is the KL divergence. A JSD close to 0 indicates high similarity between distributions, signifying that the synthetic data closely replicates the real data's statistical properties.

2.4 Metrics Validation Methodology

This section outlines our approach for validating the proposed no ground truth metrics to ensure their suitability for evaluating imputation quality.

2.4.1 Validation Methodology

To ensure our approach is both effective and reliable, we introduce a validation process for the proposed metrics. By validating these metrics, we aim to confirm that, in the absence of ground truth data, they accurately reflect the performance of different imputation methods and can be considered a reliable alternative to traditional validation metrics.

The methodology begins by validating the proposed no ground truth metrics using two complete datasets. To simulate real-world scenarios, we artificially create N gaps of different lengths at random positions within the datasets. This approach allows us to compare the gap-filled values with the true held-out observations as well as analyze and compare the performance of different gap-filling methods using both ground truth and the no ground truth metrics. The gaps are filled using three different imputation methods: an interpolation-based approach, a ML-based approach, and a DL-based approach.

After filling the gaps with each method, we evaluate the accuracy of the proposed metrics by calculating and comparing them with traditional metrics, namely the RMSE and MAE, which use the original data as ground truth. For ground truth-based metrics, we compare the gap-filled values directly to the true observations. In contrast, for the proposed metrics, we use pre-gap values as a reference. The reason behind this is that we do not have ground truth data in practice, and as shown in Figure 3.2, when considering the same gap size, the distribution of pre-gap values closely matches that of the true values. This similarity in distribution ensures that the pre-gap segment serves as an appropriate proxy for evaluating how well the gap-filled data aligns with the original data distribution, without needing access to true values. This approach enables us to evaluate whether the statistical metrics can effectively measure how well the gap-filled data aligns with the general data distribution, without requiring the ground truth.

By demonstrating that the results from both types of metrics are closely aligned, we establish that the no ground truth metrics can serve as reliable alternatives for traditional metrics.

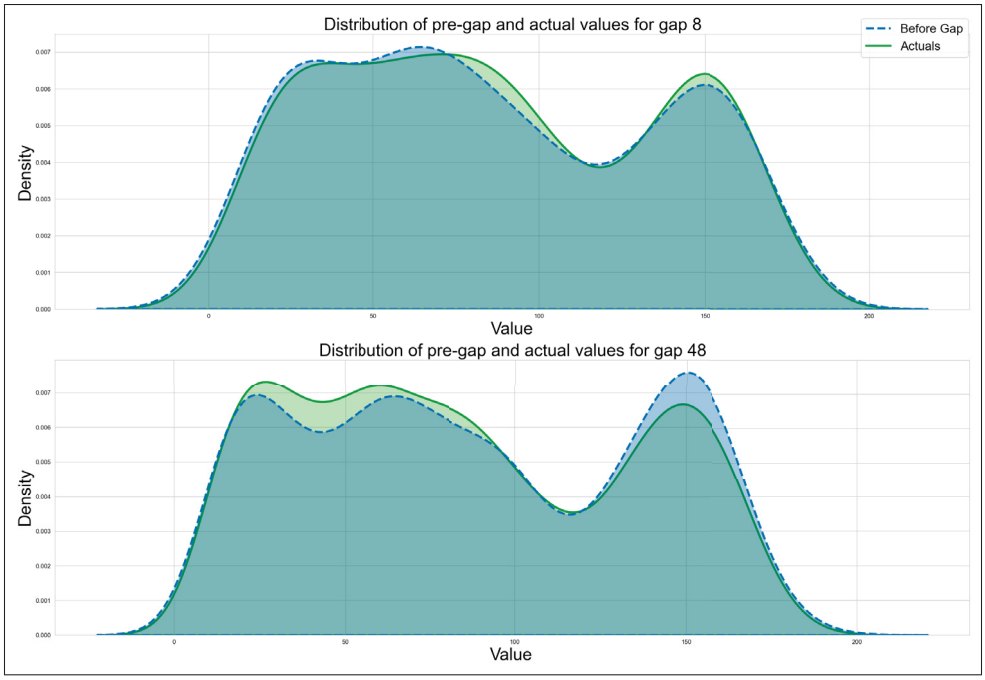

Figure 2.1 Distribution of the pre-gap and actual values

2.4.2 Description of the Datasets

We considered two datasets to validate the proposed metrics. These datasets consist of seasonal time series data, capturing vehicle counts across various transportation modes. Since the datasets are relatively complete, they are well-suited for validating the proposed metrics against ground truth based metrics.

2.4.2.1 Telraam Dataset

This dataset tracks the movement of pedestrians, cyclists, cars, and heavy vehicles every hour during the day period. Although Telraam gathers data from various cities, we selected Brussels for its diverse traffic patterns, also due to low missing rate in this particular intersection. Data are collected with cameras installed as part of the Telraam device [1] recording data from October

[1] Traffic Count Data : https://telraam.net/

2021 to January 2024. Data can be accessed in real-time via the Telraam API. The missing rate in this dataset is under 1% over all the captured period.

2.4.2.2 Madrid Dataset

This dataset provides historical and real-time traffic data in the city of Madrid with a frequency of 15 min [2]. The data is recorded from July 2013 to October 2024 with a missing data of 0%. The dataset captures the intensity of car traffic across different segments.

2.5 Numerical Results

In this section, we demonstrate that gap-filling approaches can be effectively evaluated using both ground truth and no ground truth metrics. By comparing gap-filled values to true observations and pre-gap values, we evaluate the consistency of these metrics.

2.5.1 Experimental Setup

In this study, we optimized different imputation models to accurately fill missing values in time series data. This section details the training set, model optimization techniques, and feature selection used across different imputation methods.

2.5.1.1 Interpolation-Based Imputation

Polynomial interpolation is applied directly to the available hourly data to estimate missing values. This method fits a flexible curve through known data points, ensuring continuity in the imputed data.

---

[2] City of Madrid : https://datos.madrid.es/portal/

2.5.1.2 ARIMA-Based Imputation

The auto regressive integrated moving average (ARIMA) model addresses missing values by modeling both autoregressive (AR) and moving average (MA) components, capturing trends and autocorrelations in data. For both datasets, ARIMA is trained on six weeks of data. Using the `auto_arima` function, the parameters p, d, and q are optimized automatically to represent the autoregressive, differencing, and moving average components of the model.

2.5.1.3 SARIMA-Based Imputation

The SARIMA model extends ARIMA by including seasonal terms to capture periodic patterns in data. SARIMA followed the same training durations as ARIMA, with six weeks for both datasets, incorporating seasonal terms. The `auto_arima` function optimizes parameters p, d, and q for non-seasonal components, as well as P, D, Q, and s for seasonal terms, with s representing the seasonal frequency.

2.5.1.4 XGBoost-Based Imputation

XGBoost is an advanced boosting algorithm combining multiple weak learners to improve prediction accuracy iteratively. The model is trained on two years of data for the Madrid dataset and one year for the Telraam dataset. Additionally, we optimize four hyperparameters using GridSearch, including `max_depth`, `learning_rate`, `n_estimators`, and `subsample`, to enhance model accuracy and prevent overfitting. To capture complex temporal patterns, XGBoost uses time-based features namely *simple moving average (SMA)*, *exponentially weighted moving average (EWMA)* and *hour of day*.

2.5.1.5 LSTM-Based Imputation

LSTM networks, a type of recurrent neural network (RNN) optimized for sequence prediction, are used to address missing values in time series data. LSTM is trained on two years of data for the Madrid dataset and one year for the Telraam dataset, with look-back periods of 20 hours for

Madrid and 7 hours for Telraam. Hyperparameters including, `number of units` in the hidden layers, `dropout rate`, `learning rate`, and `batch size` are optimized using GridSearch to allow LSTM to capture temporal dependencies effectively. The LSTM model also incorporates previously mentioned time-based features used in XGBoost.

2.5.2 Results and Discussion

In the following section, we present the performance of data filling methods using the proposed no ground truth and ground truth metrics. The models are applied to fill 100 distinct artificial gaps with lengths ranging from 2 to 48 hours. The results are then averaged over these gaps.

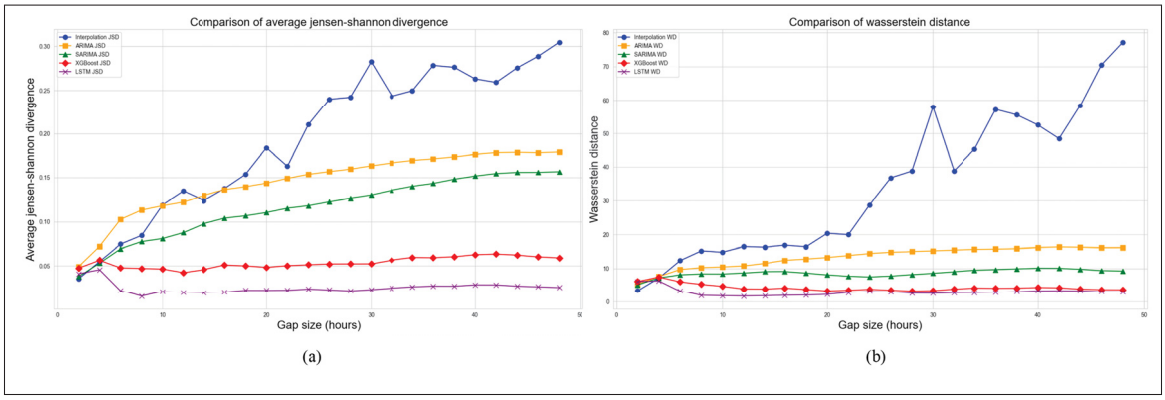

Figure 2.2 (a) Average Jensen-Shannon divergence and (b) Wasserstein distance for different gap sizes considering the Madrid dataset

As shown in Figures 2.2 and 2.3, the analysis of the Madrid dataset highlights a strong alignment between the proposed no ground truth metrics and traditional ground truth based metrics. For instance, both sets of metrics show that the LSTM model outperforms the other methods across all gap sizes. For example, in the case of a 6-hour gap, LSTM shows the lowest values for JSD and WD, as well as the smallest average RMSE and MAE. This similarity between the no ground truth and ground truth metrics suggests that JSD and WD can accurately capture the distributional consistency of the imputed values with pre gap data. Additionally, the XGBoost model also performs well in the Madrid dataset, showing a strong alignment between proposed

and traditional metrics, further reinforcing that JSD and WD can provide insights comparable to RMSE and MAE in evaluating model effectiveness.

In contrast, Interpolation and ARIMA show significantly higher JSD and WD values, especially as gap sizes increase. This trend mirrors the increases in RMSE and MAE for these models, underscoring their limitations in preserving the original data pattern over longer gaps. These observations validate that JSD and WD can highlight deviations in model performance, capturing the same limitations as RMSE and MAE without needing ground truth data. Furthermore, the SARIMA model performs moderately well, with lower deviations than ARIMA and interpolation in both JSD and WD. SARIMA demonstrates a balanced capability to retain seasonal patterns and trends in shorter gaps, although it underperforms compared to LSTM and XGBoost for larger gaps. Overall, the close alignment between the proposed no ground truth and traditional error metrics across all gap sizes and different models confirms that the proposed no ground truth metrics can reflect model performance. The strong performance of LSTM and XGBoost underscore the performance of these models. In contrast, the higher JSD and WD values for Interpolation and ARIMA highlight their limitations, further validating the use of the proposed metrics as reliable performance metrics even in the absence of ground truth.

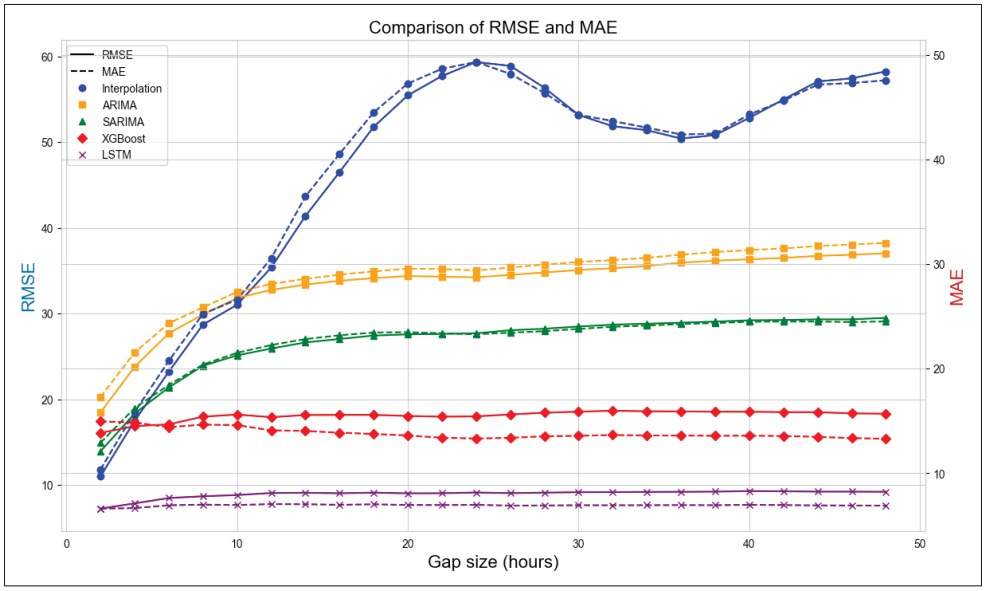

Figure 2.3 Results of MAE and RMSE for Madrid dataset

For the Telraam dataset, we apply the same gap-filling methods and evaluate them using the proposed no ground truth and ground truth based metrics, as illustrated in Figures 2.4 and 2.5. Across all gap sizes, XGBoost consistently shows the lowest JSD and WD values, which indicates that it closely maintains the original data's distribution, even for longer gaps. Furthermore,

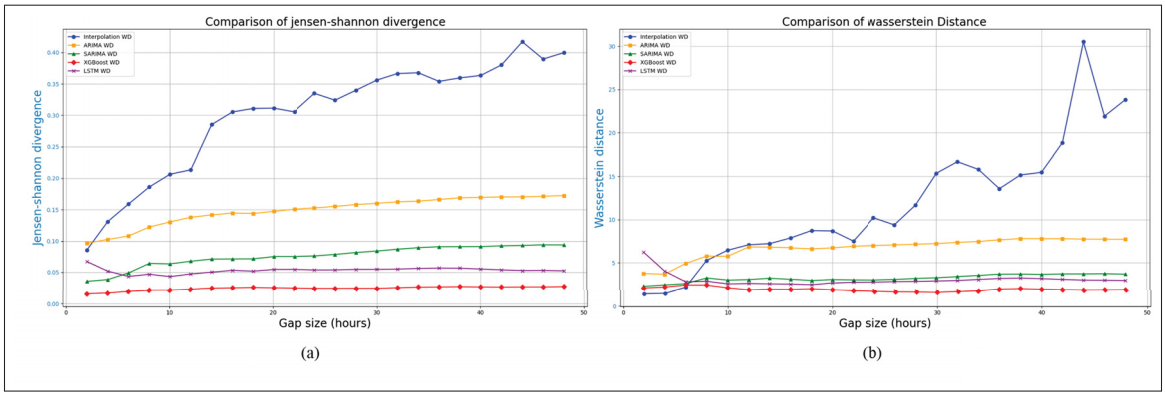

Figure 2.4 (a) Average Jensen-Shannon divergence and (b) Wasserstein distance for different gap sizes considering the Telram dataset

traditional ground truth based metrics also confirm XGBoost's superior performance, particularly for larger gaps, where it sustains low error rates compared to other models. In contrast, the LSTM model also performs well on the Telraam dataset but shows higher JSD and WD values than XGBoost. While LSTM achieves comparable performance for shorter gaps, this divergence may be due to the limited size of the training set, which affects LSTM's ability to generalize over larger gaps and makes it more sensitive to certain underlying patterns in the Telraam dataset. Similarly, as with the Madrid dataset, Interpolation and ARIMA display significant limitations for the Telraam data. Interpolation, in particular, shows notable increases in both proposed no ground truth and ground truth based metrics with growing gap sizes, indicating that it diverges from the original distribution. For instance, in the case of a 40-hour gap, interpolation results in a high JSD and RMSE, which suggests that it fails to preserve the data's distributional characteristics effectively. ARIMA demonstrates a similar trend with increasing JSD and WD for larger gaps, although it maintains slightly better stability than interpolation. Additionally, the SARIMA model shows moderate performance, achieving lower JSD and WD values than ARIMA and interpolation, though it still falls behind XGBoost and LSTM. However, for larger

gaps, SARIMA presents a higher divergence from the original data distribution, as seen in the proposed no ground truth and ground truth based metrics results.

The analysis of both datasets reinforces the capabilities of JSD and WD as reliable no ground truth metrics for evaluating model performance in the absence of real values. The strong performance of XGBoost in the Telraam dataset and LSTM in the Madrid dataset highlights the potential of these models to adapt to dataset-specific characteristics. Meanwhile, the limitations of Interpolation and ARIMA across both datasets, as evidenced by higher JSD and WD values, further validate the performance of the proposed metrics in evaluating model quality. This alignment across models, datasets, and gap sizes underscores the suitability of JSD and WD as reliable performance metrics, enabling effective evaluation even in the absence of ground truth data.

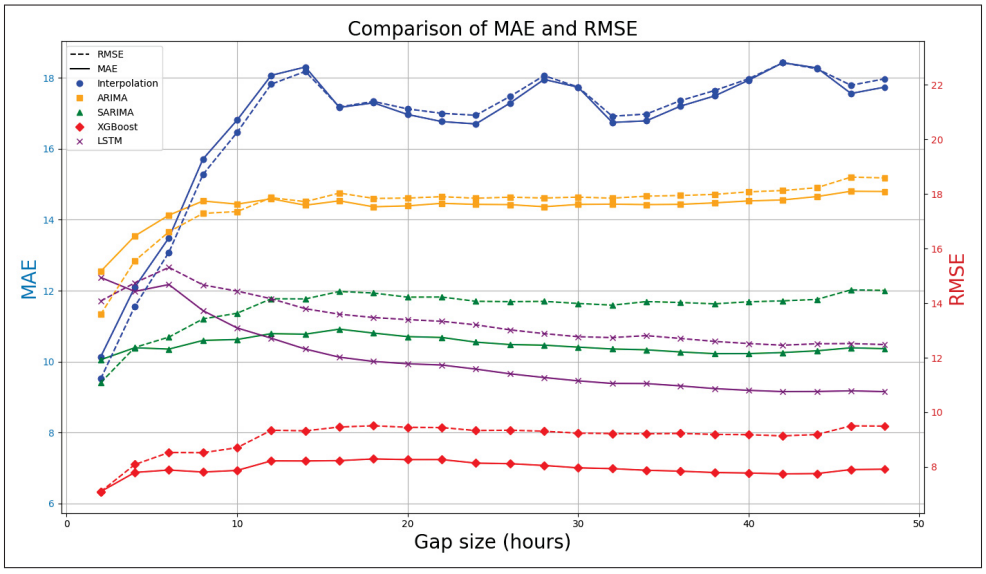

Figure 2.5 Results of MAE and RMSE for Telraam dataset

2.6 Conclusion

This chapter introduced WD and JSD as alternative validation metrics for evaluating data imputation techniques in the absence of ground truth data. By assessing the alignment between

the distributions of imputed and pre-gap data, these metrics offer a reliable method for evaluating imputation quality based on statistical consistency rather than direct comparison with known values. Experimental results with Telraam and Madrid traffic datasets demonstrate that WD and JSD effectively capture imputation quality, suggesting their potential for broader applications in environments where ground truth is unavailable. Future work will explore integrating these metrics with adaptive ML models to further improve robustness and accuracy in complex data settings.

