Telecommunication Systems (2026) 89:59
https://doi.org/10.1007/s11235-026-01429-9

Data-driven single gateway location estimation using improved channel model and LSTM

AlaaAllah ElSabaa[1] · Florimond Guéniat[1] · Wenyan Wu[1]

Received: 20 May 2025 / Accepted: 20 February 2026

Abstract

The Internet of Things (IoT) has expanded rapidly, with outdoor applications relying on Low-Power Wide-Area Networks (LPWANs) to deliver long-range, low-power connectivity. Among these, LoRaWAN enables energy-constrained sensors to communicate across large areas using minimal infrastructure. However, outdoor localization with LoRaWAN remains challenging: RSSI-based methods suffer from fading and noise, while advanced techniques such as TDOA and AoA require multiple synchronized gateways or specialized antennas, increasing cost and complexity. In this work, we propose a hybrid framework that integrates an improved empirical path-loss model with machine learning for single-gateway localization. First, we refine the Birmingham path-loss model using clustering and distribution-aware regression, improving prediction accuracy by up to 28.6% compared to regression-only baselines. Second, we develop a data-driven LSTM network that leverages sequential RSSI and SNR traces, guided by the improved channel model, to predict the sector location of mobile nodes. We further benchmark our proposed LSTM model against a CNN–LSTM baseline, a well-established deep learning paradigm for time-series classification. Experimental results highlight the superiority of our approach, achieving 72% accuracy compared to 65% for the CNN–LSTM baseline. By avoiding multi-gateway synchronization and heavy infrastructure, our approach demonstrates a practical, low-cost solution for single-gateway localization in dense urban environments. Experimental results in Birmingham city center highlight the robustness and feasibility of this method, making it suitable for smart city deployments, industrial monitoring, and other resource-constrained IoT applications.

Keywords Path-loss model · LSTM · CNN-LSTM · Location estimation · IoT · LP-WAN · LoRa · Outdoor localization

1 Introduction

With the rapid technological advances of the Internet of Things (IoT), and the expectation of contributing up to $11 trillion to the global economy by 2025 [1]. One of the core enablers of this technological wave is the ability to extract, transmit, and process spatial and environmental data from edge devices. Such information supports the development of advanced positioning techniques and algorithms [2, 3]. A prominent communication framework supporting these innovations is the Low-Power Wide-Area Network (LPWAN), which is characterized by long-range transmission capabilities at low energy and data rate costs. These networks span a wide range of applications, including environmental monitoring [4], healthcare, smart cities [5], and industrial automation.

Among LPWAN technologies, LoRa (Long Range) has emerged as a key standard, using proprietary Chirp Spread Spectrum (CSS) modulation to transmit data efficiently over several kilometers with minimal power consumption [6, 7]. With supported data rates between 0.3 kbps and 50 kbps [8], LoRa has been successfully deployed in large-scale urban and rural settings [9–11], and is expected to become the backbone of future city-scale IoT deployments [12, 13].

Localization in LP-WANs has the potential to contribute to a more sustainable and eco-friendly living, firstly with their extensive coverage, they offer a robust foundation for positioning and tracking systems in smart city environments

F. Guéniat and W. Wu have contributed equally to this work.

✉ AlaaAllah ElSabaa
alaaallah.elsabaa@mail.bcu.ac.uk

Florimond Guéniat
florimond.gueniat@bcu.ac.uk

Wenyan Wu
wenyan.wu@bcu.ac.uk

[1] Department of Engineering, Birmingham City University, Birmingham, UK

Published online: 09 April 2026

[14]. Secondly, by conserving precious battery life, making devices last longer between charges or replacements but also through reducing costs associated with deploying and maintaining GPS-based systems specially through ability to simultaneously use its signals for communication and localization. While, GPS is said to be the most commonly used method for outdoor localization [15], since they present accuracy in the range between 1 to 10m in outdoor applications. However, their high power consumption, expensive technology and extra power and bandwidth need to transmit their data are considered to be drawbacks specially in the deployment of LP-WANs when cost, power demand and bandwidth are valuable [15], encouraging the interest in finding GPS-free solutions. GPS-free localization methods can be generally categorized into signal-based and learning-based localization techniques [3].

The most significant signal-based techniques being, Pow er-based such as, Received Signal Strength Indicator (RSSI), time-based such as Time of Arrival (ToA) and Time Difference of Arrival (TDoA), and angle-based such as Angle of Arrival (AoA) methods [16]. Every method has its own pros and cons, making them useful in different applications. In the realm of IoT localization, machine learning serves as a pivotal tool in enhancing the accuracy and reliability of position predictions [17]. ML algorithms can analyze wireless signal data to extract non-obvious patterns and correct for noise introduced by multipath propagation, interference, and obstructions [3, 17]. Among these, RSSI-based fingerprinting has gained traction, where signal features at known locations are stored in a database and matched during inference [11, 18]. However, fingerprinting suffers from limitations in dynamic or outdoor settings due to environmental changes that affect signal stability [19]. Integrating path-loss model into distance metric learning helps overcome this limitation. To overcome this, hybrid models have been proposed that combine physical signal models with learning algorithms. For instance, integrating a path-loss model into a distance-metric learning framework helps improve robustness in outdoor settings. Additionally, temporal sequence models such as Long Short-Term Memory (LSTM) networks have been used to exploit the time-series nature of signal data [20, 21]. LSTM networks capture both short- and long-term dependencies in RSSI or SNR sequences, enabling more accurate mobility-aware localization.

In addition to LSTM, convolutional neural networks (CNN) have also played a crucial role, especially when combined with LSTM in architectures designed for spatiotemporal signal feature extraction. CNN-LSTM models have demonstrated high localization accuracy in indoor settings using infrared sensor data [22] and BLE beacons [23]. However, these approaches often rely on multiple gateways or labeled datasets, which limits scalability in real-world urban deployments.

Despite these advancements, real-world studies examining LoRa propagation and ML-based localization in outdoor, foliage-heavy or semi-urban environments remain limited. To bridge this gap, [24] developed a vegetation-aware path-loss model for Tunisian oases, highlighting the role of tree density and trunk size in signal attenuation. Similarly, [25] proposed an uplink path-loss model for the Amazon region, showing improved accuracy over traditional models by incorporating vegetation effects and node mobility.

Although various studies have examined the use of ML within localization in IoT environments and the use of LSTM in time-series analysis, there is a lack of research in exploring the capabilities of LSTM in time-series data for location proximity and estimation, especially in outdoor environments.

Motivated by the simplicity and affordability of signal-based localization methods, and the ability of learning-based approaches to overcome their inherent limitations, this paper introduces a hybrid localization framework that combines machine learning with an empirically enhanced path-loss model. Rather than estimating precise geographic coordinates, our approach predicts the sector location of a mobile LoRa node using signal characteristics. By leveraging sequential RSSI, SNR, and timestamp data collected from a single gateway, the system integrates a data-driven LSTM network guided by a refined path-loss model to enable robust and efficient location proximity estimation.

This work provides the following key contributions:

1. Real-world deployment and dataset: We deployed a LoRaWAN system in Birmingham city center using a single gateway and a mobile node, collecting a diverse dataset across multiple campaigns and conditions.
2. Improved channel modeling: We refine the Birmingham path-loss model using regression, k-means, and distribution-based clustering, achieving statistically significant reductions in the mean absolute error compared to the Birmingham baseline.
3. Novel hybrid localization framework: We propose a location proximity prediction method that integrates knowledge-driven path-loss models with a data-driven LSTM, leveraging RSSI/SNR sequences and distance-aware transition matrices for improved temporal prediction.
4. Robustness and ablations: We systematically evaluate robustness under noise injection, ablations with and without synthetic data, and comparisons against baselines such as HMM and CNN-LSTM.
5. Practical implications: We demonstrate that meaningful accuracy gains (20% error reduction) can be achieved with a single gateway, avoiding the cost and complexity

of multi-gateway TDOA/AoA systems while enabling scalable smart city deployment.