CHAPTER 3

EVALUATION OF TIME SERIES IMPUTATION METHODS WITHOUT GROUND TRUTH

Rania Farjallah[1] , Bassant Selim[1] , Brigitte Jaumard[2] , Samr Ali[3] , Georges Kaddoum[4,5]

[1] Department of Systems Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[2] Department of Computer Science and Software Engineering (CSSE), Concordia University,
1455 De Maisonneuve Blvd. W. Montreal, Québec, Canada H3G 1M8
[3] GAIA, AI Hub Canada, Ericsson,
8275 Trans Canada Route, Saint-Laurent, Quebec, Canada H4S 0B6
[4] Department of Electrical Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[5] Artificial Intelligence & Cyber Systems Research Center, Lebanese American University,
Beirut 03797751, Lebanon

ARTICLE SUBMITTED, IN IEEE Open Journal of the Communications Society, September 2025

3.1 Introduction

In recent years, the proliferation of data-driven solutions has significantly transformed wireless communication systems, particularly with the emergence of 5G and beyond. ML and DL techniques are increasingly integrated into communication networks to optimize performance, improve efficiency, and enable real-time decision-making (see, e.g., Fourati *et al.* (2021)). However, training such models requires access to large-scale, high-resolution traffic datasets. Concurrently, publicly available 5G datasets are extremely scarce, and the existing ones are typically small due to privacy and proprietary restrictions. One proposed approach to address this limitation is using semi-realistic urban datasets that replicate user mobility and behavioral patterns, thus enabling the modeling and generation of synthetic 5G traffic Ziazet *et al.* (2022). However, these urban datasets, frequently, suffer from missing values arising from sensor errors or connectivity issues. Since the corresponding traffic data serve as the foundation for model training and evaluation, effective handling of missing data is critical to ensure modeling accuracy

and reliability—particularly for data-driven solutions in communication systems Pratama *et al.* (2016); Song & Szafir (2019).

Despite being computationally efficient and simple in implementation, traditional methods, such as forward or backward filling and linear interpolation, fail to capture temporal dependencies, seasonality, and trends inherent in complex datasets Saad *et al.* (2020). Accordingly, these methods frequently lead to inaccurate imputations, thereby compromising model accuracy and even overall network performance. While imputation of missing values in time series may seem similar to prediction, the key distinction lies in the availability of ground truth. Whereas prediction models focus on forecasting future values based on observable data, imputation models reconstruct missing values within an existing time series i.e., without access to ground truth. Recently, inspired by recent advances in ML and DL, more sophisticated imputation methods for dealing with missing data in time series have been proposed Che *et al.* (2018). These models are capable of capturing complex dependencies, making them particularly suitable for imputation tasks in dynamic and high-dimensional time series. However, a limitation of these models is that they are typically evaluated and optimized using traditional error metrics Saad *et al.* (2020), which rely on the availability of ground truth. Yet, in real-world applications, this ground truth is often unavailable, making it difficult to evaluate the quality of imputed values. This highlights the need for alternative validation approaches capable of evaluating accuracy and consistency of imputed values with observed data.

In this study, we address key challenges in time series data imputation, with a particular focus on both filling missing values and validating the quality of imputed data without relying on ground truth. Our contributions include a comparative analysis of different imputation techniques using an iterative approach that sequentially fills gaps, as well as providing the definition of three novel validation metrics designed to assess the alignment between imputed and original data. More specifically, key contributions of the present study can be summarized as follows:

- We present a comparative study of imputation models using an iterative filling methodology that progressively imputes missing values while preserving temporal and structural characteristics of the data.

- We propose three novel metrics for evaluating imputed values in the absence of ground truth. By evaluating distributional similarity and structural integrity, these metrics enable robust evaluation of imputed values.
- We demonstrate the effectiveness of the proposed metrics and imputation methods through extensive experiments on three distinct datasets.
- We illustrate the applicability of our approach by filling missing values in a real-world dataset and validating imputation quality using our proposed metrics. Doing so allows us to assess the reliability of imputed data without ground truth, thereby addressing a critical challenge in real-world scenarios where complete datasets are often unavailable.

The remainder of this paper is organized as follows. Section 4.3 reviews related work in the field of time series imputation and its evaluation. In section 4.3.1 describes the imputation models used for filling missing values in time series data. In section 3.4 presents the iterative imputation methodology used to enhance robustness in handling multiple consecutive gaps and provide an overview of the proposed evaluation metrics. In section 3.5 the proposed evaluation metrics are validated for time series imputation, while Section 3.7 showcases the applications of the proposed approach to a realistic dataset. Finally, Section 4.5 concludes the work.

3.2 Related Work

Time series data often exhibit trends such as seasonality, which introduces systematic variations that predictive models should be capable of accounting for to ensure accurate performance. Missing values in these datasets can affect these trends, making imputation essential to maintaining data integrity.

Imputation is the process of replacing missing data with substituted values to restore the completeness of a dataset. An important limitation of traditional imputation methods is that they focus on using ground truth to estimate missing values, but often fail to account for temporal dependencies or seasonality, which considerably limits their ability to reconstruct complex relationships within the data. Overall, imputation techniques can be categorized into single

and MI Donders *et al.* (2006). Single-imputation-based techniques, such as Hot Deck, Cold Deck, and EM-based imputation, replace missing value with a single estimated value. However, these techniques do not necessarily reduce bias, as they may introduce systematic errors into the imputed dataset. Conversely, MI-based techniques are more advantageous as they provide information about the impact of missing data on parameter estimates–namely, mean, variance, regression coefficient, and standard error Donders *et al.* (2006). More advanced methods include regression-based methods, SOM Junninen *et al.* (2004), and KNN, all of which have been shown to be more effective. KNN, in particular, fills missing values by considering the k-closest points to the missing data point Tarsitano & Falcone (2011). Previous studies also used evolutionary learning to handle missing values in time series data. This technique employs the bidirectional forecasting built into automated machine learning (AutoML) to accurately and adaptively fill gaps Sarafanov, Nikitin & Kalyuzhnaya (2022).

Yet, while these approaches effectively address missing data, imputation can be further improved by incorporating seasonality. Seasonality reflects recurring patterns that can guide the reconstruction of missing values. Previously, techniques such as seasonal adjustment with Kalman filters and decomposition-based imputation applied to seasonal datasets Moritz *et al.* (2015). In addition, SARIMA models have also been used to fill gaps in seasonal time series, though their performance is limited when handling consecutive missing values Sutiene *et al.* (2011). Overall, deep learning models like MLP Wijesekara & Liyanage (2023), LSTM networks, and hybrid neural networks Bandara *et al.* (2021) are particularly effective in capturing non-linear dependencies and long-term trends, excelling in reconstructing missing values in high-dimensional time series data. These methods are particularly effective in reconstructing missing values in high-dimensional time series data. Moreover, several previous studies used pattern-based methods that deal with missing data, including the Top-k CaseMatching (TKCM) algorithm Wellenzohn *et al.* (2017).

However, despite the progress afforded by imputation techniques, evaluating their performance remains a major challenge. Most relevant studies artificially create gaps in comprehensive datasets and evaluate imputation performance using traditional metrics, such as RMSE and

MAE. These metrics, which largely rely on ground truth, are unsuitable for real-world scenarios where actual values are unavailable, highlighting the need for new metrics to evaluate imputation quality without relying on ground truth. To the best of our knowledge, no prior studies on time series imputation have employed no-ground-truth evaluation metrics to systematically assess imputation quality. To this end, in the present study, we introduce three metrics to evaluate the alignment between imputed and observed data distributions. Our approach enables validating imputed time series by examining their consistency with inherent temporal and structural patterns in the absence of ground truth. This capability is essential for real-world applications, such as 5G traffic management, where data reliability is critical.

3.3 Gap Filling for Time Series Data

In this section, we provide an overview of the methods used in this study to fill gaps in time series data. The methods were selected based on their extensive use in time series data, their ability to capture temporal dependencies, and their practical relevance in handling missing data across different domains. We include traditional statistical models, such as ARIMA and SARIMA, a machine learning technique (namely, XGBoost), and deep learning models, such as LSTM and TimeGPT–a transformer-based approaches. While this study focuses on five representative methods, this selection aims to span multiple methodological families, including statistical, machine learning, deep learning, and transformer-based approach. These methods have been widely used in literature and real-world applications due to their interpretability, availability in open-source toolkits, and suitability for temporal modeling. Our goal in this paper is to establish the utility of our evaluation methodology on a tractable yet diverse subset of techniques, without claiming to exhaustively cover all existing approaches.

3.3.1 ARIMA-based imputation

ARIMA is commonly applied for time series forecasting, capturing trends and autocorrelations in data Brockwell & Davis (2018). ARIMA can also be used to estimate missing values by modeling both AR and MA terms Chen, Lin & Zeng (2022). The AR(p) model predicts the

value y_t at time t based on the p most recent observations:

$$\text{AR}(p): \quad y_t = \mu + \sum_{i=1}^{p} \phi_i y_{t-i} + \varepsilon_t, \tag{3.1}$$

where μ is the mean of the series, ϕ_i and y_{t-i} are the AR coefficients and lagged values, respectively, for $i = 1..p$, and ε_t represents random noise.

By contrast, the MA(q) model predicts y_t based on the q most recent forecast errors:

$$\text{MA}(q): \quad y_t = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t, \tag{3.2}$$

where θ_i is the moving average coefficient and ε_{t-i} the past error for $i = 1..p$.

The selection of the optimized ARIMA model relies on determining the values of p and q using a criterion such as the akaike information criterion (AIC) and and bayesian information criterion (BIC) Chen, Niu, Liu, Jiang & Ma (2018). The performance of ARIMA can be improved by adjusting parameters such as differencing order, as well as by refining the selection of AR and MA terms to avoid overfitting or underfitting.

3.3.2 SARIMA-Based Imputation

SARIMA extends ARIMA by incorporating seasonal patterns Hanbanchong & Piromsopa (2012), making it suitable for imputing missing values in time series data with periodic patterns. These seasonal terms include seasonal autoregressive (K), seasonal differencing (D), and seasonal moving average (M) components. The seasonal period (s) defines the length of the repeating cycle (e.g., $s = 7$ for weekly seasonality). The model is mathematically represented as shown in Eq. (3) Chen *et al.* (2018).

$$\phi_k(B)\Phi_K(B^s)R_t = \theta_m(B)\Theta_M(B^s)\omega_t, \tag{3.3}$$

where $R_t = (1-B)^{\delta}(1-B^s)^D r_t$ is the differenced series, and B is the backshift operator such that $Br_t = r_{t-1}$. The terms k, δ, m represent the non-seasonal autoregressive, differencing, and moving average orders, respectively, while K, D, M represent their seasonal equivalents. The functions $\phi_k(B)$ and $\Phi_K(B^s)$ denote the non-seasonal and seasonal autoregressive components, respectively, while $\theta_m(B)$ and $\Theta_M(B^s)$ represent the non-seasonal and seasonal moving average components. The error term ω_t represents random noise. By combining non-seasonal and seasonal terms, SARIMA effectively fills missing values in time series with recurring patterns. The effectiveness of SARIMA depends on selecting the appropriate seasonal parameters, including the seasonal period length and seasonal differencing order, to improve its ability to model periodic missing values.

3.3.3 XGBoost-Based Imputation

In this study, we applied XGBoost Chen & Guestrin (2016), an advanced boosting algorithm combining multiple weak learners to iteratively improve prediction accuracy, to fill missing values in time series data by capturing nonlinear patterns and interactions through gradient-boosted decision trees. It operates through a process called boosting, where decision trees are built sequentially. Each new tree is trained to minimize the errors made by the previously constructed trees. This iterative process enables the algorithm to refine its predictions at each step, ultimately building a strong predictive model capable of handling complex data patterns. XGBoost uses the classification and regression tree (CART) model Trendowicz & Jeffery (2014). The objective function of XGBoost given by Eq. (4) Cherif & Kortebi (2019).

$$\text{OBJ}^{\text{XGBOOST}} = \sum_{i=1}^{N} \ell(v_i, \hat{v}_i) + \sum_{j=1}^{T} \Omega(g_j), \tag{3.4}$$

where N is the total number of data points, $\ell(v_i, \hat{v}_i)$ is the loss function measuring the difference between true value v_i and predicted value $\hat{v}_i$, T is the total number of trees, and $\Omega(g_j)$ is the regularization term that controls complexity of each tree g_j, thus helping to prevent overfitting Cherif & Kortebi (2019). This structure allows XGBoost to capture complex patterns within

the data. Unlike sequential models like LSTMs, XGBoost relies on feature engineering such as lags, trends, and seasonality to uncover complex patterns, making it well-suited for filling missing values in time series data. Optimizing XGBoost involves selecting an appropriate number of estimators, adjusting the learning rate, and tuning tree depth to balance complexity and generalization.

3.3.4 LSTM-Based Imputation

LSTM is a type of RNN architecture designed to capture long-term dependencies in sequential data. LSTM models are used to impute missing time series values by learning long-term temporal dependencies and nonlinear patterns. Unlike standard RNNs, LSTM addresses the vanishing gradient problem through the use of a memory cell, which stores important information over time. The structure of an LSTM unit includes the following three gates: the input gate, forget gate, and output gate, all of which are essential for updating its state and storing the information Hochreiter & Schmidhuber (1997). The input gate (i_t) is responsible for selecting which data should be stored for future states, whereas the forget gate (f_t) decides which data from the current state should be discarded. Finally, the output gate (o_t) controls what data from the current state are sent to the output. In what follows, we describe the operations of LSTM Zhang, Chen, Wang & Liu (2019) through forget gate f_t that identifies the data in the current state that must be deleted. The forget gate f_t is given by Eq. (5) Kurri, Raja & Prakasam (2021).

$$f_t = \sigma(W_f[h_{t-1}, z_t] + b_f), \tag{3.5}$$

where z_t is the input vector at the current time step t, h_{t-1} is the hidden state from the previous time step, W_f is the weight matrix, and b_f is the bias term.

Next, the input gate i_t is responsible for selecting which data should be stored for future states and is mathematically represented by Eq. (6) Kurri *et al.* (2021).

$$i_t = \sigma(W_i[h_{t-1}, z_t] + b_i), \tag{3.6}$$

where W_i is the weight matrix and b_i is the bias term. A candidate cell state $\tilde{c}_t$ is then computed, representing new information that could be stored:

$$\tilde{c}_t = \tanh(W_c[h_{t-1}, z_t] + b_c), \tag{3.7}$$

where W_c is the weight matrix and b_c is the bias term.

The new cell state c_t is updated based on the stored information from the previous state and the new candidate value:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \tag{3.8}$$

where $\odot$ denotes element-wise multiplication to ensure that only relevant information is combined and stored in the memory cell during updates, while c_{t-1} represents the previous cell state.

The output gate o_t then determines what part of the current cell state is sent to the output Kurri *et al.* (2021):

$$o_t = \sigma(W_o[h_{t-1}, z_t] + b_o), \tag{3.9}$$

where W_o is the weight matrix and b_o is the bias term.

Finally, the hidden state h_t is computed as follows:

$$h_t = o_t \odot \tanh(c_t), \tag{3.10}$$

where the activation function σ is the sigmoid function, mapping values to the range $[0, 1]$, while tanh is the hyperbolic tangent function, mapping values to $[-1, 1]$. Optimizing LSTM performance involves adjusting the number of hidden layers to balance model complexity and prevent overfitting. However, while increasing the depth of the network enables capturing more complex temporal dependencies, it may lead to longer training times and higher risk of overfitting. This risk can be mitigated by regularization techniques such as dropout that randomly deactivate neurons during training, thus improving generalization. In addition, since smaller batches may introduce more variance in weight updates, while larger batches provide

smoother gradients at the cost of increased memory usage, selecting an appropriate batch size is crucial for stabilizing training dynamics and ensuring efficient convergence.

3.3.5 TimeGPT-Based Imputation

TimeGPT is used to impute missing values in time series data by modeling both short- and long-range dependencies. Its design builds on a transformer-based architecture inspired by Large Language Models (LLMs) Garza, Challu & Mergenthaler-Canseco (2024). The architecture of TimeGPT incorporates Positional Encoding (PE), multi-head attention, and Convolutional Neural Networks (CNNs). These elements, along with residual connections and layer normalization, enhance the model's stability during training and accelerate convergence. More specifically, PE encodes the position of each input feature using sine-cosine positional coding, thus enabling the model to recognize the sequential structure within the time series data Liu *et al.* (2023):

$$\mathrm{PE}_{\mathrm{pos},2i} = \sin\left(\frac{\mathrm{pos}}{10000^{\frac{2i}{d_{\mathrm{model}}}}}\right), \tag{3.11}$$

$$\mathrm{PE}_{\mathrm{pos},2i+1} = \cos\left(\frac{\mathrm{pos}}{10000^{\frac{2i}{d_{\mathrm{model}}}}}\right), \tag{3.12}$$

where pos denotes the position index, i denotes the dimension index, and d_{model} is the feature length. With an increase of the position pos, the value of the PE changes, enabling the model to distinguish features at different positions and better understand the sequential structure of the input data. Furthermore, the multi-head attention mechanism plays a central role in TimeGPT's ability to model both short-term and long-term dependencies. By computing attention scores across multiple heads in parallel as shown in Eq. (13), the model can simultaneously focus on different parts of the input sequence Yu *et al.* (2024):

$$\mathrm{MultiHead}(Q, K, V) = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h)W^O, \tag{3.13}$$

where each attention head is defined as follows:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V), \tag{3.14}$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{3.15}$$

with Q, K, and V representing the query, key, and value matrices, respectively, while W^Q, W^K, W^V denote their respective learned weights. Additionally, the CNN layers in TimeGPT capture the latent feature to each position within the time series data. By applying convolution and pooling operations, the model extracts meaningful features yielding the following:

$$X_{\text{conv,out}} = \sigma(W_{\text{conv}} \times X_{\text{conv,in}} + B_{\text{conv}}), \tag{3.16}$$

where $X_{\text{conv,out}}$ is the output, $X_{\text{conv,in}}$ is the input to the convolution layers, σ represents the activation function, W_{conv} denotes the convolution layer's weights, and, finally, B_{conv} represents the biases.

To further improve training stability and mitigate the vanishing gradient problem, TimeGPT incorporates residual connections and layer normalization. **ln!** (**ln!**) stabilizes the output range of each sub-layer by performing as shown in Eq. (17) Liao *et al.* (2025):

$$X_{\text{LN}} = \frac{X_{\text{SL}} - \mu}{\sigma + \varepsilon} \cdot \gamma + \beta, \tag{3.17}$$

where X_{SL} represents the sub-layer output, μ and σ denote the mean and standard deviation of X_{SL}, γ and β are learnable parameters, and ε is a small constant to avoid division by zero errors. Meanwhile, the residual connections prevent gradient degradation by adding the input of a sub-layer directly to its output as shown in Eq. (18):

$$X_{\text{F,out}} = F(X_{\text{F,in}}) + X_{\text{F,in}}, \tag{3.18}$$

where $X_{\text{F,out}}$ and $X_{\text{F,in}}$ denote the output and input of sub-layer F, respectively.

TimeGPT is pre-trained on a variety of datasets from various domains, including finance, weather, and transportation, which enables it to generalize well across different temporal granularities. By adjusting hyperparameters such as the number of fine-tuning steps and the loss function, fine-tuning adjusts the pre-trained model to specific tasks, thereby preventing overfitting and allowing it to process sequences of varying lengths and frequencies. The fine-tuning process is discussed in further detail in Section. 4.4.3.

3.4 Proposed Missing Data Imputation Approach

In this section, we introduce an iterative imputation approach aimed at improving robustness and effectively handle multiple consecutive gaps. We also provide an overview of the evaluation metrics to validate imputed data in the absence of ground truth.

3.4.1 Iterative Filling Approach

In real-world applications, time series datasets frequently suffer from consecutive gaps. These gaps limit the availability of sufficient data for training ML models, as the absence of surrounding data points prevents models from learning accurate patterns. For instance, in telecommunication and 5G networks, missing values frequently occur due to signal loss, hardware failures, or network disruptions. Such consecutive gaps are particularly challenging in high-frequency data streams like those in 5G applications, where precise predictions are critical. To address this issue, iterative imputation sequentially fills missing data, thus enabling models to progressively use imputed values to reconstruct the dataset and support effective training.

The iterative approach focuses on sequentially filling each gap to incrementally improve data availability (see in Fig. 3.1). For each gap, imputation methods are applied to estimate the missing values for the first gap based on the available surrounding data. Once the first gap is filled, the imputed values are incorporated into the dataset and then treated as if these values were part of the original data. This filled dataset is then used to predict and fill the next gap. By iterating this process for all gaps, the dataset is gradually reconstructed. After all gaps are filled,

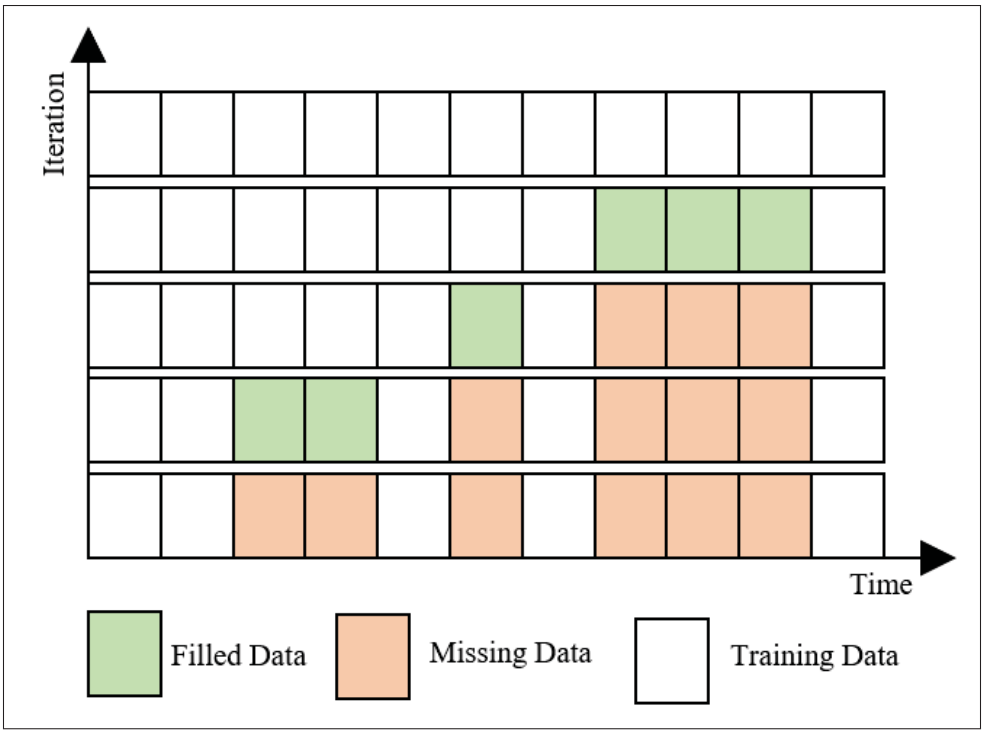

Figure 3.1 Iterative approach

the performance of each method is evaluated using appropriate validation metrics, ensuring the accuracy and reliability of the imputed values.

3.4.2 Evaluation Metrics

Evaluating imputed time series data in the absence of ground truth requires a careful selection of metrics that can accurately assess data integrity. Since both data imputation and synthetic data generation aim to create realistic data that maintain statistical properties of the original dataset, we use the following three metrics commonly applied in synthetic data evaluation: JSD, WD, and Discriminative score (DS). These metrics are chosen for their relevance in accurately assessing how closely the imputed data match the original dataset's statistical properties. Table 3.1 summarizes the definition and relevance of each metric within our context.

3.4.2.1 Wasserstein Distance (WD)

The WD, also known as the Earth Mover's Distance Yossi Rubner & Guibas (2000) or the Monge–Rubinstein metric Cédric Villani (2009), quantifies the distance between two probability distributions based on the theory of optimal transport. Let $\Omega \subset \mathbb{R}^d$ be a bounded domain, while $\mathcal{P}(\Omega)$ denotes the set of probability measures supported on Ω. Then, for any $P, Q \in \mathcal{P}(\Omega)$, the p-Wasserstein distance is defined by solving the optimal transport problem with the cost function $c(x, y) = \|x - y\|^p$ Soheil Kolouri,Se Rim Park & Rohde (2017):

$$W_p(P, Q) = \left(\inf_{\gamma \in \Gamma(P,Q)} \int_{\Omega \times \Omega} \|x - y\|^p \, d\gamma(x, y) \right)^{1/p}, \tag{3.19}$$

where $\Gamma(P, Q)$ is the set of all couplings (joint distributions) with marginals P and Q. Metric W_p defines a true distance on $\mathcal{P}(\Omega)$ for any $p \geq 1$, while the corresponding metric space $(\mathcal{P}(\Omega), W_p)$ is referred to as the p-Wasserstein space.

In this study, we use the case $p = 1$, where the metric reduces to the classic Earth Mover's Distance. For one-dimensional distributions (e.g., time series), the W_1 metric admits a closed-form solution based on the inverse cumulative distribution functions of P and Q, making it computationally efficient. This metric is particularly well-suited for the evaluation of imputed data in the absence of ground truth, as it directly compares the distribution of observed pre-gap values with that of the imputed values. A smaller WD indicates that the imputed data closely align with the statistical characteristics of the original data, preserving their inherent patterns and distributions. This alignment demonstrates effectiveness of the imputation method in maintaining the data's original characteristics.

3.4.2.2 Jensen-Shannon Divergence

The JSD, is a statistical measure for quantifying the similarity between two probability distributions Stenger *et al.* (2024), based on the KL divergence, which measures the divergence of one probability distribution, P, from another, Q. Unlike the asymmetric KL divergence, which

can produce infinite values when probabilities in one distribution are zero, the JSD is symmetric and always bounded. These properties make the JSD more robust for comparing distributions, particularly in time series imputation. The symmetry of the JSD ensures that the comparison between two distributions, P and Q, remains unbiased, as the order of the distributions does not affect the results. Moreover, in cases where P or Q contain zero-probability points, JSD avoids instability, which makes it a reliable choice for evaluating the similarity between observed (pre-gap) data and imputed values. The JSD between distributions P and Q is defined as follows:

$$JS(P\|Q) = \frac{1}{2}D_{KL}(P\|M) + \frac{1}{2}D_{KL}(Q\|M), \tag{3.20}$$

where $M = \frac{1}{2}(P+Q)$ represents the average distribution, and D_{KL} is the KL divergence. A JSD value close to 0 indicates a high similarity between distributions, demonstrating that the imputed data preserves the statistical characteristics of the original data.

3.4.2.3 Discriminative Score (DS)

The DS evaluates the similarity between real and synthetic data by measuring the performance of a binary classifier trained to distinguish between them. Every sample from the real dataset is labeled as real while each sample from the synthetic dataset is labeled as synthetic. These labeled samples are then combined to form a two class dataset, which is then split into training and testing sets Stenger *et al.* (2024).

A simple LSTM network is trained on this two class dataset, learning to classify samples as either real or synthetic. The accuracy of the LSTM model on the test set serves as the basis for calculating the discriminative score, defined as the model's accuracy minus 0.5, yielding the following:

$$DS = \text{Accuracy} - 0.5. \tag{3.21}$$

This score ranges from zero (indicating optimal similarity, where the classifier cannot distinguish real from synthetic data) to 0.5 (indicating complete distinguishability). A lower discriminative score suggests that the imputed data closely resemble the real data, thus implying high quality.

Table 3.1 Proposed evaluation metrics for time series imputation without ground truth

Metric	Theoretical Basis	Evaluated Feature	Relevance Without Ground Truth
Wasserstein Distance (WD)	Defined via the optimal transport problem with cost function $c(x, y) = \|\|x - y\|\|^p$; for $p = 1$, it reduces to the Earth Mover's Distance Cédric Villani (2009); Soheil Kolouri & Rohde (2017)	Global distributional alignment	Evaluates how much the imputed values deviate from the distribution of original values; useful for the detection of global distortions
Jensen-Shannon Divergence (JSD)	Evaluates distributional similarity between two probability distributions using a symmetric and bounded divergence derived from KL distance; uses the average of both distributions as a reference Stenger *et al.* (2024)	Entropy-based distribution similarity	Robust to zero probabilities; quantifies how well imputed data retains variability
Discriminative Score (DS)	Based on the accuracy of a binary classifier trained to distinguish real from imputed data Stenger *et al.* (2024)	Structural realism	Measures how indistinguishable the imputed values are from real data using a learned classifier

3.5 Validation of the Proposed Evaluation Metrics

In this section, we validate the proposed metrics to evaluate imputation quality in time series data. The first describe the methodology, which involves introducing artificial gaps into two complete datasets to simulate missing data. Traditional metrics are then compared against the newly proposed metrics, with a particular focus on their ability to assess imputation quality without ground truth. Next, we show that these metrics demonstrate a strong alignment with the traditional ones, thus confirming the reliability of the latter.

3.5.1 Datasets

To validate the proposed methodology, we considered two datasets. The first dataset is obtained from Telraam [1], a citizen-driven solution designed to collect multimodal traffic data through an easy-to-use traffic counter. While various datasets from Telraam are available, including some on Kaggle, we specifically selected one street segment in Brussels for its data quality and completeness. The second dataset was the Madrid dataset obtained from the open data portal of the city of Madrid [2] . Both selected datasets are comprehensive and include seasonal time series, counting the number of vehicles in different transport modes. They are suitable for validating the proposed metrics against traditional ground-truth-based evaluation metrics.

Brussels Telraam Dataset [1]. This dataset tracks the movement of pedestrians, cyclists, cars, and heavy vehicles every hour during the day period at a single street intersection. We selected Brussels for its diverse traffic patterns, as well as for its low rate of missing rate in this particular intersection (below 1% over the entire recorded period). The data were collected with cameras installed as part of the Telraam device [1] recording data from October 2021 to January 2024.

Madrid Dataset[2]. This dataset provides historical and real-time traffic data in the city of Madrid with a frequency of 15 min. The data were recorded from July 2013 to October 2024 with a 0% missing data. This dataset captures the intensity of car traffic across different segments.

3.5.2 Validation process

To ensure the reliability of the proposed evaluation metrics, a comprehensive validation process is proposed. This process evaluates the capability of the metrics to evaluate performance of imputation methods in the absence of ground truth data, thereby establishing their suitability as alternatives to traditional validation metrics.

---

[1] Traffic Count Data : https://telraam.net/

[2] City of Madrid, Open Data Portal : https://datos.madrid.es/portal/site/egob

The validation process starts with introducing artificial gaps at random positions within the datasets, simulating real-world missing data scenarios. This approach enables a comparison between the gap-filled values and the true held-out observations, as well as the analysis and comparison of performance of different gap-filling methods using both ground truth and no ground truth metrics. The gaps are filled using imputation methods detailed in Section 4.3.1.

The performance of each method is evaluated using traditional metrics, namely the RMSE and MAE, which rely on the original ground truth (complete) data. These metrics involve a direct comparison of the gap-filled values to the true observations. By contrast, for the proposed metrics, we use pre-gap values as a reference to measure the alignment of the imputed data.

The reason behind this decision is that we do not have ground truth data in practice and, as shown in Fig. 3.2, when considering the same gap size, the distribution of pre-gap values closely matches the one of the true values. This similarity in distribution ensures that the pre-gap segment serves as an appropriate proxy for evaluating how well the gap-filled data align with the original data distribution, without needing access to true values. This approach enables an evaluation whether the statistical metrics can effectively measure the alignment between the gap-filled data align and the general data distribution, without requiring the ground truth.

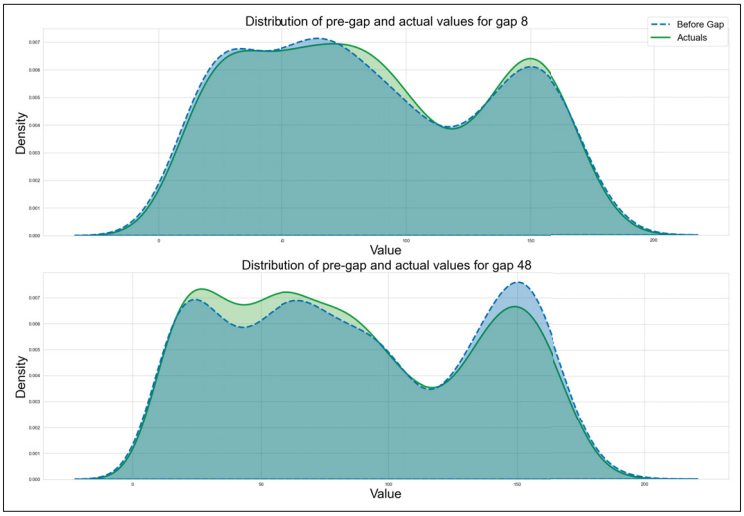

Figure 3.2 Comparison between the distribution of the missing data and pre-gap data for short and large gaps

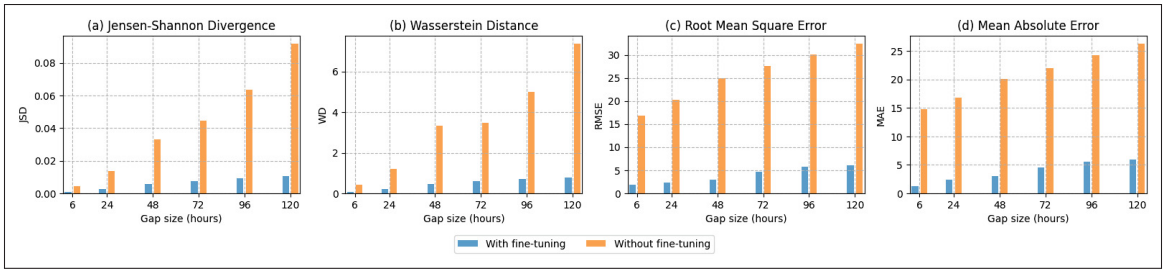

Figure 3.3 TimeGPT results with and without fine-tuning for Madrid dataset

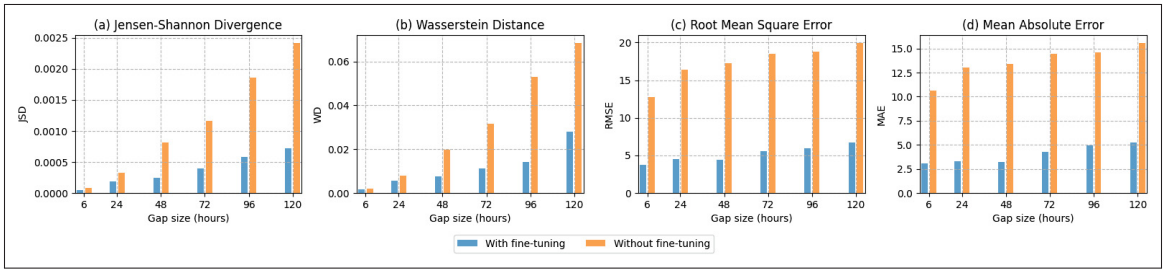

Figure 3.4 TimeGPT results with and without fine-tuning for Brussels dataset

After filling the artificially created gaps using each method, the results of the proposed metrics are compared with those of traditional, ground truth-based metrics. This comparison validates the ability of the proposed metrics to reliably evaluate imputation quality. The strong alignment between the proposed and traditional metrics confirms their effectiveness as metrics for the evaluation of imputation methods in scenarios where ground truth data are unavailable.

Once validated using complete datasets, these metrics can also be applied to real-world incomplete datasets containing naturally occurring gaps. The details of this application and the numerical

Table 3.2 Evaluation of missing data imputation on Madrid dataset across different gap lengths

Gap size	**Gap Size = 6**				**Gap Size = 24**				**Gap Size = 48**			
Metrics	**RMSE**	**MAE**	**JSD**	**WD**	**RMSE**	**MAE**	**JSD**	**WD**	**RMSE**	**MAE**	**JSD**	**WD**
ARIMA	28.1905	25.0250	0.0094	0.8552	33.6463	28.6690	0.0364	3.6348	36.4362	31.2825	0.0718	7.3482
SARIMA	25.1812	22.5349	0.0089	0.7290	29.0241	24.4845	0.0255	2.2389	30.7317	25.5677	0.0658	4.3320
XGBoost	13.8550	11.6012	0.0032	0.2380	15.1938	11.7691	0.0094	0.7073	16.0360	11.9523	0.0181	1.3160
LSTM	8.9863	7.3817	0.0024	0.1882	9.4338	7.2649	0.0066	0.4935	9.8739	7.4922	0.0101	1.0427
TimeGPT	**3.8951**	**3.2083**	**0.0008**	**0.0630**	**4.3536**	**3.3662**	**0.0027**	**0.2265**	**4.5682**	**3.4525**	**0.0058**	**0.4545**

results is presented in Section 3.7. By demonstrating reliability of the proposed metrics and their applicability to real-world data, the proposed methodology establishes a robust approach to address missing data in time series without requiring access to ground truth values.

3.6 Numerical Results

In this section, we demonstrate that our proposed metrics can effectively evaluate the quality of imputed data without ground truth. In order to ensure ensure robust performance, we first optimize the hyperparameters of each model, which plays a crucial role in enhancing accuracy and generalization across diverse datasets. The optimization process varies depending on the model type. For statistical models such as ARIMA and SARIMA, parameter selection is automated using the `auto_arima` function from the `pmdarima` library. This function explores various combinations of (p, d, q) for ARIMA and (P, D, Q, s) for SARIMA, where s represents the seasonal period. The model with the lowest AIC is selected, balancing model accuracy and complexity to avoid overfitting.

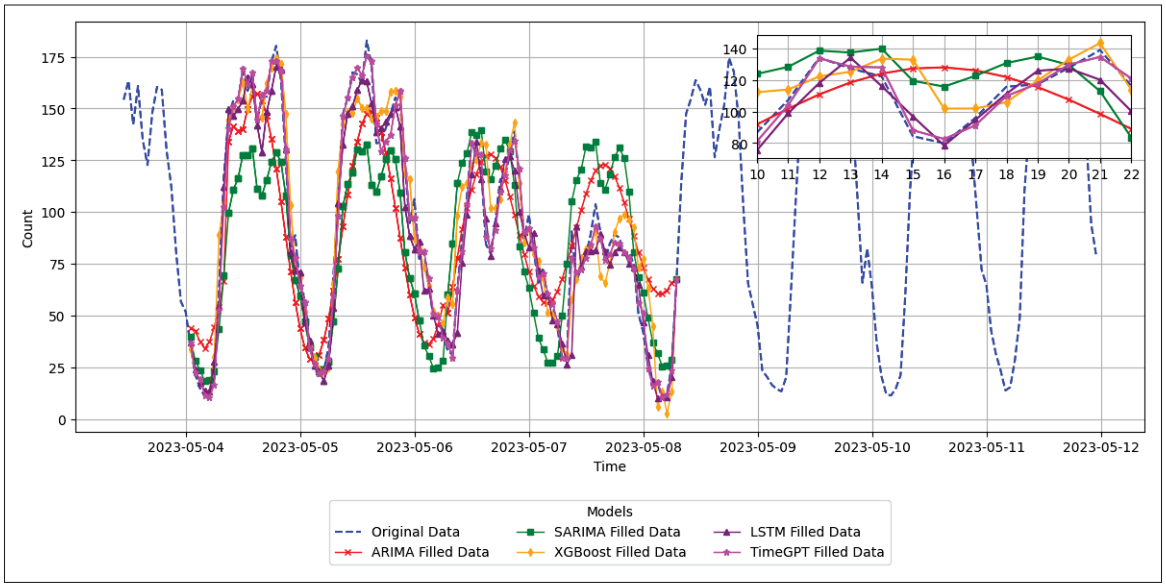

Figure 3.5 Data filling on Madrid dataset

For ML models like XGBoost, a systematic hyperparameter optimization is conducted using `GridSearchCV`, an exhaustive search approach that evaluates all possible combinations of predefined hyperparameters. The optimized parameters include the learning rate, which is tested with values of 0.1, 0.01, and 0.001; the maximum depth of the trees, with values of 3, 5, 7, and 9; the number of estimators, which ranges from 50 to 200 in increments of 50; and, finally, the subsample ratio, with values of 0.5, 0.7, and 0.9. This approach ensures that the most effective parameter settings are identified for accurate imputation of missing data.