The remainder of this paper is organized as follows. Section 2 reviews recent work on signal-based and learning-based localization techniques in LPWANs. Section 3 details the data collection process, comparative performance and path-loss model evaluation. Section 4 describes the proposed data-driven LSTM model, including training and validation. Section 5 presents experimental results, model comparisons,network evaluation, cross-validation analysis and robustness analysis. Section 6 discusses practical implications, and Section 7 concludes the paper with a summary and future directions.

2 Background study

Since the integration of IoT with industrial and non-industrial applications, its predicted that the IoT end-nodes reaches tens of billions in this decade. Location-aware IoT applications are at the forefront of Industry 4.0, which stresses the importance of automation and seamless data sharing in manufacturing processes. Smart factories use location data to optimize production lines, track inventory, and enhance worker safety. Asset tracking, inventory management, and real-time logistics benefit significantly from accurate location estimation [26]. Based on the received data, decisions and instructions are to be made.

In this section, we discuss research works, particularly related to location prediction applicable in IoT and LP-WANs environment.

2.1 Localization within LP-WANs

Fingerprinting is routenely used for localization. Fingerprint is the work of collecting data signals from the communication source being used, such as Wi-Fi, Bluetooth, RFID, etc. Then, database can be built to link the known locations to their specific signal characteristics (RSSI, ToA, etc.). For a device to be localized, it starts collecting data from these known sources and hence creating a 'fingerprint' of their current location, through comparing the fingerprint to the reference data in the database; the location can then be estimated [27]. However, the difficulty in characterizing the complex nonlinear relationship between signal vectors accurately. Machine learning and pattern recognition techniques are further developed to be incorporated [28]. As a result, ML techniques in various location tracking and estimation methods has been the current focus of many researches. Alternatively, localization methods based on angle and time have their own limitations such as, limited accuracy in non-line-of-sight scenarios and the need for precise time synchronization [29].

Hybrid solutions have been investigated to overcome the limitations of traditional localization methods. For example, in [30], the authors propose an outdoor fingerprinting scheme that combines Wi-Fi and OFDM signal values as input to an LSTM-based network.

LPWANs have recently gained wide popularity, primarily due to their low-power requirements and the resulting longer device lifetimes. Sensors in such networks can reliably capture environmental data and transmit them through different LPWAN technologies. In [31], RSSI fingerprinting was analyzed under both line-of-sight (LOS) and non-line-of-sight (NLOS) environments, highlighting the variability of performance across conditions. Similarly, [32] examined the Adaptive Data Rate (ADR) feature in LoRaWAN as an optimization method to enhance throughput, energy efficiency, and scalability. Their results show that machine learning algorithms can leverage RSSI values to estimate the number of obstacles affecting transmission.

Another notable work, LoRaLoc [33], leverages TDOA measurements generated by a LoRa network. GPS locations serve as ground truth, while machine learning models such as Random Forests and Neural Networks are trained on a reference map to perform localization. The evaluation in this study was based on simulated data.

In contrast to these prior approaches, which often rely on multiple technologies, external ground-truth data, or location history, our work investigates whether a single LoRa gateway and sequential packet transmissions can be used to predict device location. This approach reduces overhead and infrastructure requirements while still enabling accurate localization.

2.2 Path loss and propagation models

The loss of signal power during propagation can be influenced by various factors, including free space loss, diffraction, reflection, and absorption. Due to variations in urban structures, terrain profiles, and climate, path loss (PL) predictions can differ significantly across different propagation models. Accurate path loss estimation is essential for base station coverage areas and interference analysis [34].

The conventional path loss model is usually depicted by a log-distance power law, that related the PL to the distance covered by a signal logarithmically, and the distance's power is considered as the propagation exponent. To achieve the best fit of the collected data points with the PL model, it's necessary to identify the propagation exponent that minimizes the root mean square deviation of the data points from the model.

However, the path loss model is significantly influenced by propagation conditions and the range of operation. Tra-

ditional PL models cannot accurately estimate a highly variable ranging function. Therefore, we explored different regression and machine learning models to account for the variability in the ranging function. Additionally, we considered several reference path loss models, such as the Free Space Model, Okumura-Hata and Cost 231-Hata.

Recently, many researches conducted real LoRa experimentation such as Oulu, Finland [35] and Dortmund, Germany [36], where authors test different published path-loss models and check their capabilities against their collected signal measurements from a LoRa test deployment and further contribute to enhance their predictions. Given the characteristics of the city, including its geographical location and surrounding terrain, both studies developed their own tailored propagation models for the application of LoRaWAN in their respective areas and comparable environments.

The work in [37] included the comparison of different empirical models against their own collected measurements in search of the most adequate model to represent their data. The work done in this paper in finding a path loss model representing the data collected and its comparative performance to other models is discussed further in section 3.5 and was published in [38] for more detailed information.

2.3 LSTM RNNs overview

Long short-term memory (LSTM) are a type of neural network that is particularly well-suited for processing sequential data. In an LSTM, the input to the network's hidden layers consists of both the input features at the current timestep and the hidden activation from the previous timestep using a memory cell, that has selectively retained and updated information over the time steps. It allows to process information along a entire sequence. For this reasons, LSTMs capture long-range dependencies in data, crucial for tasks like time-series analysis in fields like predicting petroleum production, financial market prediction and more [39]. Furthermore, a multi-step LSTM network is highly efficient at automatically learning features from time series and sequence data [40].

LSTM has been successfully applied in various fields, as noted in [39]. LSTM models have enhanced Google's speech recognition, significantly improved machine translations on Google Translate, and refined responses from Amazon's Alexa. The study by [41] utilized LSTM for weather forecasting, highlighting the limitations of traditional prediction techniques and demonstrating how LSTM can overcome these by holding onto relevant information and discarding trivial data. The authors concluded that LSTM exhibits substantial performance in numerous real-world applications.

In [42], the authors proposed a hybrid CNN-LSTM model for time series data prediction to analyze and forecast air pollution. The paper outlines the shortcomings of traditional prediction models based on statistical methods and showcases the potential of deep learning to enhance predictive accuracy across various domains dealing with time series data. LSTM models are often preferred over transformer models due to their lower power consumption and faster inference times, as discussed in [43]. However, the article does not provide an in-depth analysis of the model's performance or a comparative performance with other prediction models.

There has been a shift in location prediction techniques as recent studies have explored the use of machine learning to better capture user behavior patterns. In [44], two sequence-based frameworks employing LSTM and fingerprinting for multi-point outdoor localization in wireless networks were introduced. They proposed using Bi-LSTM to improve localization accuracy through bringing prior and future information to achieve a lower parameters usage.

3 Methodology

In this section, we start by clarifying and formulating our problem mathematically. Then, we outline and examine the architecture of the proposed data-driven LSTM system as shown in Figure 1, with its three main components. Firstly, we present the LoRa node deployment and data collection system and how We analyze the signals obtained from the inertial sensors during the experiment to map the trajectories. Secondly, we explain in details the pre-processing, the data collected undergoes before developing the most-suited channel model and its comparative performance to previously well-established models. Lastly, we present our data-driven model with its layers and the concatenation with our presented model.

3.1 Problem formulation as a sector estimation task

We consider a LoRaWAN deployment consisting of a single gateway and a mobile transmitting node. The gateway receives uplink packets transmitted by the node over time. The objective of the proposed framework is to exploit temporal sequences of received packets, without introducing additional communication overhead, to infer the spatial location of the transmitting node relative to the gateway. For a given user u, the gateway observes a sequence of received packets over time. Each received packet is associated with a feature vector

$$\mathbf{x}_{u,t} = \left[\text{index}_{u,t},\ \text{time}_{u,t},\ \text{RSS}_{u,t},\ \text{SNR}_{u,t}\right]^{\top}, \tag{1}$$

where $\text{index}_{u,t}$ denotes the packet index used for temporal segmentation, $\text{time}_{u,t}$ is the reception timestamp, and $\text{RSS}_{u,t}$ and $\text{SNR}_{u,t}$ represent the received signal strength and signal-to-noise ratio, respectively. Over a temporal window

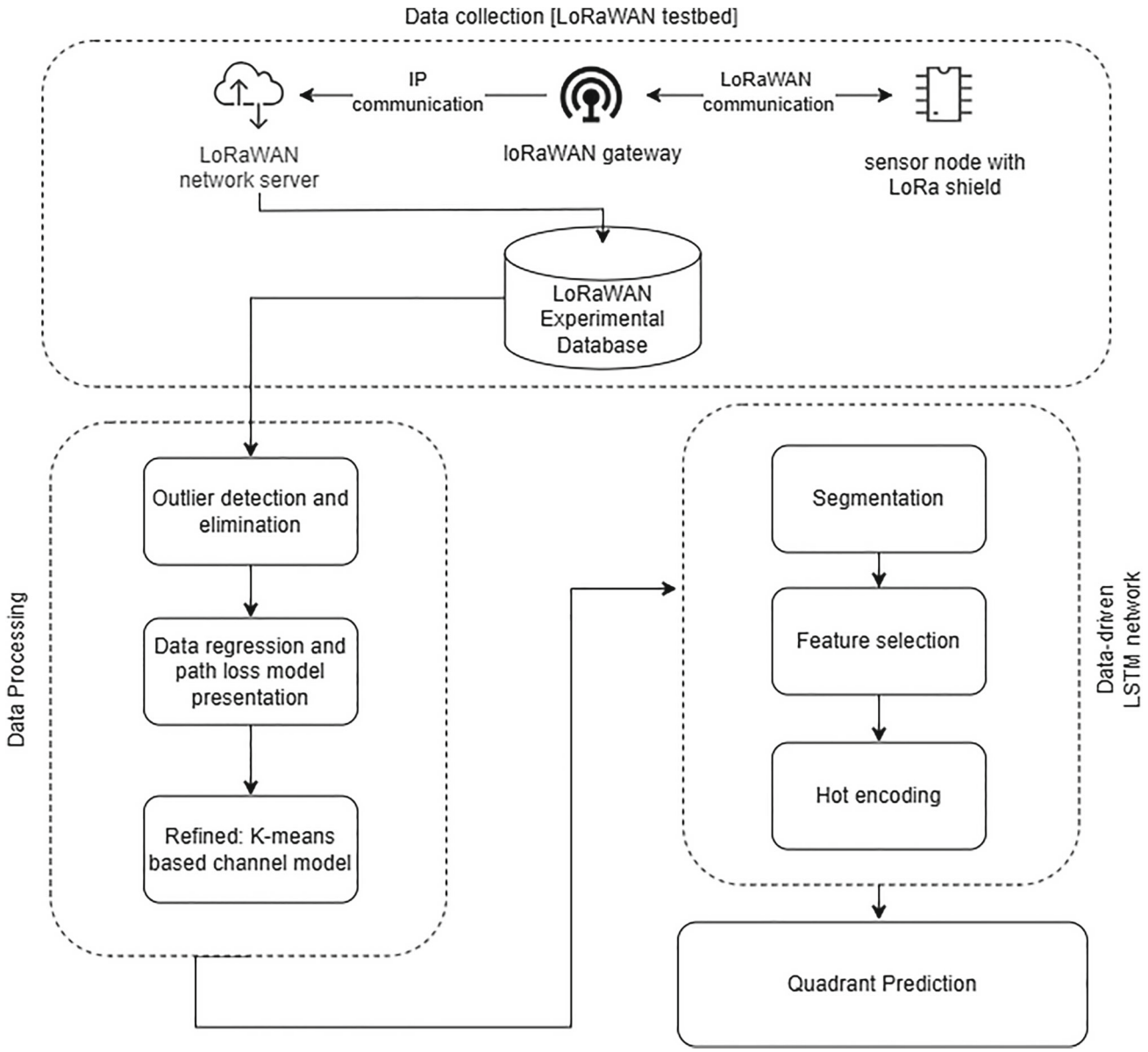


Fig. 1 Flowchart of the proposed location prediction framework

of length T, the observed signal sequence for user u is given by

$$\mathbf{X}_u = \{\mathbf{x}_{u,1}, \mathbf{x}_{u,2}, \ldots, \mathbf{x}_{u,T}\}. \tag{2}$$

The sequence length T may vary due to packet loss and asynchronous packet reception.

During data collection, the transmitting node is equipped with a GPS receiver, providing latitude and longitude measurements $(\text{Lat}_{u,t}, \text{Lon}_{u,t})$. These coordinates are used only during training and evaluation to derive ground-truth spatial labels and to estimate propagation-related quantities. To incorporate propagation-aware information, large-scale path-loss characteristics are estimated from the received signal measurements and used as intermediate, channel-aware features. These features capture distance-dependent attenuation trends and complement the raw RSS and SNR measurements when forming the input sequences to the temporal inference model.

Rather than estimating continuous coordinates directly, the coverage area of the gateway is partitioned into 3 angular sectors. Each sequence $\mathbf{X}_u$ is assigned a ground-truth sector label

$$s_u \in \{1, 2, 3\}, \tag{3}$$

derived from GPS measurements of the transmitting node and its azimuth angle relative to the gateway. Localization is

thus reformulated as a sector estimation problem, which is well-suited to single-gateway scenarios where fine-grained ranging is inherently ambiguous.

Given an observed sequence $\mathbf{X}_u$, the localization task is to estimate the sector s_u in which the user is located. This is formulated as a probabilistic estimation problem, where a parameterized model $f_\theta(\cdot)$ maps the input sequence to a posterior probability distribution over sectors,

$$\mathbf{p}_u = f_\theta(\mathbf{X}_u), \tag{4}$$

with $\mathbf{p}_u = [p_{u,1}, p_{u,2}, p_{u,3}]^\top$ and $\sum_{k=1}^{3} p_{u,k} = 1$. The estimated sector is obtained via a maximum a posteriori (MAP) decision,

$$\hat{s}_u = \arg \max_{k \in \{1,\ldots,3\}} p_{u,k}. \tag{5}$$

3.2 Optimization objective

The learning objective is to estimate the model parameters θ that minimize the empirical risk over the training dataset. In particular, we adopt the categorical cross-entropy loss, yielding the following optimization problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{u=1}^{N} \sum_{k=1}^{K} \left(-\mathbb{I}[s_u = k] \log \left(p_{u,k} \right) \right), \tag{6}$$

where $\mathbb{I}[\cdot]$ denotes the indicator function, K is the number of classes (sectors), and $p_{u,k}$ represents the predicted posterior probability that sequence X_u belongs to class k. Minimizing this objective corresponds to maximum likelihood estimation under a multinomial output distribution. This formulation encourages the temporal inference model to assign high posterior probability to the true sector label while exploiting temporal dependencies within the received signal sequences. During inference, sector estimation is performed via a maximum a posteriori (MAP) decision based on the learned posterior distribution, as defined in the previous subsection.

3.3 System model and data generation

The system considered in this work consists of a single LoRaWAN gateway receiving uplink transmissions from a mobile end device. Although measurements are collected from a single user, the dataset spans a wide range of spatial locations, distances, and angular sectors as the device moves within the gateway coverage area. Each received packet is characterized by RSSI and SNR measurements, which are inherently affected by distance-dependent path loss, shadowing, measurement noise, and packet loss typical of LPWAN deployments.

Rather than assuming a fixed analytical channel model, the proposed framework adopts a data-driven formulation in which channel variability and noise are handled empirically. Temporal packet sequences capture mobility-induced signal evolution, while robust path-loss regression and outlier handling mitigate large-scale variability. Additional robustness to realistic impairments is evaluated through controlled RSSI perturbations, packet-loss sensitivity analysis, and temporal ablation studies presented in the Results section.

Data diversity is therefore achieved through spatial mobility, temporal sampling, and sector-based partitioning, enabli ng the model to observe multiple channel realizations across time and space despite the use of a single mobile node. Details of the experimental setup, gateway configuration, and measurement environment follow the methodology described in our prior work [38], which provides a detailed characterization of the LoRaWAN testbed and data collection procedure.

3.4 Experimental setting & data acquisition

As depicted in the top block of the flow diagram in Figure 1, we start our process by setting up an LPWAN testbed at Birmingham City University (BCU), UK, utilizing LoRa technology. We employ an off-the-shelf LoRa node provided by The Things Network (TTN), specifically "The Things Uno" (illustrated in Figure 3a). The Things Uno node is built on the Arduino Leonardo and includes an added LoRaWAN module, Microchip (RN2483 - class A protocol stack) [45].