Similarly, for the LSTM model, hyperparameter tuning is performed using `GridSearchCV` to systematically explore all possible combinations of key hyperparameters. The activation function is tested with `relu` and `tanh`, while the optimizer is set to `adam`. The learning rate is optimized across the values of 0.1, 0.01, 0.001, and 0.0001, while the loss function is set to mean-squared-error. The number of neurons per layer varies among 32, 64, and 128, while the dropout rate is tested with the values of 0.1, 0.2, and 0.3. Finally, the batch size is optimized with the values of 16, 32, and 64. To ensure consistent performance across training and validation, each configuration is evaluated using cross-validation to ensure consistent performance across training and validation sets, and the best-performing configuration is selected for the final model.

Table 3.3 Evaluation of missing data imputation on Brussels dataset across different gap lengths

Gap size	**Gap Size = 6**				**Gap Size = 24**				**Gap Size = 48**			
Metrics	**RMSE**	**MAE**	**JSD**	**WD**	**RMSE**	**MAE**	**JSD**	**WD**	**RMSE**	**MAE**	**JSD**	**WD**
ARIMA	16.6071	14.7049	0.0215	0.9653	18.8103	15.7392	0.0863	4.0319	19.1740	15.9330	0.1481	6.4287
SARIMA	14.2548	12.1787	0.0151	0.7038	15.9922	12.3009	0.0515	2.6972	16.9085	12.7198	0.0903	4.5555
XGBoost	5.9712	4.9187	0.0052	0.1246	6.5084	4.9819	0.0119	0.3430	6.6382	5.0505	0.0177	0.5792
LSTM	11.1637	9.2738	0.0092	0.4545	12.2004	9.3780	0.0320	1.7115	12.3971	9.3911	0.0547	2.8116
TimeGPT	**3.7529**	**3.0647**	**0.0004**	**0.0147**	**4.4210**	**3.2233**	**0.0018**	**0.0552**	**4.4631**	**3.2616**	**0.0024**	**0.0762**

The TimeGPT model is also fine-tuned to improve its performance in predicting missing values for various gap sizes. This fine-tuning process involves setting `finetune_steps`, which controls the number of gradient descent updates, and `finetune_depth`, which determines how many layers of the model are updated. The MAE is selected as the loss function to focus

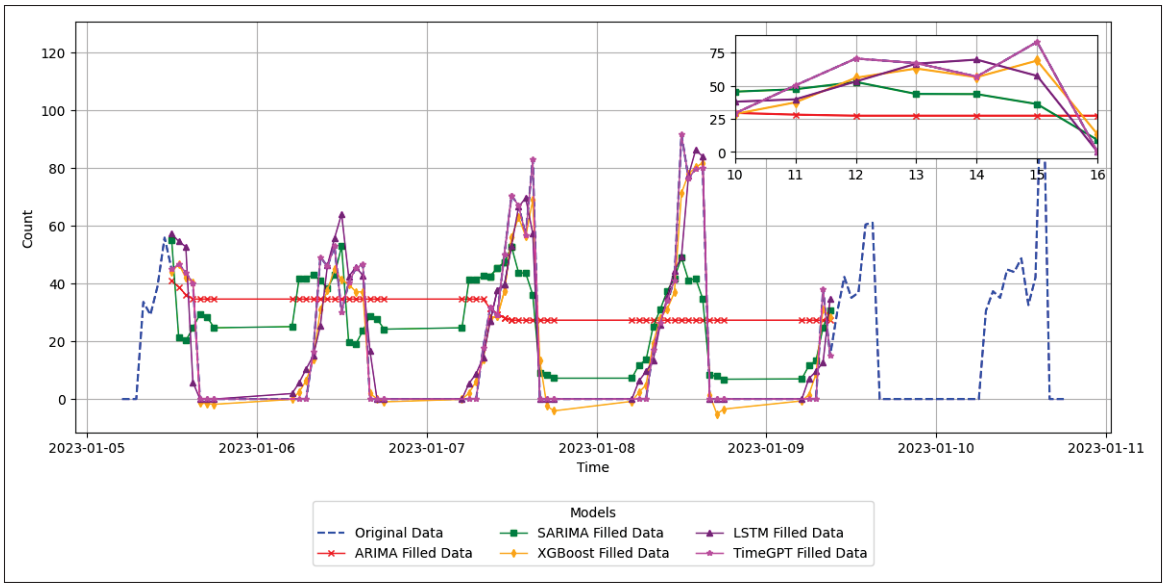

Figure 3.6 Data filling on Brussels dataset

on minimizing absolute differences between predictions and actual values. In addition, the `timegpt-1-long-horizon` variant of the model is used, designed specifically for long-range forecasts. We optimize over the following parameters and their ranges during fine-tuning:

- `finetune_steps` $\in \{30, 60, 100, 150\}$
- `finetune_depth` $\in \{1, 2, 3, 4, 5\}$
- `model` $\in$ {timegpt-1, timegpt-1-long-horizon}
- `finetune_loss` $\in$ {mae, mse, rmse, mape, smape}
- `Exogenous variables`

The impact of fine-tuning adjustments on the TimeGPT model for both datasets is illustrated in Figures 3.3 and 3.4. In the Madrid dataset, the performance before fine-tuning (orange bars) shows a noticeable increase in errors as the gap size grows from 6 to 120 hours. The JSD and WD values are particularly high for larger gaps, implying that the imputed values substantially deviate from the original data distribution. Similarly, the RMSE and MAE values also increase, indicating a lower accuracy. However, after applying the fine-tuning step (blue bars) a significant improvement across all metrics is observed. The JSD and WD scores remain consistently low, demonstrating that the fine-tuned model maintains the statistical structure of the data even for

large gaps. The RMSE and MAE values also substantially decrease, indicating more precise predictions that closely align with the true values.

Similarly, in the Brussels dataset, the model struggles with larger gaps before fine-tuning, as indicated by high JSD, WD, RMSE, and MAE values. The imputed data show a noticeable drift from the original patterns, particularly for the gaps beyond 48 hours. After fine-tuning, the model's performance significantly improves, with much lower values across all metrics. The JSD and WD scores demonstrate that the imputed values remain closely aligned with the pre-gap distribution, highlighting the model's ability to maintain both short-term fluctuations and long-term seasonal patterns. The RMSE and MAE values also demonstrate substantial reductions, indicating better accuracy. Therefore, fine-tuning significantly improves TimeGPT's imputation performance. By adjusting the model's layers and parameters, it becomes more effective at handling longer gaps, resulting in imputed values that closely reflect the original data distribution.

After optimizing the models, we now analyze their imputation performance. Tables 3.2 and 3.3 show the results on the performance of different imputation methods for different gap sizes in the Madrid and Brussels datasets, including a comparison between traditional error-based metrics and the proposed metrics.

Table 3.4 Discriminative score by method

Method	**Madrid dataset**	**Telraam dataset**
ARIMA	0.1060	0.1453
SARIMA	0.0453	0.1085
Xgboost	0.0242	0.0755
LSTM	0.0053	0.0977
TimeGPT	**0.00098**	**0.0033**

For the Madrid dataset, the results reveal that TimeGPT outperforms other models for all gap sizes. For small gaps, as indicated by the lowest RMSE and MAE values among all methods, TimeGPT's imputed values remain closest to the original data. This is reflected in its low JSD and WD scores, thus confirming that the imputed values preserve the same distribution

as the pre-gap data. Other models–namely XGBoost and LSTM– also perform well for small gaps, but show slightly higher error and divergence values as compared to TimeGPT. With an increase of the gap size to 24 and 48 hours, TimeGPT continues to provide accurate imputations, maintaining low JSD and WD scores. By contrast, ARIMA and SARIMA show significant increases in RMSE, MAE, JSD, and WD, highlighting a decline in performance. XGBoost also reveals inconsistencies for longer gaps, as seen by higher scores as compared to TimeGPT. The discriminative score further confirms TimeGPT's superior performance, as shown by low values, meaning the imputed data are almost indistinguishable from the original. Taken together, these results confirm that JSD, WD, and DS are reliable alternatives to traditional metrics, providing a comprehensive evaluation of imputation methods and enhancing the effectiveness of data filling in real-world applications.

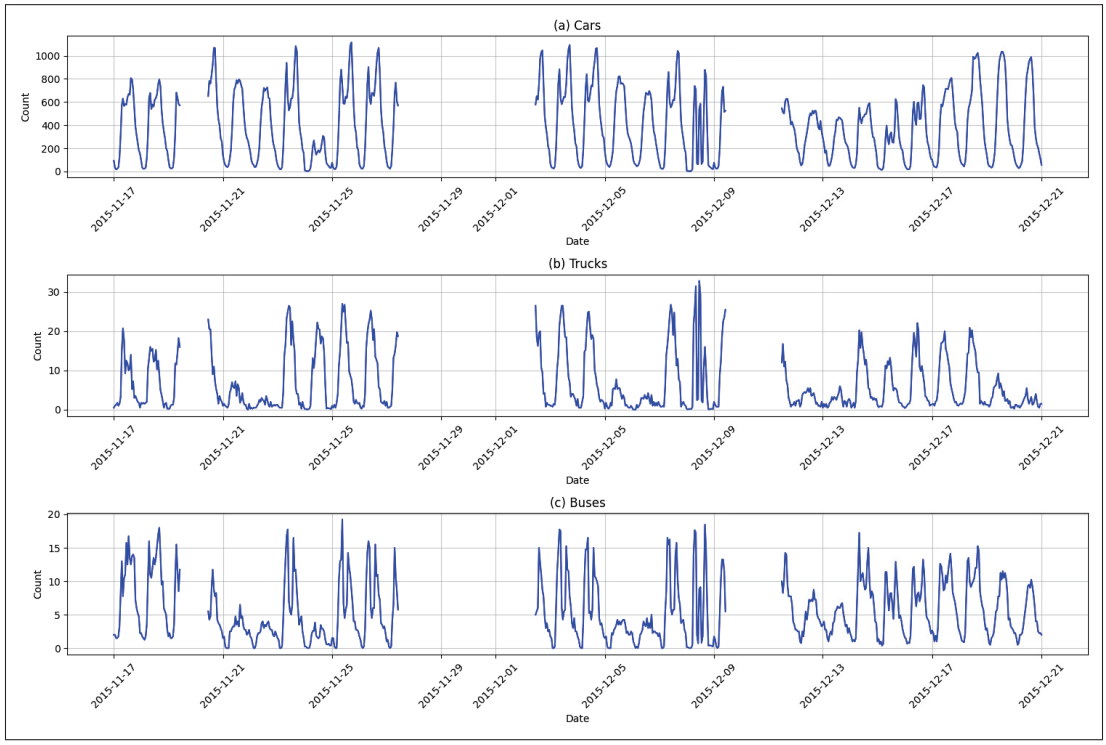

Figure 3.7 Hourly traffic counts for different transport categories

Similarly, the results of the analysis of Brussels dataset further validate the reliability of the proposed metrics. Across all gap sizes, the alignment between no ground truth and traditional

metrics remains high, demonstrating their effectiveness in evaluating imputation quality without ground truth. TimeGPT consistently outperforms the other models, achieving the lowest RMSE, MAE, JSD, WD, and DS values across all gap sizes. Of note, for small gaps, most imputation methods produce reasonable results; however, TimeGPT provides the closest match to the original data in terms of both metrics. For longer gaps, ARIMA and SARIMA struggle to maintain distributional consistency, as indicated by high JSD and WD scores. LSTM performs better for small gaps, but becomes less reliable for 48-hour gaps, as seen in the higher RMSE and WD values. Yet, TimeGPT maintains low scores across all metrics, which clearly demonstrates its ability to preserve the underlying structure and trends in the data.

Collectively, the findings across both datasets and gap sizes consistently demonstrate that the proposed metrics closely align with traditional metrics. This alignment validates their reliability as performance measures, even in the absence of ground truth. The ability of JSD and WD to capture distributional consistency makes these metrics particularly valuable for real-world applications where direct comparison with ground truth is not possible. Figures 3.5 and 3.6 illustrate the performance of the imputation models for Madrid and Brussels datasets, respectively. As can be seen in the figures, TimeGPT's predictions closely follow the original patterns, highlighting TimeGPT's superior performance in preserving statistical and temporal patterns of the data. Despite slight declines for longer gaps, XGBoost and LSTM also demonstrate strong performance for small gap sizes. By contrast, ARIMA and SARIMA exhibit higher JSD and WD values, particularly for large gaps, highlighting their limitations in capturing long-term dependencies and complex patterns. It is important to note, however, that this superior performance of TimeGPT comes at the cost of higher computational overhead compared to XGBoost and LSTM. While XGBoost remains highly efficient and LSTM offers a reasonable trade-off between performance and computation, the Transformer-based architecture of TimeGPT requires more memory and inference time, particularly for longer input sequences. Based on this evidence, we can conclude that, by accurately reflecting the strengths and weaknesses of different imputation methods, JSD, WD, and DS provide a robust alternative to traditional

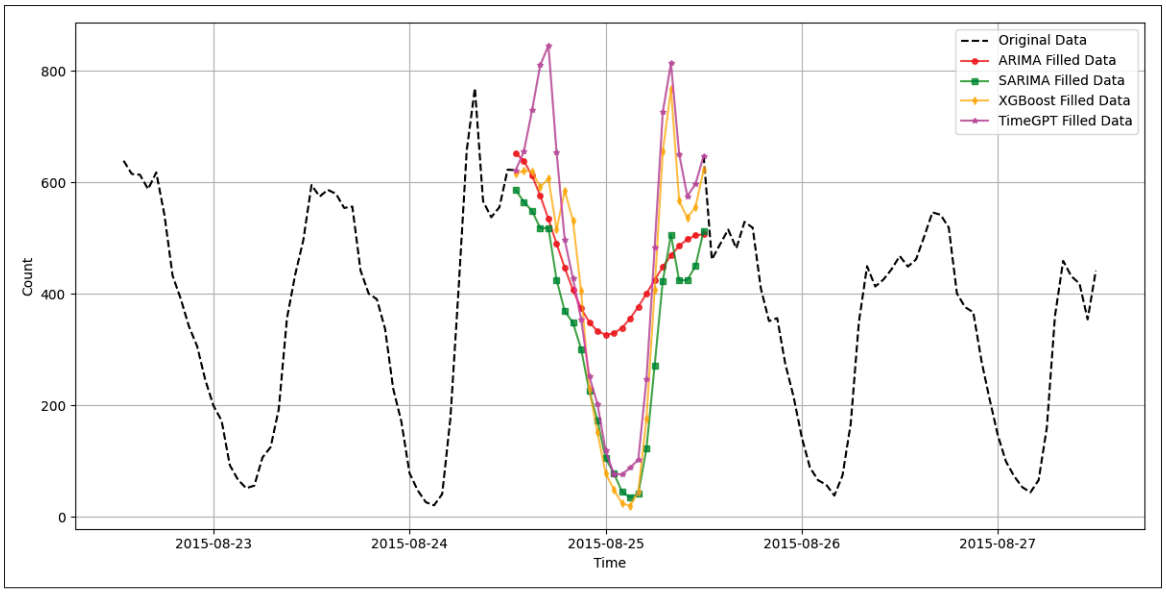

Figure 3.8 Data filling for Montreal dataset

metrics, offering a comprehensive way to evaluate imputation quality. These metrics enhance the assessment of data filling approaches in real-world scenarios where ground truth is unavailable.

3.7 Case Study

Table 3.5 JSD, DS, and WD performance of different imputation methods on the Montreal dataset

Gap size	**Cars**			**Trucks**			**Buses**		
Metrics	**JSD**	**WD**	**DS**	**JSD**	**WD**	**DS**	**JSD**	**WD**	**DS**
ARIMA	0.1949	11.6503	0.3745	0.2742	4.3894	0.2733	0.1642	1.2349	0.1346
SARIMA	0.0963	4.83093	0.2019	0.0918	2.0208	0.1053	0.1249	1.1419	0.1106
XGBoost	0.0788	3.35968	0.1901	0.0876	1.2962	0.0786	0.1036	1.0956	0.0920
TimeGPT	**0.0249**	**1.20856**	**0.0546**	**0.0483**	**1.0618**	**0.0098**	**0.0323**	**0.5098**	**0.0093**

In this section, we demonstrate the practical applicability of the proposed methodology by applying it to a Montreal vehicle dataset [3] characterized by a high missing rate and several consecutive gaps, depicted in Figure 3.7. In what follows, we present hourly traffic counts of

[3] Traffic and Pedestrian Counting Dataset of Montreal City : https://donnees.montreal.ca/dataset/comptage-vehicules-pietons.

the Montreal dataset for different vehicle categories—including cars, trucks, and buses—over a selected period and highlight the numerous consecutive gaps.

The Montreal dataset [3] comprises the data associated with different types of vehicles and recorded since 2008 using sensors placed on traffic lights with a sampling interval of 15 minutes. Each point corresponds to a type of vehicle for 15 minutes. Considering the large volume of missing data, we resampled the data to an hourly frequency by computing the mean within each hour.

This dataset had to be refactored for 5G data generation and traffic prediction (see Ziazet *et al.* (2022)). Accordingly, it was adapted to generate traffic patterns that mimic network slices/services in 5G applications. Using the available urban traffic data, one can create more realistic network scenarios, rather than rely solely on synthetic data. The study refactors the dataset to simulate various classes of network traffic, enabling the use of traffic prediction models and resource management algorithms using machine learning. Using real-world mobility models, it makes it possible to test accuracy of predictive traffic models for 5G network slicing and proactive resource allocation.

To improve usability of the dataset, we applied a clustering approach similar to the one used in Jaumard & Ziazet (2023). Specifically, certain vehicle categories (i.e., buses, school buses, different types of trucks) were grouped together. Similarly, bicycles and motorcycles were combined due to their comparable road behavior.

We then used the previously mentioned models—ARIMA, SARIMA, XGBoost, LSTM, and TimeGPT—to sequentially impute missing values, evaluating their performance using the validated proposed metrics. To enhance model accuracy, we applied hyperparameter optimization and fine-tuning, as previously described in Section. 3.5. Over a 4 month period, from 2015-08-24 to 2015-12-22, we filled 15 distinct gaps in the dataset, with gap lengths ranging from 11 to 144 hours. The total duration of missing data across all gaps amounts to 937 hours. The size of the training data set was initially seven days, it was later increased while imputing the gaps, ensuring that the model used recent temporal patterns before each missing period. Due to the small size

of the initial training set, we did not use LSTM, as deep learning models typically require larger datasets for an effective generalization. The results, highlight the superiority of the TimeGPT model, which achieves the best results across all vehicle categories, with the lowest JSD, DS, and WD values (see Table 3.5). Figure 3.8 provides a visual comparison of the imputation results for the car category, illustrating how different models reconstruct missing values. Among the tested models, TimeGPT effectively preserved the underlying structure of the dataset, retaining both short-term variations and long-term trends. This finding underlines robustness of TimeGPT in handling large gaps. Likewise, XGBoost also displayed excellent performance, with lower JSD, WD and DS values than SARIMA. However, XGBoost was outperformed by TimeGPT in all categories.