We enhance the node by integrating a GPS module for ground-truth measurements. The Adafruit 746 GPS module, which uses the MTK3339 GPS System on Chip (SoC), can track up to 22 satellites across 66 channels and has a sensitivity of -165 dBm. The LoRa node transmits data packets to the LoRaWAN gateway using LoRaWAN modulation operating at 868 MHz (within the ISM band in Europe) with a receiver sensitivity up to -148 dBm.

The measurement system includes a single LoRa MultiConnect Conduit IP67 Base Station (illustrated in Figure 3b). While the TTN node is mobile and can be easily carried around the campus vicinity, the LoRa MultiConnect Conduit is installed indoors on campus. Table 1 presents more information on the equipment used in our deployment.

The mobile node is encoded to transmit packets at a constant rate of a packet every 20 sec, which is further used to calculate the packet delivery ratio for every stop. The time spent at each stop is kept to 2 min. for equality in data collection. The experimentation stage included data collection from different paths as shown in Figure 2a, although the time spent at every stop was equal, still the map shows that for some locations, the reception failed. That might be due to

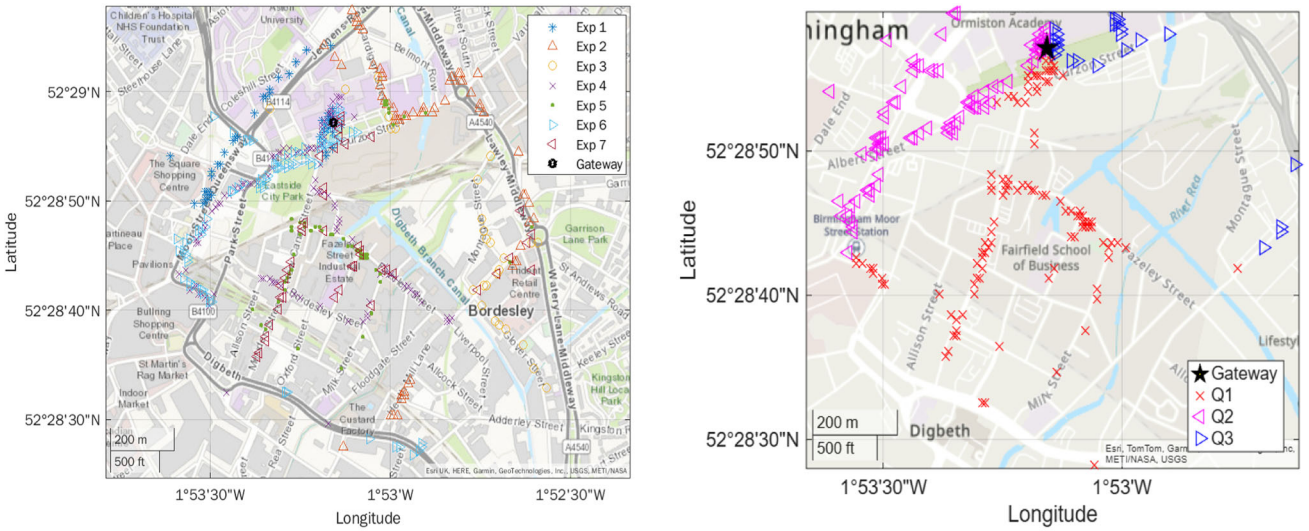


Fig. 2 Experimental Data collected. a): Gateway and all data points. b): Showing Sectors division Q1 in red 1181 points, Q2 in magneta 729 points and Q3 in blue 524 points

Table 1 Characteristics of MultiTech GW and The TTN

Characteristic	Gateway	Node
Module	MTCDIP-LEU1-267A-868	The Things Uno
LoRa Chip	-	Microchip RN2483 LoRa
Operating Frequency	867.9	867.9
Modulation	LoRa	LoRa
SF	12	12
Coding Rate	4/5	4/5
Rx Input Sensitivity (dB)	-	-148
Tx Power	14 dBm	-
Rx current consumption	-	14.2 mA

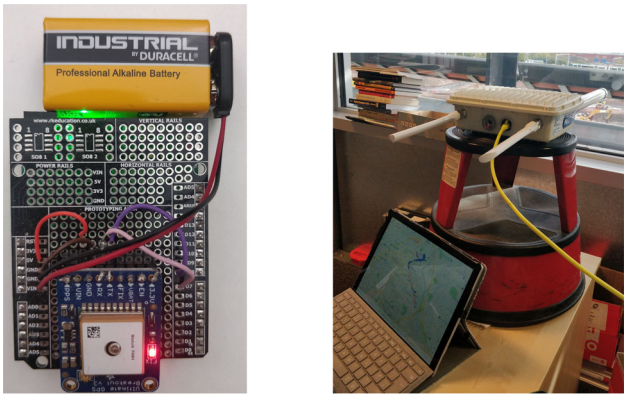
obstructions in the terrain or reflections from buildings in the surrounded area.

The data collected by the node, transmitted to the gateway is then saved on an online console through UDP/IP protocol. The LoRa data received and stored is of a sequential nature, that is limited by the data and coding rate chosen within the LoRAWAN settings along with the encoded chosen settings mentioned before. The database was further filtered from incomplete packages and duplicates before further processing. The filtered received data was then used to form a database in the form of ; index, time, latitude, longitude, RSS and SNR values.

The dataset inherently includes dynamic mobility, as the LoRa node was mobile during all campaigns. Measurements were collected along varying paths across different days and environmental conditions, ensuring that the training and evaluation data reflect realistic temporal and spatial variability. Multipath fading was naturally present due to the dense urban setting (city center with building reflections and obstructions). Additionally, augmentation via jitter/noise injection further emulates fading. Although explicit LOS/NLOS labeling was not performed, the Birmingham city-center dataset inherently includes numerous NLOS conditions due to dense urban morphology. Thus, the LSTM was trained and tested on a realistic mixture of LOS and NLOS links, supporting its applicability in obstructed environments.

3.5 Developed path loss model and comparative performance

As discussed earlier, it is important to have a robust estimation of the power loss against distance i.e path-loss for the area in concern. In our work, our data was experimentally collected in Birmingham,UK, with no empirical path loss



(a) (b)

Fig. 3 Measurement setup. a): TTN node. b): MultiConduit gateway

models specifically designed for the area to adopt. Although, similar LoRaWAN research was conducted in different cities, still to ensure precise localization, it is essential to develop an accurate path-loss model for the specific environment in advance [46].

Indeed, well-established models like Okumura Hata and Cost 231-Hata were initially tested but were not able to represent the propagation well in Birmingham (discussed in details in sec. 3.5). Therefore, we formulated a model for Birmingham and similar geographical urban areas.

The initial fit model of the collected data is determined using eq. 7 as reference, with the average PL:

$$PL_{avg} = B + 10n \log_{10}(d \times 10^{-3}), \tag{7}$$

where the distance d is given in m , B defines the PL intercept and n being the PL exponent.

We utilized a function based on minimizing the mean square error to formulate a suitable regression model. The cost function identified by $J(\zeta) := \sum_i ||RSSI_{model}(d_i, \zeta) - RSSI_i||^2$, where i corresponds to the i_{th} measurement, and $\zeta \in \mathbb{R}^2$ are the parameters of the path loss model: $\zeta := (B, n)$. The optimal parameters $\zeta^\star$ of the model are then obtained by minimization of J, such that $\zeta^\star = \underset{\zeta \in \mathbb{R}^2}{\operatorname{argmin}} J(\zeta)$, following the derivative-free Nelder-Mead method [47]. To further enhance the initial regression model, data outlier detection and elimination is considered.

In [48], the authors introduced a data outlier detection method utilizing the median absolute deviation (MAD) as a more robust and effective approach for measuring the dispersion of data points.

In statistical modeling, regression analysis is considered a collection of techniques used to evaluate the relationships between variables. It is essential to highlight that the path loss exponent (n) plays a significant role in the design of radio signal propagation. By applying linear regression analysis, (n) can be identified by minimizing the divergence between the measured values (Pm) and predicted values (Pr).

To compare the accuracy of the propagation models used in our study, we consider the following statistical parameters $\Delta_i = RSSI_{estimate} - RSSI_{measured}$, $|\Delta y| = \frac{1}{N}\sum_{i=1}^{N}|\Delta_{yi}|$ and $\sigma_e = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\Delta_i - |\Delta y|)^2}$. Where $|\Delta y|$ is the mean absolute error and the σ_e is the standard deviation of the estimated error. N is the total number of data samples collected and Δ_i represents the difference between the estimated and measured samples. Table 2 highlights the results. The model accurately estimated the RSSI values for distances up to 600m but showed deviations at greater distances. Meanwhile, both the OH and COST-231 models yielded the least precise estimations of the RSSI in Birmingham, with $|\Delta y|$ values of 13.52 and 13.34, indicating significant variations across both short and long distances.

Figure 4 shows a comparison of exponent (n) and intercepts (B) obtained in different environments. Although the Oulu and Dortmund models display similar initial PL intercept values, the proposed regression model for Birmingham has a higher PL intercept value. The intercept reflects the shadowing effect caused by power losses when signals propagate through obstacles. This difference could be attributed to BCU being surrounded by several tall and large buildings, while Oulu is situated near a waterfront.

It is worth noting that Dortmund benefited from a higher probability of line-of-sight (LOS) transmission, with the transmitting antenna positioned at a height of 30m, compared to 24m in Oulu. In our experiment, the antenna was fixated at an approximate height of 15m. In contrast, regarding the exponent (n), the Birmingham model exhibited the lowest exponent value among all. A lower exponent corresponds to a slower decrease in RSSI as the distance increases, resulting in a gentler slope. The lesser value of (n) reflects the open space surrounding BCU, where the gateway is located, highlighting the suitability of the location for enhanced coverage.

Our experiment revealed a loss of LoRa packets within the 1.1 to 1.3 km range at specific angles, despite the omnidirectional nature of the transmitting antenna as shown in Figure 5. This packet loss may indicate a high density of signal barriers, leading to attenuation of RSSI at the precise angles where measurements were taken. A constructive approach to address this would be to conduct additional measurements at the same distance while varying the deviations from the gateway.

Additionally, the higher power levels received at longer distances in Birmingham (approximately 1.4 km) compared to other models influenced the proposed channel model. This indicates a greater likelihood of line-of-sight (LOS) communication at extended ranges, paired with relatively higher RSSI values compared to shorter distances. This character-

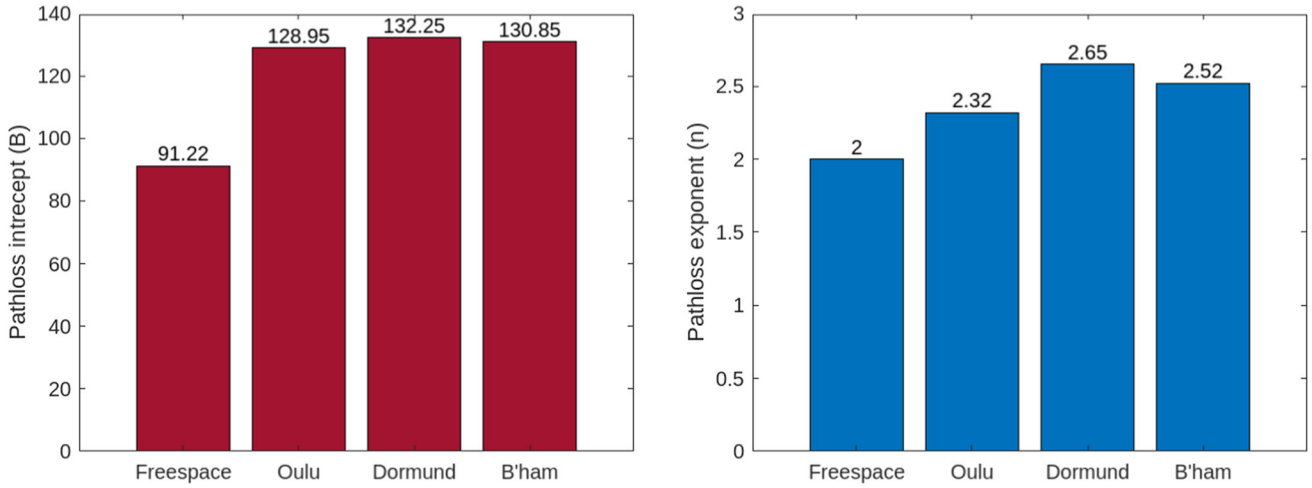


Fig. 4 Path loss exponent and intercept of different models. Left: pathloss intercept. Right: pathloss exponent

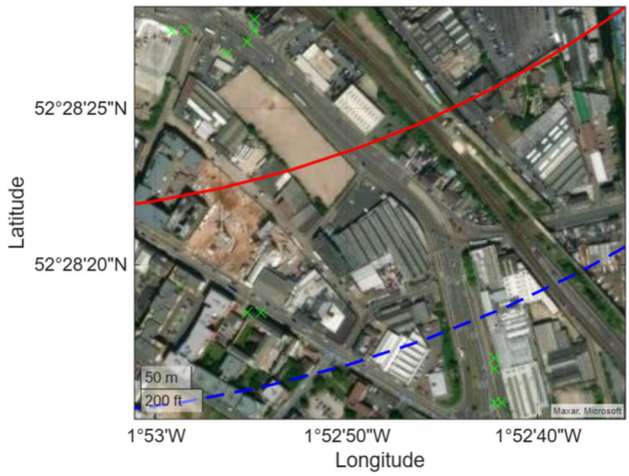


Fig. 5 Measurement results showing received packets (green markers) along the route. The red (1.1 km) and blue (1.3 km) arcs illustrate the radial distance from the gateway

istic distinguishes the Birmingham environment from other models.

Understanding the power-distance relationship within the topology of the area tested encouraged further consideration of dividing the map grid for better analysis. Dividing the map grid into uniform angular sectors as shown in Figure 2band studying the RSS and SNR variation as Figure 10 highlights.

3.6 K-means and distribution-based clustering

As the observations are affected by natural variables that can be stationary or random, it requires a large sample data to reveal their alternating regularity. Ensuring that the mean remains relatively constant over time or across different samples and that the variance does not change significantly across different subsets of data or over time is essential for reliable statistical modeling [49].

To achieve this, we employed two methods. The first involved collecting multiple measurements for each location during the same experiment and averaging the values to obtain a reliable estimate. The second method entailed gathering measurements from similar locations at different time intervals. Over a 15-month period, data from seven successful experiments were individually analyzed, as illustrated in Figure 2, with data points from each experiment represented in different colors. While each experiment followed a distinct path, there were overlapping points. The analysis incorporated an algorithm to process each data-location pair, calculating the haversine distance, which represents the distance between the GPS coordinates and the gateway location. Subsequently, the calculated distances were cross-referenced with the GPS coordinates to identify nearby locations, as the distance alone could correspond to any point along the circumference of a circle centered at the gateway. The algorithm identified nearby locations using a threshold of approximately 15m. This value was determined based on the accuracy of the GPS chip utilized in the experiments (3m) and the GIS system (11m).

Upon completing this analysis, we observed that the model deviated from the average RSSI values obtained during repeated experiments, even after accounting for measurement variance. Specifically, the model tended to overestimate the predicted received power at shorter distances and underestimate it at longer distances. The results led to doubting if a single global regression was the optimal method to accurately represent the refined data. As a result, we investigated the possibility of grouping data with shared characteristics or similarities into clusters. To begin, we applied the k-means algorithm to cluster the data using squared Euclidean distance and the sum of absolute differences, given that they aligned

Table 2 Statistical Performance Metrics

Error Parameters (dB)	Okumura Hata	Cost 231-Hata	Dortmund	Oulu	B'ham regression	K-mean B'ham	DBC B'ham
$\|\Delta y\|$	13.52	13.34	10.05	9.966	9.93	7.07	8.3
σ_e	17.43	17.48	22.88	22.57	22.63	11.76	14.58

well with our dataset characteristics. The data were normalized to enhance the efficiency of the clustering algorithm [50]. Subsequently, by defining three variables —distance, RSSI, and SNR— we utilized the k-means algorithm to divide the data into two clusters, as illustrated in Figure 7.