Meanwhile, SARIMA performed moderately well, benefiting from its ability to capture seasonal trends, but struggled with abrupt traffic fluctuations, which resulted in higher JSD and WD values as compared to those obtained using XGBoost and TimeGPT. Although SARIMA maintained seasonal consistency, it lacked precision to align imputed values with the original distribution. Conversely, ARIMA consistently delivered the lowest performance across all categories, with the highest JSD and WD values, particularly for car traffic, which exhibited substantial variations. This finding highlights the limitations of ARIMA in handling complex, long-term dependencies. Table 3.5 provides a breakdown of the performance metrics across different vehicle types, reinforcing TimeGPT's effectiveness in preserving data distribution consistency.

The Montreal dataset results underscore the practical applicability of JSD and WD as reliable evaluation metrics. Their ability to capture discrepancies between imputed and original data distributions, even in the absence of ground truth, highlights the value of these metrics for real-world scenarios with incomplete datasets.

3.8 Conclusion

In this study, addressing the challenge of handling multiple consecutive gaps, we introduced an iterative filling approach to sequentially impute missing values in time series data. To evaluate

quality of imputed values without relying on ground truth, we proposed WD, JSD, and DS as alternative evaluation metrics to evaluate data imputation techniques. The results revealed that, by assessing the alignment between the distributions of imputed and pre-gap data, these metrics offer a reliable method for evaluating imputation quality based on statistical consistency, rather than direct comparison with known values. However, these metrics evaluate only distributional similarity, they may overlook essential temporal properties such as phase, autocorrelation, or seasonality patterns. In scenarios where such temporal dynamics are critical, supplementary metrics or visual diagnostics should be considered. Furthermore, experimental results with Brussels and Madrid traffic datasets demonstrated that WD, JSD, and DS effectively capture imputation quality, suggesting their potential for broader applications in environments where ground truth is unavailable. Future research should explore integrating these metrics with adaptive ML models to further improve robustness and accuracy in complex data settings.

CHAPTER 4

MAXIMUM ENTROPY-BASED TRAFFIC GENERATION

Rania Farjallah[1] , Bassant Selim[1] , Brigitte Jaumard[2] , Samr Ali[3] , Georges Kaddoum[4] , Jean-Michel Sellier[5]

[1] Department of Systems Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[2] Department of Computer Science and Software Engineering (CSSE), Concordia University,
1455 De Maisonneuve Blvd. W. Montreal, Québec, Canada H3G 1M8
[3] BNEW RTE ST, AI Accelerator Hub 1, Ericsson,
8275 Trans Canada Route, Saint-Laurent, Quebec, Canada H4S 0B6
[4] Department of Electrical Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[5] Ericsson,
8275 Trans Canada Route, Saint-Laurent, Quebec, Canada H4S 0B6

ARTICLE SUBMITTED, IN IEEE Transactions on Network and Service Management,
September 2025

4.1 Introduction

In recent years, mobile applications have become deeply embedded in everyday life, driving unprecedented growth in mobile data consumption. This surge is not merely a matter of user engagement, but rather reflects changing needs in mobile network management. From social media and streaming to e-commerce and navigation, these services generate vast volumes of traffic, producing rich spatiotemporal datasets that expose detailed patterns of network usage and human activity across urban environments Li *et al.* (2022b, 2023b); Chen *et al.* (2023). The continuous evolution of mobile technology, particularly with the proliferation of advanced 5G networks and the expansion of IoT devices, has accelerated the scale and complexity of data generation. Understanding and effectively interpreting these complex data patterns is essential for enabling intelligent and adaptive approaches to network control and optimization Jin *et al.* (2022); Lu *et al.* (2021); Ding *et al.* (2021). Spatiotemporal datasets play a foundational role in this process, supporting data-driven decisions related to infrastructure provisioning, energy-efficient operations, and service quality enhancement. Beyond operational utility, these

datasets underpin the development and evaluation of artificial intelligence (AI) and ML models used for traffic forecasting, anomaly detection, and resource allocation. However, while these models rely on high-quality, comprehensive traffic datasets, these data are frequently difficult to obtain due to limited infrastructure, irregular sampling, and privacy constraints. As a result, existing datasets frequently fail to capture variability and richness of modern mobile traffic, posing significant challenges for training reliable predictive models and deploying them in dynamic network environments.

To address these limitations, there has been growing interest in generating synthetic yet realistic traffic data as an alternative to empirical measurements Wu, He, Chen, Yu & Zhang (2022). Synthetic datasets provide a practical solution for the development and evaluation of ML models, particularly when access to real-world traffic data is limited due to privacy concerns or incomplete coverage. These datasets can be used to train predictive algorithms, evaluate performance under varying traffic conditions, and support scenario-based planning and testing. More specifically, the ability to generate realistic traffic data that reflects temporal fluctuations and spatial heterogeneity is essential for the simulation of modern mobile networks.

However, although recent advances in generative modeling, including GAN, VAE, and diffusion models, have improved the ability to learn from real traffic data, existing methods still face difficulties with high controllability Yin *et al.* (2022); Xu *et al.* (2022). One major limitation is the difficulty that these models encounter in capturing the multi-scale temporal patterns characteristic of real mobile traffic. Regular patterns such as hourly usage peaks, daily commuting cycles, and weekend variations are frequently underrepresented or smoothed out during generation Zhang (2023); Hui (2023); Li *et al.* (2021). This limits the effectiveness of synthetic data in enabling accurate traffic forecasting and adaptive network resource management. Another common limitation concerns how spatial relationships are represented in synthetic mobile traffic. In real-world networks, traffic patterns are affected by the physical and functional structure of cities, such as correlations between adjacent regions or areas with similar land use. However, many models fail to represent these inter-regional interactions and generate instead the data that appears realistic in some locations, but lack spatial consistency observed in empirical

datasets Yin *et al.* (2022); Gong *et al.* (2022); Xu *et al.* (2022). This can reduce fidelity of generated data when used to train models for spatial prediction, anomaly localization, or load balancing.

Instead of relying on DL architectures, in the present study, we adopt a statistical modeling approach based on the Maximum Entropy Principle (MEP). By encoding known empirical constraints, such as average traffic volumes, variances, and periodicity into the modeling process, the MEP ensures that the generated distributions remain consistent with observed behavior while making no additional assumptions about unknown aspects of the data Jaynes (1957a,b). Said differently, the model reproduces what is known from data without adding patterns that have no empirical basis. It also offers a transparent modeling process, since it is guided entirely by clearly defined constraints Jaynes (1957a,b). These constraints are derived from observed traffic behavior, thus enabling tracing how each one shapes the generated output. Unlike black-box models, where the influence of inputs on results frequently remains unclear, the MEP-based approach provides a direct link between assumptions and outcomes. This transparency allows the model to be adapted to specific scenarios, as well as justified in terms of the data it reflects.

Most importantly, our MEP-based approach offers a significant advantage in scenarios where empirical data are limited. Unlike deep generative models, which typically require large-scale training data and extensive tuning, the MEP framework defines a distribution based on a small number of observed statistical properties Jaynes (1957a,b). Such data efficiency makes it particularly suitable for mobile environments with sparse measurements or short observation histories. In addition, the MEP framework's reliance on statistical constraints, rather than on extensive empirical datasets, ensures greater robustness and interpretability, facilitating its application in regions with developing network infrastructures or stringent data privacy regulations.

We evaluate our approach using real-world urban traffic datasets collected by the City of Calgary, recorded at 15-minute intervals, and preprocess the data through scaling, shifting, and transformation functions prior to modeling. These adjustments are deemed to be necessary

because the statistical patterns of urban traffic differ from those typically found in network traffic datasets, such as those related to gaming Tarng, Chen & Huang (2008) or video streaming Rahman *et al.* (2018). Our Maximum Entropy model is designed to capture the statistical structure of the transformed data, thus providing a flexible, distribution-based approach to traffic synthesis that avoids the complexity of black-box generative neural networks.

Major contributions of this paper can be summarized as follows:

- The proposed novel approach for traffic generation based on the MEP, offers a transparent, interpretable, and data-efficient alternative to conventional deep generative models. This approach explicitly incorporates known statistical properties to ensure realistic and unbiased synthetic traffic.
- We explore different statistical formulations within the Maximum Entropy framework and demonstrate how they influence the model's ability to reflect observed traffic patterns.
- We validate effectiveness of our approach using two traffic datasets with 15-minute granularity from the City of Calgary. Through comprehensive evaluation, we show that the generated traffic pattern exhibit high fidelity, closely aligning with the statistical behavior and temporal dynamics of empirical observations.
- To ensure numerical stability and enhance generalization in diverse urban traffic scenarios, we develop a robust optimization and tuning strategy using L-BFGS-B method and cross-validation.

The remainder of this paper is organized as follows. Section 4.3 reviews the state-of-the-art in network traffic generation, highlighting limitations in traditional approaches and advances in machine learning and generative models. Thereafter, section 4.3.1 provides a detailed description of the proposed MEP-based traffic modeling framework, including the optimization process, and hyperparameter tuning strategy. Section 4.4.3 presents experimental results, evaluating the performance of the proposed method on real-world traffic data. Finally, section 4.5 concludes the paper with a summary of findings and directions for future work.

4.2 Literature Review

In recent years, the synthesis of mobile traffic has gained considerable importance due to its crucial role in network optimization, service evaluation, and adaptive network management Zhang, Tang, Zhang, Ouyang & Wang (2015). As network environments become increasingly complex and dynamic, generating realistic traffic patterns has become essential for effective network evaluation and strategic planning. However, while traditional approaches primarily relied on analytical modeling and simulation-based techniques to estimate mobile data Biasio, Chiariotti, Polese, Zanella & Zorzi (2019); Bothe, Qureshi & Imran (2019), these methods frequently struggled to capture the inherent complexity and high-dimensional variability observed in real-world scenarios. Recent research in this field addressed both traffic prediction and traffic generation. While traffic prediction focused on forecasting future traffic based on historical data, traffic generation aimed to synthesize realistic traffic samples that reflect real-world conditions for simulation and planning.

With advancements in ML, more robust methodologies to accurately model and synthesize mobile traffic emerged. In previous research, time series forecasting methods using autoregressive models, RNN, and LSTM networks were extensively employed to capture temporal correlations within traffic data Zhu, Liu & Lin (2020); Dalgkitsis, Louta & Karetsos (2018a); Cardoso & Vieira (2019a). For instance, Wang et al. Wang *et al.* (2022) proposed an adaptive mechanism using Q-learning to dynamically adjust thresholds in LSTM models, which significantly improved their responsiveness to fluctuating traffic conditions. Beyond temporal modeling, the complex behavior of mobile traffic was found to be strongly influenced by spatial factors such as user mobility, base station topology, and urban environments. Recent research also increasingly adopted spatio-temporal learning techniques that integrate spatial and contextual dependencies. For instance, Graph Convolutional Networks (GCN)-based models were coupled with attention mechanisms to effectively capture interdependencies across both time and space Li *et al.* (2023a); He, Chen, Wu, Yu & Zhou (2022). For instance, Fang et al. Fang, Ergüt & Patras (2022) proposed a model, which used dynamic graphs derived from handover frequency to capture interactions between network nodes. In another relevant study, authors Wu *et al.* (2022) extended

this by integrating GCN with GAN, thus enabling the model to learn spatial correlations across multiple cities. The MVSTGN model Yao, Gu, Su & Guizani (2023) further advanced spatial modeling by partitioning urban areas into multi-attribute graphs, facilitating the learning of low-dimensional yet informative spatial representations of traffic dynamics.

Yet, while traffic prediction was widely explored in previous studies, traffic generation is emerging as a complementary research direction focused on generating realistic mobile traffic data for modeling and network simulation. In this context, GAN architectures became prominent, as evidenced by methods employing multiple GAN to generate diverse network flows simultaneously Ring, Schlör, Landes & Hotho (2019). Similarly, a GAN-based data augmentation model incorporating LSTM networks was used to generate realistic cellular traffic sequences that preserve temporal dependencies, thereby enhancing the performance of sequence-to-sequence prediction models Wang, Hu, Min, Zhao & Wang (2021).

An important evolution within generative modeling is the incorporation of contextual urban factors, such as population density, land use, and points of interest (POIs), directly influencing mobile traffic synthesis. Context-aware architectures like SpectraGAN implemented an encoder that transformed these inputs into spatial embeddings, guiding a conditional GAN in generating high-fidelity, city-scale spatiotemporal traffic data Xu *et al.* (2021b). In Hui *et al.* (2022) and Hui *et al.* (2023), the authors further extended this concept by developing contextual embeddings for base stations and incorporating device-specific features to generate traffic realistically. Other relevant studies used CNNs to extract spatial urban image characteristics, enabling context-driven synthetic mobile traffic generation at the web service level Sun *et al.* (2022). Moreover, transfer learning paradigms, based on hierarchical GAN architectures, enabled improved traffic generation across distinct urban areas through unified knowledge graph alignment techniques Zhang *et al.* (2023). One of the most recent advancements in traffic generation involves the use of transformer-based architectures. The PAC-GPT approach, which uses multiple GPT-3 language models in sequence, demonstrated a remarkable potential in efficiently generating highly diverse and realistic network traffic Zhang *et al.* (2015). This method benefits from the strong sequence modeling capabilities of transformers and offers an alternative to traditional GAN-based systems.

Despite their demonstrated utility, GAN-based models frequently encounter challenges in mobile traffic synthesis, particularly in terms of effectively modeling contextual dependencies and generalizing across different spatial environments. This limitation considerably reduces flexibility and representational fidelity, particularly when specific contextual environmental conditions must be accurately modeled. These shortcomings highlight the need for a more controllable and reliable approach capable of producing diverse, high-fidelity traffic data that would remain sensitive to spatial and contextual variations.

4.3 Maximum entropy principle model and hyperparameter optimization

In this section, we provide a detailed description of the MEP and the optimization strategy used to infer traffic flow distributions under empirically derived constraints.

4.3.1 Maximum Entropy Principle (MEP)

The *Maximum Entropy Principle* (MEP), originally introduced by Jaynes in 1957 Jaynes (1957a,b), establishes a systematic methodology for probabilistic inference under incomplete information. When the available information about a system is limited to partial knowledge—such as mean values or higher-order moments—multiple probability distributions may satisfy the corresponding constraints. MEP points to the distribution that maximizes entropy, ensuring that the result is maximally unbiased and noncommittal with respect to the missing information. In the context of traffic generation, where full knowledge of underlying distributions is frequently unavailable due to the dynamic and heterogeneous nature of user behavior and network demands, MEP provides a principled approach to infer realistic traffic profiles while adhering to known constraints, such as average packet rates, flow durations, or session inter-arrival times. This approach is grounded in the concept of entropy, originally introduced in information theory by Shannon, which quantifies the uncertainty in a probability distribution.

Shannon's information theory Karmeshu & Pal (2003)introduces entropy as a measure of uncertainty associated with a discrete probability distribution $p = \{p_1, p_2, \ldots, p_n\}$. Mathematically,

it is expressed as shown below:

$$H(p) = -\sum_{i=1}^{n} p_i \log p_i \tag{4.1}$$

This expression is derived from several intuitive axioms: continuity, symmetry under permutation, maximality under uniformity, and the branching principle. Shannon's goal was to measure information loss in communication systems and although this formula mirrors the one for thermodynamic entropy, the connection was originally coincidental. Jaynes Jaynes (1957a,b) later formalized this connection by interpreting entropy as a tool for inference, thereby laying the foundation for MEP.

Historically, the challenge of assigning probabilities under uncertainty has prompted different philosophical perspectives. In *Principle of Insufficient Reason* Jaynes (1957a,b), Laplace suggested that, in the absence of distinguishing information, all outcomes should be considered equally likely. MEP extends this idea by accommodating additional constraints and justifying the resulting distribution via entropy maximization. This aligns with the *subjectivist* interpretation of probability, where it represents degrees of belief based on current knowledge, as opposed to the *objectivist* view which relies strictly on empirical frequencies Jaynes (1957a,b).

Building on this foundation, the MEP framework formulates entropy maximization as a constrained optimization problem. Let us assume that a system has n possible discrete states, each associated with a probability p_i. To reflect known information, the probability distribution must satisfy certain constraints—namely, normalization and the expected values of specific observable functions $f_j(x)$. These constraints are given by the following Kouvatsos (2003):

$$\sum_{i=1}^{n} p_i = 1 \tag{4.2}$$

$$\sum_{i=1}^{n} p_i f_j(x_i) = \langle f_j(x) \rangle, \quad j = 1, 2, \ldots, m \tag{4.3}$$

To determine distribution $\{p_i\}$ that maximizes the entropy subject to these constraints, we use the method of Lagrange multipliers. This leads to the construction of the following Lagrangian function:

$$\begin{aligned}\mathcal{L} = &- \sum_{i=1}^{n} p_i \log p_i + \lambda_0 \left(\sum_{i=1}^{n} p_i - 1 \right) \\ &+ \sum_{j=1}^{m} \lambda_j \left(\sum_{i=1}^{n} p_i f_j(x_i) - \langle f_j(x) \rangle \right)\end{aligned} \tag{4.4}$$

Maximizing $\mathcal{L}$ involves taking its derivative with respect to each p_i and setting it equal to zero, as shown below:

$$\frac{\partial \mathcal{L}}{\partial p_i} = - \log p_i - 1 + \lambda_0 + \sum_{j=1}^{m} \lambda_j f_j(x_i) = 0 \tag{4.5}$$

Solving for p_i, we obtain the following:

$$p_i = \exp\left(\lambda_0 - 1 + \sum_{j=1}^{m} \lambda_j f_j(x_i) \right) \tag{4.6}$$

To satisfy the normalization condition, we define the partition function Z as follows:

$$Z = \sum_{i=1}^{n} \exp\left(\sum_{j=1}^{m} \lambda_j f_j(x_i) \right) \tag{4.7}$$

Substituting Z into the expression for p_i yields the following final form of the maximum entropy distribution:

$$p_i = \frac{1}{Z} \exp\left(\sum_{j=1}^{m} \lambda_j f_j(x_i) \right) \tag{4.8}$$

This form ensures that all imposed constraints are satisfied. The expected value of each function $f_j(x)$ can be computed by differentiating the log-partition function as shown below:

$$\langle f_j(x) \rangle = - \frac{\partial}{\partial \lambda_j} \ln Z \tag{4.9}$$

The corresponding maximum value of entropy under these constraints is given by Eq. (4.10):

$$S_{\max} = \lambda_0 + \sum_{j=1}^{m} \lambda_j \langle f_j(x) \rangle \tag{4.10}$$

Moreover, the variance of each constraint function is derived from the second derivative of the log-partition function:

$$\Delta^2 f_j = \langle f_j^2 \rangle - \langle f_j \rangle^2 = \frac{\partial^2}{\partial \lambda_j^2} \ln Z \tag{4.11}$$

In 5G traffic modeling, the MEP offers a powerful solution to the challenge of synthesizing realistic traffic traces from limited or aggregate statistical data. Instead of assuming arbitrary models or relying solely on empirical sampling, MEP allows for the generation of traffic patterns that match the observed metrics—such as average packet size, session duration, inter-arrival times, or throughput—without introducing unjustified assumptions about higher-order structure. This makes it possible to create synthetic traffic datasets that are both statistically consistent and maximally non-committal.