Algorithm 1 Distribution-Based Clustering (DBC)

1: Normalize features (distance, RSSI, SNR).
2: **for** $k = 2$ to 5 **do**
3: Apply k-means clustering with cityblock metrics.
4: Evaluate clusters using $|\Delta y|$.
5: **end for**
6: Select $k = 2$ (lowest error).
7: **for** each cluster **do**
8: Generate RSS histograms and compare empirical distributions.
9: Identify distributional shift from binomial-like to normal-like.
10: **end for**
11: Define cut-off at $\approx$ 200 m, where the distribution shifts.
12: Output final cluster assignments.

An iterative clustering procedure was carried out to identify the most suitable number of clusters for the collected data. Candidate solutions ranging from two to five clusters were evaluated using the cityblock distance metric, and performance was quantified in terms of the $|\Delta y|$. As illustrated in Figure 6, clustering solutions with 3, 4, and 5 clusters were examined, but the two-cluster configuration consistently produced the lowest error.

The Distribution-Based Clustering (DBC) algorithm, summarized in Algorithm 1, extends conventional k-means by incorporating the underlying distribution of the RSSI values. Whereas standard k-means identifies centroids purely from distance minimization, DBC explicitly leverages the statistical characteristics of the data. In our analysis, RSSI measurements within approximately $\approx$200m of the gateway followed a binomial-like distribution (Figure 8), while those at larger distances exhibited a normal-like distribution. This shift in distribution was used to define the cluster boundary.

To assess the clustering algorithms used in modeling and ensure a fair comparison, we calculate and compare the $|\Delta y|$. The clustering results show that both clustering approaches substantially reduce error compared to the Birmingham regression model. Specifically, the DBC method lowers the $|\Delta y|$ to 8.30 dB (16.16% improvement), while k-means further reduces it to 7.07 dB (28.6% improvement). Paired t-tests confirmed that both clustering approaches significantly outperformed the Birmingham baseline regression model, with K-means reducing mean absolute error by 2.01–3.76 dB ($t(746)=6.48$, $p < 0.0001$) and the DBC method reducing it by 0.96–2.36 dB ($t(746)=4.66$, $p < 0.0001$). Importantly, these improvements arise because clustering allows the data to be segmented into near-field and far-field regions, with separate regression models fitted to each. A single regression typically overestimates path loss at short distances and underestimates it at longer distances. By contrast, the hybrid piecewise approach produced by clustering corrects this bias: the near-field model captures the steep decay at short ranges, while the far-field model more accurately reflects the slower attenuation beyond several hundred meters. Both K-means and DBC outperform the traditional single-regression model; however, the choice of clustering method influences how well the transition boundary is detected, and thus may depend on the distances being considered.

4 Data analysis and experimentation

This section describes the analysis the data undergoes before its ready to pass through the LSTM model, as shown in the block diagram of Figure 9. Firstly, we explain the preprocessing steps performed on the collected data to remove outliers and reduce noise. Subsequently, data augmentation is performed to address the issue of an imbalanced dataset. This process involves oversampling the minority class to ensure its sample size matches that of the majority class. The resulting balanced dataset is then used to train and classify the proposed model. Then we discuss the stages through which the data pass through in our proposed model.

4.1 Data processing

As shown in the left block of our model in Figure 1, the data being fed to the LSTM passes through various submodels. The data collected is based on a LoRa device as sender and a single gateway receiver, the proposed system is divided into 2 main phases: the training phase, where real experimented LoRaWAN data, are collected and processed to construct a database based on timeseries RSS and SNR values of various locations before being fed into the LSTM model for the following testing stage. The testing stage includes evaluation and validation for optimal results. The training phase is

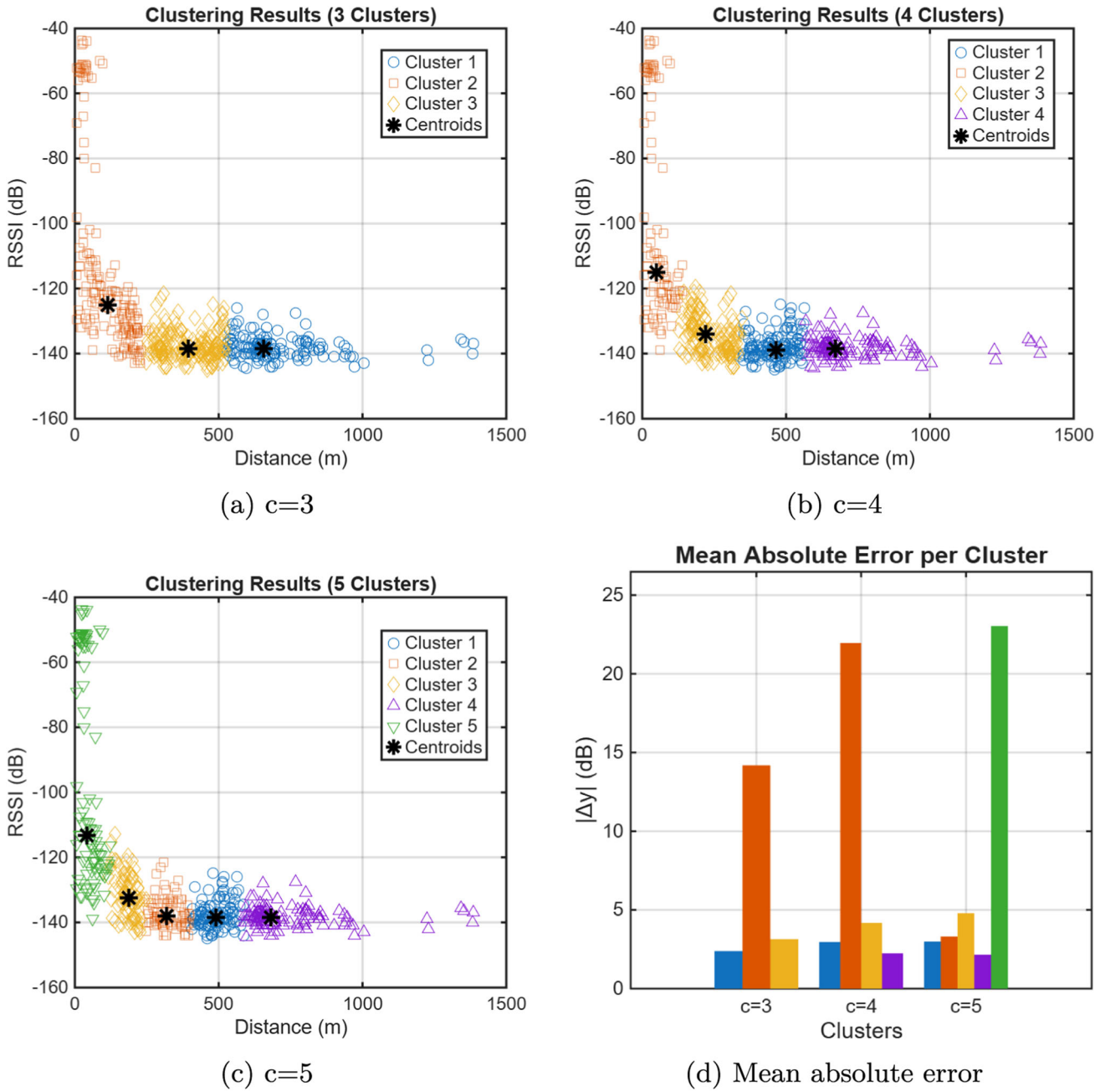
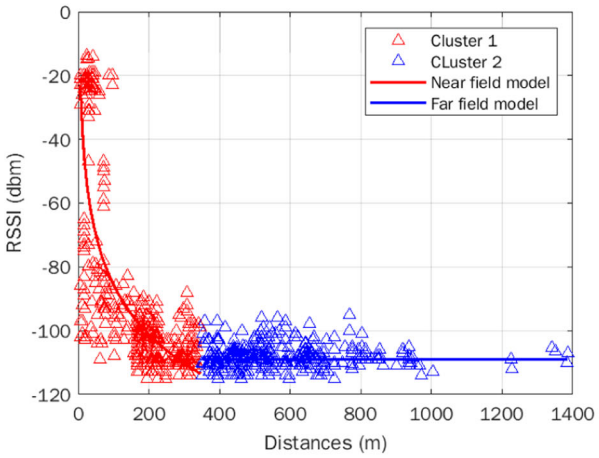


Fig. 6 Cityblock clustering results: (a) c=3, (b) c=4, (c) c=5, and (d) mean absolute error for each cluster

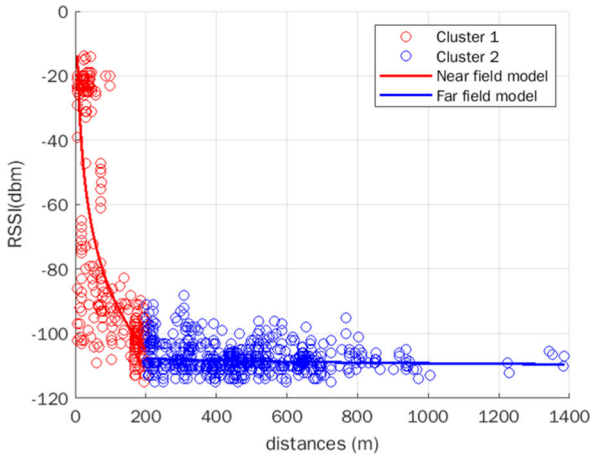
structured into three main stages: (i) cluster analysis to identify representative training samples and determine effective data segmentation, (ii) application of the developed path-loss model as a feature extractor, and (iii) one-hot encoding of the target output before feeding the data into the LSTM model for sector prediction. Through experimentation, we found that dividing the coverage area into three sectors yielded the best signal variation (see Figure 10), with clear directional patterns both toward and away from the gateway.

To clarify terminology used in the proposed model, we define the following:

1. **Predicted Sector (PS)**: A functional area (e.g., shopping center, open field) defined on the digital map, covering multiple location points. We evaluate both 3-sector and 4-sector divisions.
2. **User Data Record (UDR)**: A vector representing a user's signal at time t, given by $[Time, RSS, SNR]$.



(a) Regression of K-mean clustered data for far and near clusters



(b) Regression of Distribution-based clustered data for far and near clusters

Fig. 7 .

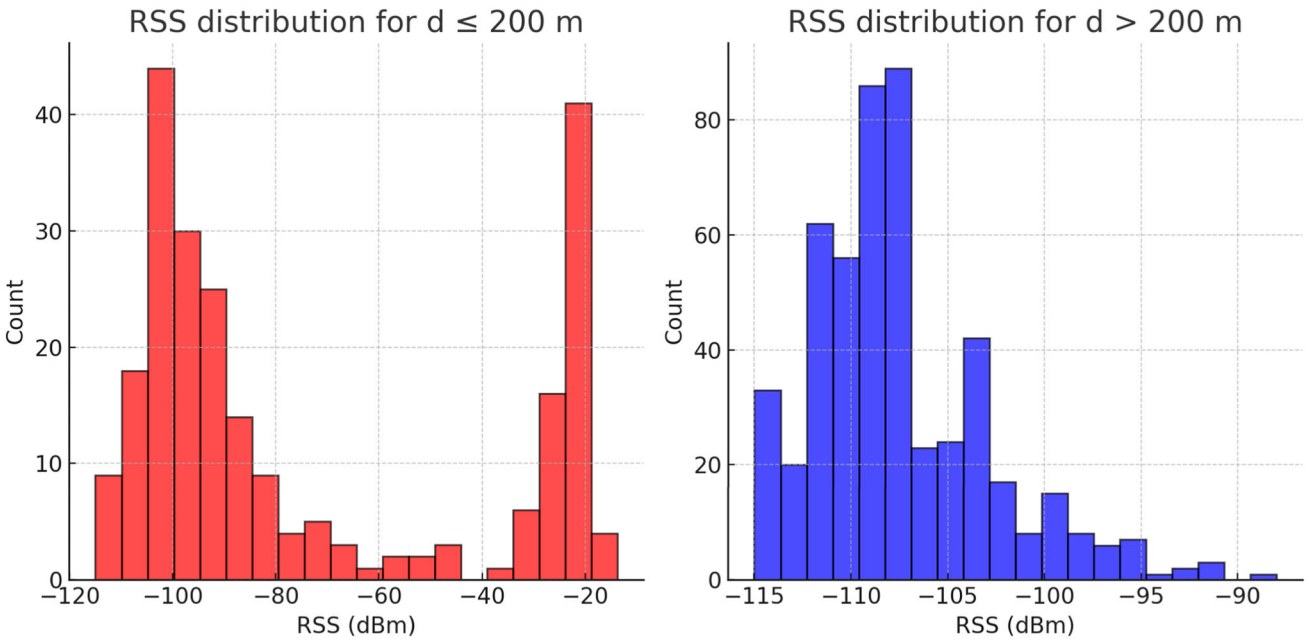


Fig. 8 RSS Distribution with the 200m threshold

3. **One-hot Encoded Output**: The model's classification output, represented as binary vectors: {001, 010, 100}.

4.2 Data augmentation

Obtaining large datasets can be challenging for many time series recognition tasks. A common approach to address this issue is data augmentation, which involves generating synthetic data from existing datasets to improve the performance of machine learning models [51].

In the context of outdoor localization, data augmentation can expand the training dataset's size and diversity, enabling the model to better generalize across various environments and conditions [44]. It also addresses the challenge of having a limited number of drive test samples compared to the training requirements for deep learning, helping prevent overfitting. Furthermore, the need for generalization is particularly crucial when working with real-world data, as it enables networks to handle the challenges posed by small datasets effectively [52].

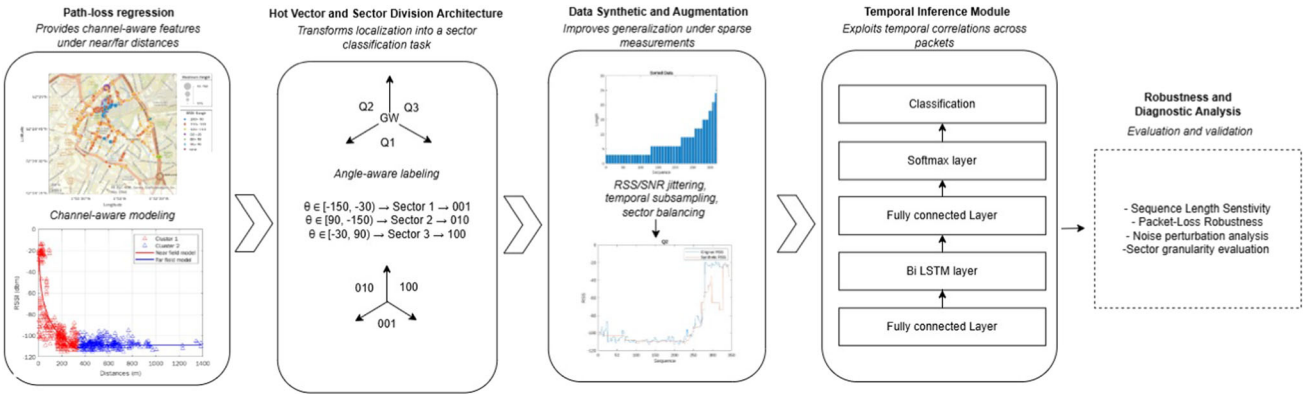
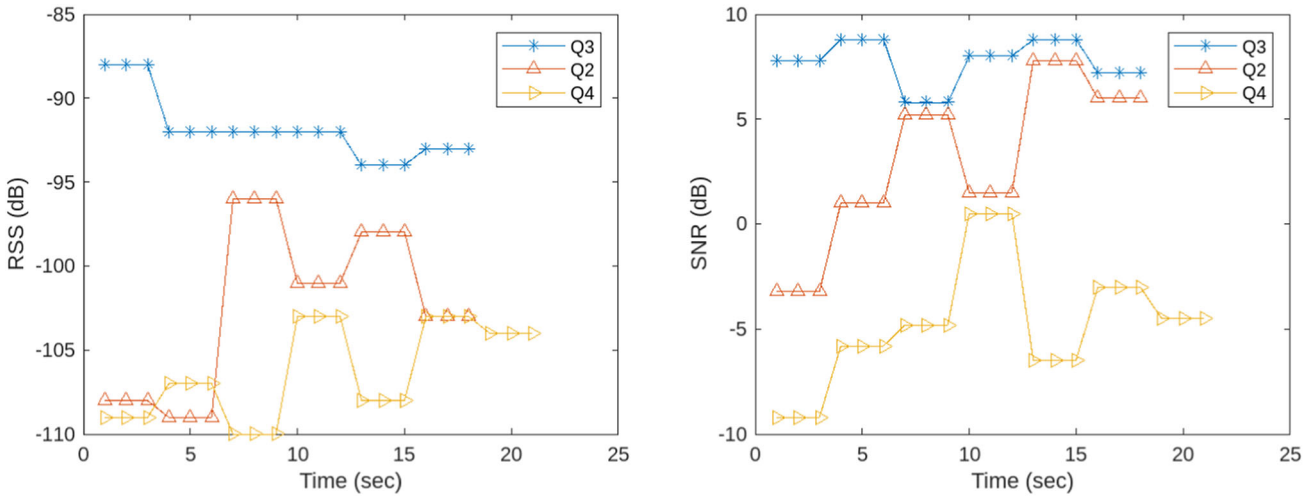