The aforementioned approach is particularly valuable for network simulation, emulator design, and AI-driven traffic classification tasks, where a lack of fine-grained labeled data tends to limit model accuracy. By adjusting the constraint functions, MEP can flexibly adapt to diverse 5G use cases, from enhanced mobile broadband (eMBB) to massive machine-type communications (mMTC), and generate traffic scenarios reflecting variability and complexity of real-world deployments.

Accordingly, MEP serves not only as a theoretical framework, but also as a practical tool for bridging the gap between abstract statistical knowledge and concrete, reproducible traffic generation. This makes it well-suited for evaluating 5G architectures under realistic, yet controlled, conditions.

4.3.2 Optimization of MEP-based models

To improve the predictive performance and stability of the MEP-based model, several components of the optimization process should be carefully designed and tuned. These components include parameter initialization, choice of optimization algorithm, probabilistic structure through log-partition functions and hyperparameter tuning. In this study, we systematically address each of these components to develop a robust training pipeline for learning traffic distributions under empirical constraints.

4.3.2.1 Parameter Initialization

The Maximum Entropy model includes several key parameter groups, each requiring a careful initialization to ensure numerical stability and reliable convergence.

At the core of the model lies latent variable $v_t \in \mathbb{R}^N$, which represents a linear combination of lagged traffic inputs modulated by temporal interaction weights and bias terms. It is defined as shown below:

$$v_t = h + \sum_{d=1}^{\text{lag}} J_d x_{t-d}, \tag{4.12}$$

where $h \in \mathbb{R}^N$ is a bias vector, $J_d \in \mathbb{R}^{N \times N}$ are interaction matrices corresponding to lag d, and x_{t-d} are past traffic observations. This latent variable serves as input to the log-partition function and directly governs the expected value and shape of the resulting probability distribution under the Maximum Entropy formulation.

To ensure stable optimization and improve convergence, parameter initialization is aligned with the analytical form of the chosen distribution (e.g., exponential-like or Gaussian-like), thus ensuring that the initialization lies within a feasible and well-scaled region of the parameter space. The model parameters are initialized as follows:

- **Interaction Matrices** (J): These matrices encode temporal and inter-zone dependencies. Each J_d is initialized with Gaussian noise (mean 0, standard deviation 0.01) to introduce weak prior structure while preserving flexibility during training.

- **Bias Vectors (h)**: Representing baseline traffic levels in each zone, vector h is initialized to 10% of the empirical mean traffic. This provides a small positive offset encouraging faster convergence without overly biasing the model.
- **Distribution Parameters (a)**: These parameters control the dispersion of the distribution:
  - *Exponential-like models:* The probability density is given by the following:

$$p(x) \propto e^{-(a-v)x}, \tag{4.13}$$

    which requires $a > v$ to ensure positivity and normalizability. Therefore, a is initialized to exceed the maximum observed traffic values.
  - *Gaussian-like models:* The distribution takes the following form:

$$p(x) \propto e^{-ax^2+vx}, \tag{4.14}$$

    where a relates to the inverse variance. In this case, a is initialized proportionally to the empirical variance and is clipped to a small positive threshold to maintain numerical stability.
- **Mixture Weights (w_k)**: In mixture models, component weights w_k control the contribution of each distribution component. They are initialized uniformly in logit space and normalized via softmax to satisfy the following constraint:

$$\sum_{k=1}^{K} w_k = 1. \tag{4.15}$$

4.3.2.2 Optimization Procedure

The model parameters are optimized by minimizing the *Negative Log-Likelihood (NLL)* of the observed data, with added regularization to improve generalization. The parameter set is defined as follows:

$$\theta = \{J, h, a, \text{weights}\}, \tag{4.16}$$

where $J \in \mathbb{R}^{\text{lag}\times N\times N}$ are temporal interaction matrices, $h \in \mathbb{R}^N$ are bias terms, $a \in \mathbb{R}^{K\times N}$ are per-zone distribution parameters, and weights $\in \mathbb{R}^K$ are mixture component weights (when applicable). The objective function to be minimized is a regularized NLL, given by Eq. (4.17):

$$\min_{\theta} \text{NLL}(\theta) + \lambda_{\text{reg}} \|J\|_2^2, \tag{4.17}$$

where $\lambda_{\text{reg}} = 0.01$ is a small constant that penalizes large temporal interactions, thus encouraging smoother and more generalizable models. To compute the NLL, we rely on the latent input vector v_t (see Eq. (4.12)), which integrates lagged traffic observations through temporal interaction weights and a bias term.

This loss function is derived from the exponential family form of distributions, as established by the Maximum Entropy Principle. The general form of the model distribution is as follows:

$$p(x) = \frac{1}{Z(\lambda)} \exp\left(\sum_{j=1}^{m} \lambda_j f_j(x)\right), \tag{4.18}$$

where λ_j are learned parameters corresponding to constraint functions $f_j(x)$. Log-partition function $\log Z(\lambda)$ ensures proper normalization and encodes the statistical structure of the distribution. Minimizing the NLL under this form aligns the model with the empirical data while preserving consistency with the selected distributional assumptions.

Depending on the distribution type, the NLL is computed differently. In what follows, we provide both single-component and mixture-based formulations for each supported distribution.

4.3.2.2.1 Exponential Distribution

For exponential models, the per-zone conditional log-likelihood is computed as follows:

$$\log p(x_{t,i} \mid v_{t,i}) = -\log Z_i + x_{t,i} \cdot v_{t,i} \tag{4.19}$$

The corresponding log-partition function $\log Z_i$ is defined as follows:

$$\log Z_i = \begin{cases} -\log(a_i - v_{t,i}) & \text{(single)} \\ \log\left(\sum_{k=1}^{K} w_k \cdot \frac{1}{a_{k,i} - v_{t,i}}\right) & \text{(mixture)} \end{cases} \tag{4.20}$$

The total negative log-likelihood for exponential models is given in Eq. (4.21):

$$\text{NLL}_{\text{exp}} = \sum_{t=D}^{T-1} \sum_{i=1}^{N} \left[\log Z_i - x_{t,i} \cdot v_{t,i}\right] \tag{4.21}$$

4.3.2.2.2 Gaussian-like Distribution

For Gaussian-like models, the conditional log-likelihood includes a quadratic term as shown below:

$$\log p(x_{t,i} \mid v_{t,i}) = -a_i x_{t,i}^2 + x_{t,i} \cdot v_{t,i} - \log Z_i \tag{4.22}$$

Log-partition function $\log Z_i$ is given by the following:

$$\log Z_i = \begin{cases} -\frac{1}{2}\log a_i + \frac{v_{t,i}^2}{4a_i} + \log\left(1 + \text{erf}\left(\frac{v_{t,i}}{2\sqrt{a_i}}\right)\right) & \text{(single)} \\ \log\left(\sum_{k=1}^{K} w_k \cdot \exp(\log Z_{k,i})\right) & \text{(mixture)} \end{cases} \tag{4.23}$$

The complete NLL for Gaussian-like models then becomes

$$\text{NLL}_{\text{gauss}} = \sum_{t=D}^{T-1} \sum_{i=1}^{N} \left[a_i x_{t,i}^2 - x_{t,i} \cdot v_{t,i} + \log Z_i\right] \tag{4.24}$$

In previous research, a variety of optimization algorithms was proposed for high-dimensional parameter learning, including stochastic gradient descent (SGD) Li, Xiong & Shang (2022a), Adam Kingma & Ba (2017), Root Mean Square Propagation (RMSprop) Xu, Zhang, Zhang & Mandic

(2021a), conjugate gradient methods, and second-order approaches such as Broyden Fletcher Goldfarb Shanno (BFGS) Dai (2002) and its limited-memory variant L-BFGS Liu & Nocedal (1989). However, although SGD-based methods are widely used in deep learning due to their scalability and simplicity, they frequently require careful tuning and may converge slowly or unstably in deterministic environments. Standard BFGS offers robust convergence, but becomes impractical in high dimensions due to its memory requirements and full Hessian computation.

To overcome these challenges, in this study, we adopt the *L-BFGS-B* algorithm Zhu, Byrd, Lu & Nocedal (1997), a limited-memory quasi-Newton method tailored for large-scale problems with simple bound constraints. L-BFGS-B combines second-order curvature information with efficient memory use and allows for enforcing positivity constraints (e.g., $a_i > 0$) on model parameters. It also provides significantly faster and more stable convergence as compared to SGD in our structured optimization landscape.

During training, several key strategies are employed to ensure model robustness and generalization. The process begins by initializing parameters based on empirical statistics computed from the data, which provides a meaningful starting point for optimization. To maintain stability during training, bound constraints are applied, particularly on sensitive parameters such as a. In addition, to control model complexity and prevent overfitting, regularization is applied to the temporal interaction matrix J. To further enhance generalization and avoid overtraining, early stopping is used, with training terminated once performance on a validation set shows no further improvement.

Overall, this optimization procedure enables effective training of high-dimensional probabilistic models with rich temporal and spatial structure, while ensuring well-calibrated uncertainty through the Maximum Entropy formulation.

4.3.2.3 Hyperparameter Tuning and Cross-Validation

Hyperparameter selection exerts a strong impact on model expressiveness and generalization. We adopt a rolling-window cross-validation approach to systematically evaluate the following configurations:

- Temporal lag values: $\{24, 48, 96\}$,
- Distribution types: Exponential, Gaussian, mixture-exponential, and mixture-Gaussian,
- Number of mixture components: $K \in \{1, 2, 3\}$.

Each configuration is evaluated using the mean squared error (MSE) computed on held-out validation windows. The optimal hyperparameter set is selected based on the lowest average MSE across all validation splits.

4.4 Numerical experiments

To evaluate effectiveness of our proposed MEP-based traffic generation approach, we conducted a series of numerical experiments using real-world urban mobility data. In this section, we first describe the datasets and the preprocessing steps applied to align their characteristics with mobile network traffic. This is followed by a description of the experimental setup used to evaluate our approach. The section concludes with a detailed presentation and discussion of the results.

4.4.1 Datasets Description

In this study, we used two publicly available traffic datasets from the City of Calgary's Open Data Portal[1]. Both datasets provide traffic observations and are recorded every 15 minutes, enabling detailed modeling of temporal dynamics in urban mobility.

[1] City of Calgary Open Data Portal: https://data.calgary.ca

The first dataset comprises bicycle and pedestrian counts collected via Eco-Counter[2] sensors installed at key active transportation corridors. These sensors detect cyclists using embedded induction loops and pedestrians via infrared sensors. The dataset spans the data collected from January 1, 2018 to December 31, 2024, providing continuous 15-minute interval data supporting the analysis of mobility patterns across daily, weekly, monthly, and annual timeframes.

The second dataset consists of permanent vehicle count station data[3], gathered from fixed traffic sensors deployed across the city. These stations continuously record vehicle flows every 15 minutes and are primarily used to monitor long-term traffic trends. The dataset used in this study spans the data collected from January 1, 2020 to December 31, 2021, enabling the analysis of seasonal and weekly variations the aforementioned 2-year period. It is also commonly used to normalize short-term portable traffic studies.

Together, these datasets provide comprehensive insights into Calgary's multimodal traffic landscape and serve as a robust foundation for the development and evaluation of generative models for spatiotemporal traffic synthesis.

4.4.2 Setup of numerical experiments

To generate synthetic mobile network traffic from urban mobility data, we adopted a transformation strategy inspired by Ziazet *et al.* (2022), who proposed refactoring real-world vehicular and pedestrian traffic into representative 5G service classes. The aforementioned approach involves mapping categories of mobility data—such as cars, cyclists, and pedestrians—to different types of mobile traffic (e.g., video streaming, cloud gaming, MIoT). This mapping is complemented by the application of scaling, shifting, and pattern reshaping techniques to ensure the resulting synthetic traffic aligns with the temporal and intensity characteristics of typical 5G network demands.

[2] Eco-Counter dataset: https://www.eco-counter.com/

[3] Calgary Permanent Traffic Count in 2020: https://data.calgary.ca/Transportation-Transit/2020-Traffic-Counts-at-Permanent-Stations/undq-t5qf/about_data

In our case, we adapted this methodology by using traffic count data recorded at 15-minute intervals to maintain high temporal fidelity. Each mobility observation was associated with a specific service slice and is subsequently rescaled to reflect the expected traffic volume of the corresponding mobile service. This mapping, detailed in Table 4.1, allowed us to emulate distinct service patterns and peak usage times, such as evening surges for video streaming or midday peaks for urban mobility.

Table 4.1 Mapping of Urban Traffic to 5G Slices

Service Chains	**Slices**	**User Types**
MIoT	Slice 0	Cars
Industry 4.0	Slice 1	Pedestrians
Video Streaming	Slice 2	Bikes

Once transformed, the data underwent a temporal normalization procedure to remove periodic fluctuations and stabilize intra-week variability. This was achieved by grouping observations according to their exact time slot within the week (e.g., “Monday at 08:15”) and computing the mean and standard deviation of counts for each cluster and time slot. Each data point was then normalized using its corresponding temporal pattern statistics. This approach allowed the model to focus on deviations from expected behavior at specific times, rather than being dominated by routine patterns. As a result, it could learn more meaningful patterns, such as unusual peaks or drops, while preserving the relative dynamics and anomalies that are essential for accurate modeling and prediction.

This normalization process ensured that all spatial clusters were treated consistently and that temporal behaviors—such as rush hours or weekend effects—were captured in a standardized manner. It also improved robustness and generalization capacity of downstream machine learning models trained for traffic prediction, classification, or resource allocation.

Further detail on the optimization procedures, including parameter initialization methods, the selection and application of optimization algorithms, hyperparameter tuning rationale, and the

cross-validation strategy used to achieve robust and stable model performance, is provided in Section 4.3.2.2.

To quantitatively evaluate the generated traffic, we used standard performance metrics—namely, RMSE, MAE, and the Coefficient of Determination (R^2). RMSE and MAE capture the magnitude of error between real and generated traffic data, while R^2 measures how well the generative model explains the variance in the observed data. This enabled a systematic comparison of generative quality across the different distributional assumptions explored in our MEP-based approach.

4.4.3 Results and discussion

Instead of focusing on architectural comparisons, in this study, we prioritized evaluating the statistical consistency of the generated traffic with realistic patterns observed in actual mobile network data. Considering that empirical traffic frequently exhibits characteristics aligned with Gaussian or Poisson-like behaviors Navarro-Ortiz *et al.* (2020), we explored different distributional assumptions within the Maximum Entropy framework to ensure fidelity to the intrinsic properties of traffic data. This approach allowed for the generation of traffic patterns that are both temporally coherent and statistically representative, which is critical for downstream tasks such as simulation, anomaly detection, and network performance evaluation.

Table 4.2 Evaluation metrics (RMSE, MAE, R^2) of different distributions on the MIoT dataset across clusters

Cluster	**Cluster 0**			**Cluster 1**			**Cluster 2**		
Distribution	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2
Gaussian	**9.69**	**6.80**	**0.9940**	9.38	7.04	0.9819	**16.28**	**11.85**	**0.9760**
Exponential	10.95	7.57	0.9910	11.47	8.82	0.9712	17.21	12.43	0.9734
Mix. Gaussian	10.84	7.47	0.9933	**8.84**	**6.58**	**0.9839**	17.12	12.38	0.9752
Mix. Exponential	25.22	17.80	0.9601	11.70	8.94	0.9729	17.22	12.44	0.9711

The performance of our MEP-based traffic generation model was evaluated by comparing the synthetic traffic it generates with real observations. We evaluated both visual and statistical fidelity using various distributional assumptions, including Gaussian, exponential, and their

mixtures. The detailed numerical results for each service type are summarized in Table 4.2 for MIoT traffic, Table 4.3 for Industry 4.0 traffic, and Table 4.4 for video streaming traffic. These tables summarize the RMSE, MAE, and R^2 metrics for the evaluated distributions, thus providing a quantitative basis for the subsequent analysis.

Table 4.3 Evaluation metrics (RMSE, MAE, R^2) of different distributions on the Industry 4.0 dataset across clusters

Cluster	**Cluster 0**			**Cluster 1**			**Cluster 2**		
Distribution Type	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2
Gaussian	9.46	6.48	0.6265	7.35	4.90	0.5190	21.16	14.35	0.8564
Exponential	9.73	6.37	0.6350	7.63	5.16	0.4651	21.53	14.57	0.8531
Mix. Gaussian	**9.44**	**6.30**	**0.6260**	**6.56**	**4.08**	**0.5992**	**20.43**	**13.33**	**0.8928**
Mix. Exponential	21.90	14.93	-0.8923	17.98	12.99	-1.9706	21.45	14.58	0.8521

The comparative results, visualized through CDF and KDE plots in Figures 4.1 and 4.2, reveal that, across all clusters and services, the Gaussian and mixture Gaussian distributions consistently provided the closest alignment to actual traffic patterns. The CDF illustrates the proportion of observations falling below a given value, enabling for a direct assessment of the alignment between generated and real traffic distributions over the entire range. The KDE plots approximate the underlying probability distribution of the data, allowing us to visually compare the shape, spread, and modality between actual and synthetic samples.

For MIoT traffic, Gaussian-based models excelled in replicating multi-modal behaviors and capturing the full shape of the distribution. The KDE plots in Figure 4.2 show a close alignment of peaks and troughs between the actual and generated data, while the CDF plots in Figure 4.1 exhibit the minimal divergence across the cumulative probability range, indicating a strong match not only in central tendency but also in tail behavior. The quantitative metrics in Table 4.2 reinforce this observation, with Gaussian models achieving the lowest RMSE and MAE in Clusters 0 and 2, and mixture Gaussian models outperforming others in Cluster 1. This demonstrates that, while Gaussian models are robust across most traffic patterns, mixture Gaussians offer added flexibility in handling clusters with more complex.

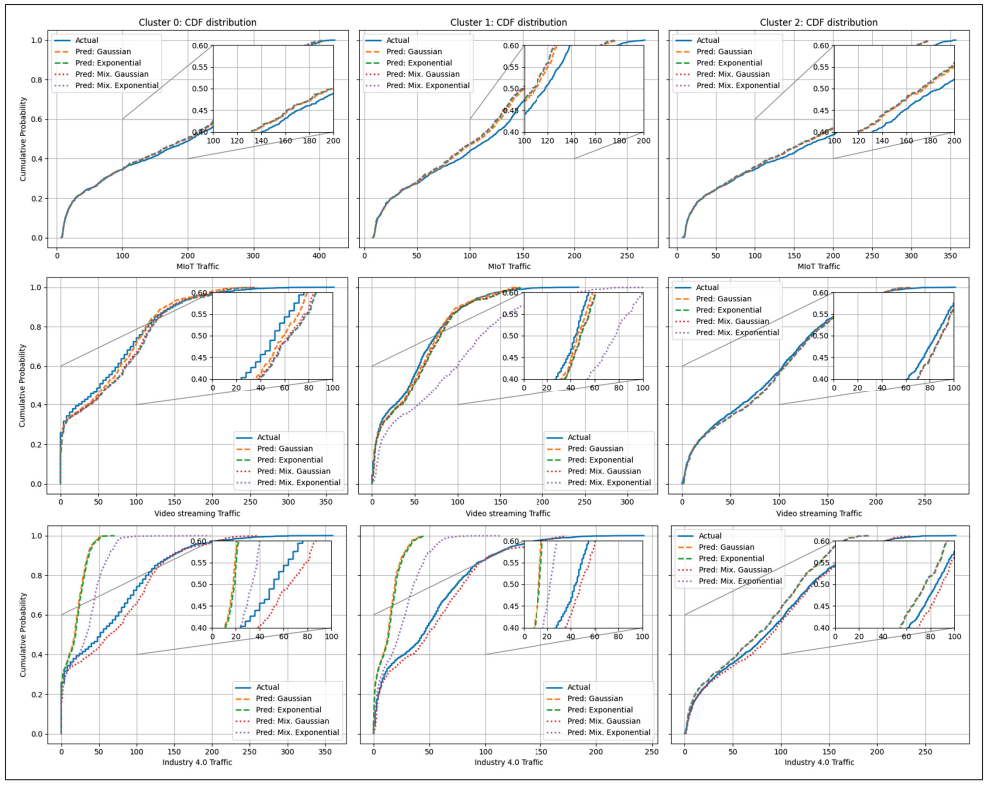

Figure 4.1 CDF comparison between real vs. generated traffic in each service type

In the case of Industry 4.0 traffic, mixture Gaussian models achieved the lowest RMSE, MAE, and the highest R^2 values in all clusters (see Table 4.3). These models demonstrated the ability to preserve subtle distributional features such as secondary peaks or gradual decay rates usually smoothed over by pure Gaussian models. Conversely, both exponential and mixture exponential models consistently underperformed: the KDE plots show oversimplified distributions with distorted peak amplitudes, and the CDF plots reveal misalignment in both mid-range probabilities and tail regions, confirming their inability to represent the more intricate statistical patterns present in this dataset.

Video streaming traffic, characterized by high throughput and pronounced periodicity, was best modeled by Gaussian distributions across all clusters (see Table 4.4). The KDE plots demonstrate an accurate reproduction of both the magnitude and location of primary peaks, while CDF curves show a strong overlap in cumulative probabilities, suggesting that temporal regularity

of video streaming is well-suited to Gaussian modeling. Mixture Gaussian models offered negligible improvement here, reinforcing that the simpler Gaussian assumption is sufficient for traffic with this degree of regularity.

Overall, the quantitative metrics (RMSE, MAE, R^2) are well-aligned with the visual analyses, with mixture Gaussian models being generally the most adaptable, particularly for heterogeneous and multi-modal traffic patterns, whereas pure Gaussian models excelled for traffic with stable periodicity and moderate variance. Exponential-based models, including mixtures, consistently failed to capture statistical complexity of the observed traffic, making them unsuitable for high-fidelity synthetic traffic generation in this context. Taken together, the results confirmed that

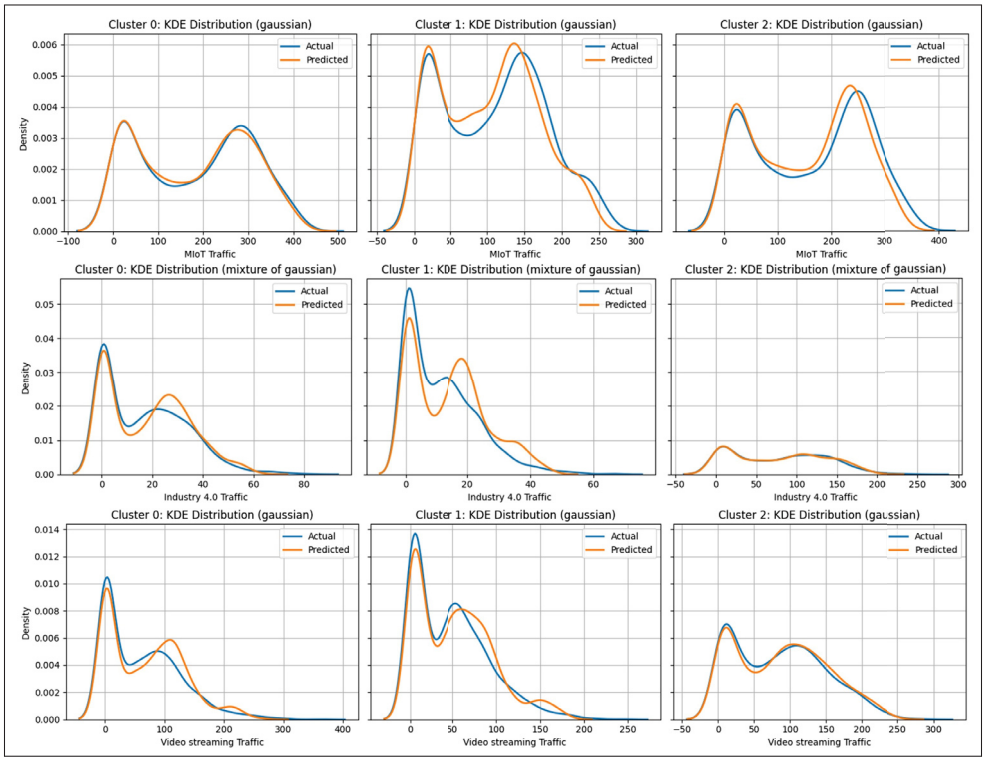

Figure 4.2 KDE distribution comparison between real vs. generated traffic best-performing distributions in each service type

our MEP-based framework, when combined with Gaussian or mixture Gaussian distributions, can robustly reproduce both the temporal coherence and statistical structure of real-world network

traffic, highlighting its suitability for realistic synthetic traffic generation in mobile network contexts.

Table 4.4 Evaluation metrics of different distributions on the Video Streaming dataset across clusters

Cluster Number	**Cluster 0**			**Cluster 1**			**Cluster 2**		
Distribution Type	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2
Gaussian	**31.61**	**20.23**	**0.7497**	**17.05**	**11.70**	**0.8439**	**23.16**	**16.18**	**0.8639**
Exponential	32.20	20.11	0.7408	17.19	11.80	0.8415	23.19	16.20	0.8636
Mix. Gaussian	32.04	20.77	0.7428	17.08	11.72	0.8435	23.18	16.19	0.8637
Mix. Exponential	32.34	21.01	0.7380	49.27	35.91	-0.3032	23.99	17.53	0.8443

4.5 Conclusion

In this paper that aimed to address critical limitations in existing generative models, we introduced a MEP-based approach for generating synthetic traffic. Using comprehensive real-world datasets from the City of Calgary, the results of our evaluation demonstrated that the proposed method can faithfully reproduce the complex temporal dynamics and statistical properties characteristics of urban network traffic. Both Gaussian and Gaussian mixture models achieved a high fidelity in capturing periodic fluctuations, multi-modal structures, and fine-grained distributional patterns across diverse traffic scenarios. This framework offers substantial benefits in terms of interpretability, flexibility, and data efficiency, which makes it highly suitable for practical applications in network simulation, anomaly detection, and optimization. To further enhance its applicability to various mobile network scenarios, future research should aim to integrate additional contextual and spatial variables into the MEP framework.

CONCLUSION AND RECOMMENDATIONS

The evolution of fifth-generation (5G) mobile networks has introduced unprecedented opportunities for high-throughput communications, massive device connectivity, and support for advanced applications such as autonomous systems, real-time analytics, and smart city services. However, these advances come with significant challenges, particularly in modeling, forecasting, and optimizing network traffic, which is inherently dynamic, heterogeneous, and highly complex. Machine learning (ML) has emerged as a powerful tool for addressing these challenges, but its success depends on the availability of large-scale, representative, and high-quality datasets. Unfortunately, publicly available 5G datasets remain scarce due to privacy concerns, proprietary restrictions, and infrastructure limitations.

To overcome this limitation, this thesis leveraged urban mobility datasets as a proxy for 5G traffic, motivated by their similarity in temporal patterns, statistical properties, and traffic dynamics. However, these datasets often contain missing observations caused by sensor failures, device malfunctions, and connectivity issues, which significantly degrade the performance of downstream forecasting and optimization models. To address this, an adaptive imputation approach was proposed, specifically designed to fill incomplete time series while preserving temporal dependencies and seasonal structures.

Furthermore, evaluating the quality of imputation presents a critical challenge when ground truth values are unavailable. To tackle this issue, The thesis introduced novel distribution-based evaluation metrics that serve as an alternative to traditional metrics, enabling more robust and informative evaluation of filled data. Comprehensive experiments conducted on real-world urban datasets demonstrated that these metrics are consistent, interpretable, and effective across different gap sizes and model families.

Building upon the reconstructed datasets, the thesis proposed a Maximum Entropy Principle (MEP)-based traffic generation framework. By encoding empirically observed constraints, this

approach produces realistic synthetic 5G traffic traces that accurately capture both temporal dynamics and distributional properties without relying on unsupported assumptions. Extensive experiments demonstrated that the MEP-based approach is data-efficient, interpretable, and capable of generating synthetic 5G traffic that accurately preserves the temporal dynamics and distributional properties observed in real-world data.

The contributions of this thesis collectively enhance the robustness and reliability of data-driven 5G traffic analysis. By addressing the challenges of data scarcity, missing information, evaluation reliability, and synthetic data generation, this work provides a complete framework that supports improved traffic forecasting, anomaly detection, and network performance evaluation in next-generation mobile networks.

Looking forward, several research directions remain open. First, extending the proposed imputation framework to handle multi-modal and multi-source datasets that combine traffic, mobility, and contextual information could further improve reconstruction accuracy and robustness. Second, enhancing the traffic generation framework by incorporating adaptive mechanisms capable of capturing evolving network behaviors would make it more suitable for real-time applications in operational 5G environments. Finally, as sixth-generation (6G) networks emerge, future research could focus on adapting the proposed methodologies to model new traffic characteristics introduced by ultra-reliable low-latency communications, integrated sensing, and AI-driven network services.

In summary, this thesis bridges critical gaps in 5G traffic modeling by improving data preparation, introducing robust evaluation methodologies, and enabling realistic traffic synthesis. The proposed frameworks lay the groundwork for more reliable machine learning models and provide a foundation for advancing data-driven solutions in next-generation wireless networks.

BIBLIOGRAPHY

Front Matter. *Statistical Analysis with Missing Data, 3rd Ed.* (pp. i–xii). John Wiley & Sons, Ltd. Retrieved from: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119482260.fmatter.

Al-Douri, Yamur K., Hamodi, Hussan & Lundberg, Jan. (2018). Time Series Forecasting Using a Two-Level Multi-Objective Genetic Algorithm: A Case Study of Maintenance Cost Data for Tunnel Fans. *Algorithms*, 11(8). doi: 10.3390/a11080123.

Bandara, Kasun, Bergmeir, Christoph & Hewamalage, Hansika. (2021). LSTM-MSNet: Leveraging Forecasts on Sets of Related Time Series With Multiple Seasonal Patterns. *IEEE Transactions on Neural Networks and Learning Systems*, 32(4), 1586–1599. Retrieved from: http://dx.doi.org/10.1109/TNNLS.2020.2985720.

Barb, Gordana & Otesteanu, Marius. (2020). 4G/5G: A Comparative Study and Overview on What to Expect from 5G. *Proc. 43rd Int. Conf. Telecommun. Signal Process. (TSP)*, pp. 37-40. doi: 10.1109/TSP49548.2020.9163402.

Biasio, A. De, Chiariotti, Federico, Polese, Michele, Zanella, Andrea & Zorzi, Michele. (2019). A QUIC implementation for Ns-3. *Proc. 2019 Workshop Ns-3*, pp. 1–8.

Bidyuk, Peter, Kalinina, Irina & Gozhyj, Aleksandr. (2022). An Approach to Identifying and Filling Data Gaps in Machine Learning Procedures. In Babichev, Sergii & Lytvynenko, Volodymyr (Eds.), *Lecture Notes in Computational Intelligence and Decision Making* (pp. 164–176). Springer International Publishing.

Borgeaud, A. (2023). *5G deployment by country in Latin America 2023* (Report n°2). Germany.

Bothe, S., Qureshi, H. N. & Imran, A. (2019). Which statistical distribution best characterizes modern cellular traffic and what factors could predict its spatiotemporal variability? *IEEE Communications Letters*, 23(5), 810–813.

Brockwell, N. P. J. & Davis, R. A. (2018). ARMA Model Identification. In *Linear Models and Time-Series Analysis* (ch. 9, pp. 405–442). John Wiley & Sons, Ltd. doi: https://doi.org/10.1002/9781119432036.ch9.

Brockwell, P. J. & Davis, R. A. (1998). Biometrics. *Biometrics*, 54(3), 1204–1204. Retrieved from: http://www.jstor.org/stable/2533882.

Brophy, Eoin, Wang, Zhengwei, She, Qi & Ward, Tomás. (2023). Generative Adversarial Networks in Time Series: A Systematic Literature Review. *ACM Computing Surveys*, 55(10), 1 – 31.

Cardoso, A. A. & Vieira, F. H. T. (2019a). Generation of synthetic network traffic series using a transformed autoregressive model based adaptive algorithm. *IEEE Latin America Transactions*, 17(08), 1268–1275.

Cardoso, Alisson Assis & Vieira, Flávio Henrique Teles. (2019b). Generation of Synthetic Network Traffic Series Using a Transformed Autoregressive Model Based Adaptive Algorithm. *IEEE Latin America Transactions*, 17(08), 1268–1275.

Che, Zhengping, Purushotham, Sanjay, Cho, Kyunghyun, Sontag, David & Liu, Yan. (2018). Recurrent Neural Networks for Multivariate Time Series with Missing Values. *Scientific Reports*, 8(1), 6085.

Chen, Kai, Wang, Bowen, Wang, Wei, Li, Xuan & Ren, Fan. (2023). DeeProphet: Improving HTTP adaptive streaming for low latency live video by meticulous bandwidth prediction. *Proc. ACM Web Conf.*, pp. 2991–3001.

Chen, P., Niu, A., Liu, D., Jiang, W. & Ma, B. (2018, July). Time Series Forecasting of Temperatures using SARIMA: An Example from Nanjing. *IOP Conf. Ser.: Mater. Sci. Eng.*, 394(5), 052024.

Chen, Shuo, Lin, Rongheng & Zeng, Wei. (2022). Short-Term Load Forecasting Method Based on ARIMA and LSTM. *Proc. 2022 IEEE 22nd Int. Conf. Commun. Technol. (ICCT)*, pp. 1913–1917. doi: 10.1109/ICCT56141.2022.10073051.

Chen, Tianqi & Guestrin, Carlos. (2016). XGBoost: A Scalable Tree Boosting System. pp. 785 – 794. Retrieved from: https://doi.org/10.1145/2939672.2939785.

Cheng, Adriel. (2019). PAC-GAN: Packet Generation of Network Traffic using Generative Adversarial Networks. *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 0728-0734.

Cherif, Iyad Lahsen & Kortebi, Abdesselem. (2019). On Using eXtreme Gradient Boosting (XGBoost) Machine Learning Algorithm for Home Network Traffic Classification. *Proc. 2019 Wireless Days (WD)*, pp. 1–6. doi: 10.1109/WD.2019.8734193.

Chintalapudi, Nalini, Battineni, Gopi & Amenta, Francesco. (2020). COVID-19 Virus Outbreak Forecasting of Registered and Recovered Cases after Sixty Day Lockdown in Italy: A Data Driven Model Approach. *J. Microbiol., Immunol. Infect.*, 53(3), 396–403. Retrieved from: https://www.sciencedirect.com/science/article/pii/S1684118220300980.

Choi, Yong-Hoon, Kim, Daegyeom, Ko, Myeongjin, Cheon, Kyung-yul, Park, Seungkeun, Kim, Yunbae & Yoon, Hyungoo. (2023). ML-Based 5G Traffic Generation for Practical Simulations Using Open Datasets. *IEEE Communications Magazine*, 61(9), 130-136.

Corcoran, Diarmuid, Kreuger, Per & Schulte, Christian. (2020). Efficient Real-Time Traffic Generation for 5G RAN. 1-9.

Cédric Villani. (2009). *Optimal Transport: Old and New*. Berlin: Springer.

Dai, Y.-H. (2002). Convergence Properties of the BFGS Algorithm. *SIAM Journal on Optimization*, 13, 693–701.

Dalgkitsis, A., Louta, M. & Karetsos, G. T. (2018a). Traffic forecasting in cellular networks using the LSTM RNN. *Proc. 22nd Pan-Hellenic Conf. Inform.*, pp. 28–33.

Dalgkitsis, Anestis, Louta, Malamati & Karetsos, George T. (2018b). Traffic Forecasting in Cellular Networks Using the LSTM RNN. *Proc. 22nd Pan-Hellenic Conf. Inform.*, pp. 28–33.

Ding, Zhiguo, Schober, Robert & Poor, H. Vincent. (2021). No-pain no-gain: DRL assisted optimization in energy-constrained CR-NOMA networks. *IEEE Transactions on Communications*, 69(9), 5917–5932.

Donders, Arno R. T., van der Heijden, Geert J. M. G., Stijnen, Theo & Moons, Karel G. M. (2006). Review: A Gentle Introduction to Imputation of Missing Values. *Journal of Clinical Epidemiology*, 59(10), 1087–1091. doi: https://doi.org/10.1016/j.jclinepi.2006.01.014.

Eigenschink, Peter, Reutterer, Thomas, Vamosi, Stefan, Vamosi, Ralf, Sun, Chang & Kalcher, Klaudius. (2023). Deep Generative Models for Synthetic Data: A Survey. *IEEE Access*, 11, 47304-47320. doi: 10.1109/ACCESS.2023.3275134.

Eze, Nnaemeka Martin, Asogwa, Oluchukwu Chukwuemeka, Obetta, A. O., Ojide, Kingsley Chiedozie & Okonkwo, Clement Ikechukwu. (2020). A Time Series Analysis of Federal Budgetary Allocations to Education Sector in Nigeria (1970-2018). *American Journal of Applied Mathematics and Statistics*, 8(1), 1–8.

Fang, Y., Ergüt, S. & Patras, P. (2022). SDGNet: A handover-aware spatiotemporal graph neural network for mobile traffic forecasting. *IEEE Communications Letters*, 26(3), 582–586.

Fourati, Hasna, Maaloul, Rihab & Chaari, Lamia. (2021). A survey of 5G network systems: challenges and machine learning approaches. *International Journal of Machine Learning and Cybernetics*, 12(2), 385–431. doi: 10.1007/s13042-020-01178-4.

Garza, Azul, Challu, Cristian & Mergenthaler-Canseco, Max. (2024). TimeGPT-1. Retrieved from: https://arxiv.org/abs/2310.03589.

Gong, X., Jia, L. & Li, N. (2022). Research on mobile traffic data augmentation methods based on SA-ACGAN-GN. *Mathematical Biosciences and Engineering*, 19(11), 11512–11532.

Gómez-Ríos, A., Luengo, J. & Herrera, F. (2017, 06). A Study on the Noise Label Influence in Boosting Algorithms: AdaBoost, GBM and XGBoost. *Proc. 2017 14th Int. Conf. Softw. Eng. Adv.*, pp. 268–280. doi: 10.1007/978-3-319-59650-1_23.

Hanbanchong, Aphichit & Piromsopa, Krerk. (2012). SARIMA Based Network Bandwidth Anomaly Detection. *Proc. 2012 9th Int. Conf. Comput. Sci. Softw. Eng. (JCSSE)*, pp. 104–108. doi: 10.1109/JCSSE.2012.6261934.

He, K., Chen, X., Wu, Q., Yu, S. & Zhou, Z. (2022). Graph attention spatial-temporal network with collaborative global-local learning for citywide mobile traffic prediction. *IEEE Transactions on Mobile Computing*, 21(4), 1244–1256.

Hochreiter, Sepp & Schmidhuber, Jürgen. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735-1780.

Hui, S. (2023). Large-scale urban cellular traffic generation via knowledge-enhanced GANs with multi-periodic patterns. *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, pp. 4195–4206.

Hui, Shuodi, Wang, Huandong, Wang, Zhenhua, Yang, Xinghao, Liu, Zhongjin, Jin, Depeng & Li, Yong. (2022). Knowledge enhanced GAN for IoT traffic generation. *Proc. ACM Web Conf.*, pp. 3336–3346.

Hui, Shuodi, Wang, Huandong, Li, Tong, Yang, Xinghao, Wang, Xing, Feng, Junlan, Zhu, Lin, Deng, Chao, Hui, Pan, Jin, Depeng & Li, Yong. (2023). Large-scale urban cellular traffic generation via knowledge-enhanced GANs with multi-periodic patterns. *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, pp. 4195–4206.

Jaumard, Brigitte & Ziazet, Junior Momo. (2023). 5G E2E Network Slicing Predictable Traffic Generator. *2023 19th Int. Conf. Network Service Manag. (CNSM)*, pp. 1–7. doi: 10.23919/CNSM59352.2023.10327908.

Jaynes, Edwin Thompson. (1957a). Information Theory and Statistical Mechanics. *Physical Review*, 106, 620–630. doi: 10.1103/PhysRev.106.620.

Jaynes, Edwin Thompson. (1957b). Information Theory and Statistical Mechanics. II. *Physical Review*, 108, 171–190. doi: 10.1103/PhysRev.108.171.

Jin, Guangyin, Liang, Yuxuan, Fang, Yuchen, Shao, Zezhi, Huang, Jincai, Zhang, Junbo & Zheng, Yu. (2024). Spatio-Temporal Graph Neural Networks for Predictive Learning in Urban Computing: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 36(10), 5388–5408.

Jin, Junchen, Rong, Dingding, Zhang, Tong, Ji, Qingyuan, Guo, Haifeng, Lv, Yisheng, Ma, Xiaoliang & Wang, Fei-Yue. (2022). A GAN-based short-term link traffic prediction approach for urban road networks under a parallel learning framework. *IEEE Transactions on Intelligent Transportation Systems*, 23(9), 16185–16196.

Junninen, Heikki, Niska, Heikki, Tuppurainen, Kari, Ruuskanen, Jarkko & Kolehmainen, Mika. (2004). Methods for Imputation of Missing Values in Air Quality Data Sets. *Atmospheric Environment*, 38(18), 2895–2907. doi: https://doi.org/10.1016/j.atmosenv.2004.02.026.

Karmeshu & Pal, N. R. (2003). Uncertainty, Entropy and Maximum Entropy Principle — An Overview. In Karmeshu (Ed.), *Entropy Measures, Maximum Entropy Principle and Emerging Applications* (pp. 1–53). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-540-36212-8_1.

Khattab, Abd Alhamid Rabia, Elshennawy, Nada Mohamed & Fahmy, Mahmoud. (2023). GMA: Gap Imputing Algorithm for time series missing values. *Journal of Electrical Systems and Information Technology*, 10(1), 41. Retrieved from: https://doi.org/10.1186/s43067-023-00094-1.

Khayati, Mourad, Lerner, Alberto, Tymchenko, Zakhar & Cudré-Mauroux, Philippe. (2020). Mind the gap: an experimental evaluation of imputation of missing values techniques in time series. *Proc. VLDB Endow.*, 13(5), 768–782. Retrieved from: https://doi.org/10.14778/3377369.3377383.

Kholgh, Danial Khosh & Kostakos, Panos. (2023). PAC-GPT: A Novel Approach to Generating Synthetic Network Traffic With GPT-3. *IEEE Access*, 11, 114936-114951.

Kim, Daegyeom, Ko, Myeongjin, Kim, Sunghyun, Moon, Sungwoo, Cheon, Kyung-Yul, Park, Seungkeun, Kim, Yunbae, Yoon, Hyungoo & Choi, Yong-Hoon. (2022). Design and Implementation of Traffic Generation Model and Spectrum Requirement Calculator for Private 5G Network. *IEEE Access*, 10, 15978-15993.

Kingma, Diederik P. & Ba, Jimmy. (2017). Adam: A Method for Stochastic Optimization. Retrieved from: https://arxiv.org/abs/1412.6980.

Kostas Tzoumpas,Aaron Estrada, Pietro Miraglio & Zambelli, Pietro. (2022). A Data Filling Methodology for Time Series Based on CNN and (Bi)LSTM Neural Networks. *IEEE Commun. Mag.*, 61(6), 86–92.

Kouvatsos, Demetres. (2003). A Universal Maximum Entropy Solution for Complex Queueing Systems and Networks. In Karmeshu (Ed.), *Entropy Measures, Maximum Entropy Principle and Emerging Applications* (pp. 137–162). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-540-36212-8_7.

Kurri, Varun, Raja, Vishweshvaran & Prakasam, P. (2021). Cellular Traffic Prediction on Blockchain-Based Mobile Networks Using LSTM Model in 4G LTE Network. *Peer-to-Peer Networking and Applications*, 14(3), 1088–1105. doi: 10.1007/s12083-021-01085-7.

Li, He, Jin, Duo, Li, Xuejiao, Huang, Jianbin, Ma, Xiaoke, Cui, Jiangtao, Huang, Deshuang, Qiao, Shaojie & Yoo, Jaesoo. (2023a). DMGF-Net: An efficient dynamic multi-graph fusion network for traffic prediction. *ACM Transactions on Knowledge Discovery from Data*, 17(7), 1–19.

Li, Qing, Xiong, Diwen & Shang, Mingsheng. (2022a). Adjusted stochastic gradient descent for latent factor analysis. *Information Sciences*, 588, 196–213. Retrieved from: https://www.sciencedirect.com/science/article/pii/S0020025521012871.

Li, Tao, Braud, Tristan, Li, Yong & Hui, Pan. (2021). Lifecycle-aware online video caching. *IEEE Transactions on Mobile Computing*, 20(8), 2624–2636.

Li, Tong, Xia, Tong, Wang, Huandong, Tu, Zhen, Tarkoma, Sasu, Han, Zhu & Hui, Pan. (2022b). Smartphone app usage analysis: Datasets, methods, and applications. *IEEE Communications Surveys & Tutorials*, 24(2), 937–966. Second Quarter.

Li, Wei, Zheng, Weike, Xiao, Xi & Wang, Shoujin. (2023b). STAN: Stage-adaptive network for multi-task recommendation by learning user lifecycle-based representation. *Proc. 17th ACM Conf. Recommender Syst.*, pp. 602–612.

Li, Yun, Yu, DAZHOU, Liu, ZHENKE, Zhang, MINXING, Gong, XIAOYUN & Zhao, LIANG. (2023c). Graph Neural Network for Spatiotemporal Data: Methods and Applications.

Liao, Wenlong, Wang, Shouxiang, Yang, Dechang, Yang, Zhe, Fang, Jiannong, Rehtanz, Christian & Porté-Agel, Fernando. (2025). TimeGPT in Load Forecasting: A Large Time Series Model Perspective. *Applied Energy*, 379, 124973. doi: 10.1016/j.apenergy.2024.124973.

Lin, Z., Jain, A., Wang, C., Fanti, Giulia & Sekar, Vyas. (2020). Using GANs for sharing networked time series data: Challenges, initial promise, and open questions. *Proc. ACM Internet Meas. Conf.*, pp. 464–483.

Liu, Dong C. & Nocedal, Jorge. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45, 503–528.

Liu, Jingxuan, Zang, Haixiang, Cheng, Lilin, Ding, Tao, Wei, Zhinong & Sun, Guoqiang. (2023). A Transformer-Based Multimodal-Learning Framework Using Sky Images for Ultra-Short-Term Solar Irradiance Forecasting. *Applied Energy*, 342, 121160. doi: 10.1016/j.apenergy.2023.121160.

Lu, H., Zhang, Y., Li, Y., Jiang, Chunxiao & Abbas, H. (2021). User-oriented virtual mobile network resource management for vehicle communications. *IEEE Transactions on Intelligent Transportation Systems*, 22(6), 3521–3532.

Miao, X., Wu, Y., Chen, L., Gao, Y. & Yin, J. (2023). An Experimental Survey of Missing Data Imputation Algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 35(7), 6630–6650. doi: 10.1109/TKDE.2022.3186498.

Moritz, Steffen, Sardá-Espinosa, Alexey, Bartz-Beielstein, Thomas, Zaefferer, Martin & Stork, Johannes. (2015). Comparison of Different Methods for Univariate Time Series Imputation in R. Retrieved from: https://arxiv.org/abs/1510.03924.

Navarro-Ortiz, J., Romero-Diaz, P., Sendra, S., Ameigeiras, P., Ramos-Munoz, J. J. & Lopez-Soler, J. M. (2020). A Survey on 5G Usage Scenarios and Traffic Models. *IEEE Communications Surveys & Tutorials*, 22(2), 905–929. doi: 10.1109/COMST.2020.2971781. Secondquarter.

Park, Jangho, Müller, Juliane, Arora, Bhavna, Faybishenko, Boris, Pastorello, Gilberto, Varadharajan, Charuleka, Sahu, Reetik & Agarwal, Deborah. Long-term missing value imputation for time series data using deep neural networks. Retrieved from: https://doi.org/10.1007/s00521-022-08165-6.

Pratama, Irfan, Permanasari, Adhistya Erna, Ardiyanto, Igi & Indrayani, Rini. (2016). A review of missing values handling methods on time-series data. *2016 International Conference on Information Technology Systems and Innovation (ICITSI)*, pp. 1-6.

Rahman, S., Mun, H., Lee, H., Lee, Y., Tornatore, Massimo & Mukherjee, Biswanath. (2018). Insights from analysis of video streaming data to improve resource management. *IEEE 7th Int. Conf. Cloud Netw. (CloudNet)*, pp. 1–3.

Rebala, Gaurav, Ravi, Abhishek & Churiwala, Sanjay. (2019). *An Introduction to Machine Learning*. Springer Int. Publ. doi: 10.1007/978-3-030-15729-6.

Ring, Matthias, Schlör, Daniel, Landes, Daniel & Hotho, Andreas. (2019). Flow-based network traffic generation using generative adversarial networks. *Computers & Security*, 82, 156–172.

Ryabko, Daniil. (2019). *Asymptotic Nonparametric Statistical Analysis of Stationary Time Series*. Retrieved from: https://api.semanticscholar.org/CorpusID:71146257.

Saad, Muhammad, Chaudhary, Mohita, Karray, Fakhri & Gaudet, Vincent. (2020). Machine Learning Based Approaches for Imputation in Time Series Data and their Impact on Forecasting. *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 2621-2627.

Sarafanov, Mikhail, Nikitin, Nikolay O. & Kalyuzhnaya, Anna V. (2022). Automated data-driven approach for gap filling in the time series using evolutionary learning. In *16th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2021)* (pp. 633–642). Springer International Publishing.

Sarkar, Sutapa & Debnath, Aritri. (2021). Machine Learning for 5G and Beyond: Applications and Future Directions. *2021 Second Int. Conf. Electronics Sustainable Commun. Systems (ICESC)*, pp. 1688–1693. doi: 10.1109/ICESC51422.2021.9532728.

Soheil Kolouri,Se Rim Park, Matthew Thorpe Dejan Slepčev & Rohde, Gustavo K. (2017). Optimal Mass Transport: Signal Processing and Machine-Learning Applications. *IEEE Signal Process. Mag.*, 34(4), 43–59.

Song, Hayeong & Szafir, Danielle Albers. (2019). Where's My Data? Evaluating Visualizations with Missing Data. *IEEE Transactions on Visualization and Computer Graphics*, 25(1), 914-924.

Stenger, Michael, Leppich, Robert, Foster, Ian, Kounev, Samuel & Bauer, André. (2024). Evaluation is key: a survey on evaluation measures for synthetic time series. *Journal of Big Data*, 11(1), 66. doi: 10.1186/s40537-024-00924-7.

Sun, C., Xu, K., Fiore, M., Marina, M. K., Wang, Y. & Ziemlicki, C. (2022). AppShot: A conditional deep generative model for synthesizing service-level mobile traffic snapshots at city scale. *IEEE Transactions on Network and Service Management*, 19(4), 4136–4150.

Sutiene, K., Vilutis, G. & Sandonavicius, D. (2011). Forecasting of GRID Job Waiting Time from Imputed Time Series. *Electron. and Electr. Eng.*, 114. doi: 10.5755/j01.eee.114.8.706.

Tarng, P.-Y., Chen, K.-T. & Huang, Polly. (2008). An analysis of WoW players' game hours. *7th ACM SIGCOMM Workshop on Network and System Support for Games, NETGAMES*, pp. 1–7.

Tarsitano, Agostino & Falcone, Marianna. (2011). Missing-Values Adjustment for Mixed-Type Data. *Journal of Probability and Statistics*, 2011, 290380.

Trendowicz, Adam & Jeffery, Ross. (2014). Classification and Regression Trees. *Software Project Effort Estimation*, 295–304. doi: 10.1007/978-3-319-03629-8_10.

Villani, Cédric. (2009). The Wasserstein distances. In *Optimal Transport: Old and New* (pp. 93–111). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-540-71050-9_6.

Wang, Xiaojie, Nie, Laisen, Ning, Zhaolong, Guo, Lei, Wang, Guoyin, Gao, Xinbo & Kumar, Neeraj. (2022). Deep learning-based network traffic prediction for secure backbone networks in Internet of Vehicles. *ACM Transactions on Internet Technology*, 22(4), 1–20.

Wang, Z., Hu, J., Min, G., Zhao, Z. & Wang, J. (2021). Data-augmentation-based cellular traffic prediction in edge-computing-enabled smart city. *IEEE Transactions on Industrial Informatics*, 17(6), 4179–4187.

Wellenzohn, Kevin, Böhlen, Michael Hanspeter, Dignös, Anton, Gamper, Johann & Mitterer, Hannes. (2017, March). Continuous Imputation of Missing Values in Streams of Pattern-Determining Time Series. *International Conference on Extending Database Technology (EDBT)*, pp. 1 – 12.

Wijesekara, Lakmini & Liyanage, Liwan. (2021). Air quality data pre-processing: A novel algorithm to impute missing values in univariate time series. *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 996-1001.

Wijesekara, Lakmini & Liyanage, Liwan. (2023). Mind the Large Gap: Novel Algorithm Using Seasonal Decomposition and Elastic Net Regression to Impute Large Intervals of Missing Data in Air Quality Data. *Atmosphere*, 14(2), 355.

Wu, Q., He, K., Chen, X., Yu, S. & Zhang, J. (2022). Deep transfer learning across cities for mobile traffic prediction. *IEEE/ACM Transactions on Networking*, 30(3), 1255–1267.

Xiang, Jinyong, Qiu, Zhifeng, Hao, Qihan & Cao, Huhui. (2020). Multi-Time Scale Wind Speed Prediction Based on WT-bi-LSTM. *MATEC Web Conf.*, 309, 05011. Retrieved from: https://doi.org/10.1051/matecconf/202030905011.

Xu, Dongpo, Zhang, Shuai, Zhang, Hai & Mandic, Danilo P. (2021a). Convergence of the RMSProp deep learning method with penalty for nonconvex optimization. *Neural Networks*, 139, 17–23.

Xu, K., Singh, R., Bilen, H., Fiore, M., Marina, M. K. & Wang, Y. (2022). CartaGenie: Context-driven synthesis of city-scale mobile network traffic snapshots. *Proc. 2022 IEEE Int. Conf. Pervasive Comput. Commun.*, pp. 119–129.

Xu, Kai, Singh, Rajkarn, Fiore, Marco, Marina, Mahesh K., Bilen, Hakan, Usama, Muhammad, Benn, Howard & Ziemlicki, Cezary. (2021b). SpectraGAN: Spectrum based generation of city scale spatiotemporal mobile network traffic data. *Proc. 17th Int. Conf. Emerg. Netw. Experiments Technol.*, pp. 243–258.

Xu, Qiantong, Huang, Gao, Yuan, Yang, Guo, Chuan, Sun, Yu, Wu, Felix & Weinberger, Kilian. (2018). An empirical study on evaluation metrics of generative adversarial networks. Retrieved from: https://arxiv.org/abs/1806.07755.

Xu, Yichao, Zhang, Kaisa, Chuai, Gang, Yang, Xu & Xiong, Zihao. (2023). Cellular Traffic Prediction Using Multivariate Time Series LSTM Based on Meteorological Data. *2023 Cross Strait Radio Science and Wireless Technology Conference (CSRSWTC)*, pp. 1-3.

Yang, Suorong, Guo, Suhan, Zhao, Jian & Shen, Furao. (2024). Investigating the Effectiveness of Data Augmentation from Similarity and Diversity: An Empirical Study. *Pattern Recognition*, 148, 110204.

Yao, Y., Gu, B., Su, Z. & Guizani, M. (2023). MVSTGN: A multi-view spatial-temporal graph network for cellular traffic prediction. *IEEE Transactions on Mobile Computing*, 22(5), 2837–2849.

Yin, Y., Lin, Z., Jin, M., Fanti, G. & Sekar, V. (2022). Practical GAN-based synthetic IP header trace generation using NetShare. *Proc. ACM SIGCOMM Conf.*, pp. 458–472.

Yossi Rubner, Carlo Tomasi & Guibas, Leonidas J. (2000). The Earth Mover's Distance as a Metric for Image Retrieval. *Int. J. Comput. Vis.*, 40(2), 99–121. Retrieved from: https://doi.org/10.1023/A:1026543900054.

Yu, Feng, Yu, Chenyu, Tian, Zhangyuan, Liu, Xiaoxiao, Cao, Jiacheng, Liu, Li, Du, Chenghu & Jiang, Minghua. (2024). Intelligent Wearable System With Motion and Emotion Recognition Based on Digital Twin Technology. *IEEE Internet of Things Journal*, 11(15), 26314–26328. doi: 10.1109/JIOT.2024.3394244.

Zhang, J., Tang, J., Zhang, X., Ouyang, W. & Wang, D. (2015). A survey of network traffic generation. *Proc. 3rd Int. Conf. Cyberspace Technol.*, pp. 1–6.

Zhang, J., Chen, Feng, Wang, Zijia & Liu, Hanxiao. (2019). Short-Term Origin-Destination Forecasting in Urban Rail Transit Based on Attraction Degree. *IEEE Access*, 7, 133452 – 133462.

Zhang, S. (2023). Deep transfer learning for city-scale cellular traffic generation through urban knowledge graph. *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, pp. 4842–4851.

Zhang, Shiyuan, Li, Tong, Hui, Shuodi, Li, Guangyu, Liang, Yanping, Yu, Li, Jin, Depeng & Li, Yong. (2023). Deep transfer learning for city-scale cellular traffic generation through urban knowledge graph. *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, pp. 4842–4851.

Zhang, Xinyu, Zhang, Yong, Wei, Xiulan, Hu, Yongli & Yin, Baocai. (2022). Traffic Forecasting with Missing Data via Low Rank Dynamic Mode Decomposition of Tensor. *IET Intell. Transp. Syst.*, 16, 1164–1176.

Zhu, Ciyou, Byrd, Richard H., Lu, Peihuang & Nocedal, Jorge. (1997). Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software*, 23(4), 550–560. doi: 10.1145/279232.279236.

Zhu, F., Liu, L. & Lin, T. (2020). An LSTM-based traffic prediction algorithm with attention mechanism for satellite network. *Proc. 3rd Int. Conf. Artif. Intell. Pattern Recognit.*, pp. 205–209.

Ziazet, J. M., Jaumard, Brigitte, Duong, H., Khoshabi, P. & Janulewicz, E. (2022). A Dynamic Traffic Generator for Elastic 5G Network Slicing. *Proc. IEEE Int. Symp. Meas. Netw. (M&N)*, pp. 1–6. doi: 10.1109/MN55117.2022.9887734.