Fig. 9 Conceptual architecture of the proposed channel-aware temporal localization framework, illustrating how channel-aware modeling, geometry-based sectorization, data enrichment, and temporal inference are combined for single-gateway LoRaWAN localization. The figure highlights the diagnostic and robustness analyses used to assess sensitivity to temporal context, packet loss, noise perturbations, and sector granularity



(a) RSS Variation: Towards the gateway (b) SNR Variation: towards the gateway

Fig. 10 Signal Variations towards the gateway for far field distances. a): RSS Variation. b): SNR Variation

An impactful transformation-based data augmentation method is jittering, or noise injection into time series. This involves introducing Gaussian perturbations to RSSI and SNR values, where the parameters (mean and variance) are derived directly from the empirical distributions of the collected data. This ensures that the injected noise is statistically consistent with real-world measurement variability, such as device sensitivity or environmental fluctuations.

Jittering can be defined by:

$$x' = x_1 + n_1, ..., x_t + n_t, ..., x_T + n_T, \tag{8}$$

where n is the Gaussian noise added every time component t where $n \sim \mathcal{N}(\mu, \sigma^2)$. The standard deviation σ of the noise is a hyperparameter that needs to be defined in advance.

Beyond jittering, we also apply channel-model-based synthesis to generate additional RSSI samples. Here, new data points are produced by slightly perturbing existing node locations and estimating the corresponding RSSI using the adopted Birmingham path-loss model for near- and far-field propagation. This reverse-engineering of the collection process allows us to expand spatial coverage without additional measurements.

Together, these two approaches form our augmentation pipeline: jittering enriches the variability of the signal space, while channel-model synthesis extends the coverage of the dataset. Fig. 11 illustrates the agreement between original and synthetic RSS across the three sectors, highlighting that the generated data preserves the statistical patterns of the experimental measurements.

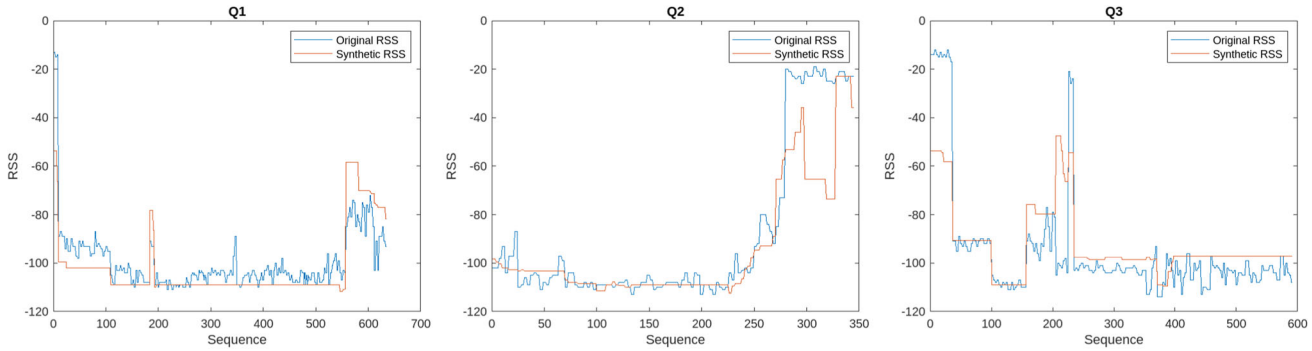


Fig. 11 Variation of original and synthetic RSS

4.3 Proposed LSTM architecture

This subsection explains the phases our proposed model undergoes. The right most block in Figure 9 shows the layer stages the training of the proposed model passes through. Initially, we explore different ways to differentiate between the data collected, whether it is near and far distances, or dividing the map into angle-based segments i.e sectors and study their characteristics. We found that dividing the map into 3 portions highlighted the differences and similarities of the data. During the training stage, we aim to obtain the optimal model with the highest accuracy, for further usage in the process of localizing the sector in the testing phase. The layers of the proposed LSTM is as shown below;

- Sequence Layer: To re-struct and normalize the sequential data to fit as an input for the following layers.
- Fully-connected layer: It is used to link all inputs from the previous layer to every activation unit in the next layer. The proposed model includes two fully-connected layers. The first contains [25, 50, 100] hidden neurons designed to find the correlation between the input features and the defined classes. The second functions as a classification layer, that matches the dataset classes to its hidden neurons.
- ReLU and Dropout layer: Every fully-connected layer is equipped with the Rectifier Linear Unit (ReLU) nonlinearity activation function and a dropout layer with a 60 percent reduction rate.
- Bidirectional Long Short Term Memory (BiLSTM) [53]:The major benefit of BiLSTM is that it can capture both forward and backward temporal dependencies. In other words, the BiLSTM learns from the full sequence of information at each time step. The BiLSTM is usually supported by a Dropout layer to help prevent overfitting. The dropout layer helps to drop random overhead neurons during training and, hence, reduces the sensitivity to the specific weights of individual neurons.
- Classification Layer: In the classification layer, the most suitable output expected would be a probability representing the odds of sector location probability, hence the SoftMax activation function used in the classification layer.

An essential component of the optimization algorithm is the loss function, which is used to evaluate the model's loss during each training epoch. Given that sector localization is a multi-class problem, the categorical cross-entropy loss function is employed that is calculated as:

$$(p, t) = -\sum_{n=1}^{K} y_{o,c} \log(p_{o,c}), \tag{9}$$

where number of classes is K, $y_{o,c}$ and $p_{o,c}$ are the real and predicted class label for the observation o in class c, [54]. Table 3 presents the hyperparameters along with their tested values and the optimal value determined for each.

The input of the LSTM model is the N features (F_1, F_2, ..., F_N), where $N = 5$ resulting from the collected data and the regression path-loss model. We use M training data samples representing the processed and augmented data, where each record in the M samples contains all of the N input features. The models' output is the probability that the input sample belongs to the designated sector represented as a hot encoded vector 001, 010, 100. To speed up training and mitigate overfitting, batch normalization was applied in intermediate layers. The network was trained using categorical cross-entropy loss combined with Adam optimization, which is the standard choice for multi-class classification with softmax outputs.

The dataset was collected over seven independent LoRa WAN measurement campaigns in Birmingham city center, using a single LoRa transmitter and one fixed gateway. Each campaign produced comma-separated value (CSV) files containing RSSI, SNR, and GPS coordinates, among other metadata. The raw data exhibited missing values, as is typical in outdoor field measurements. After preprocessing —

Table 3 Hyperparameter settings tested for the proposed model

Parameter	Value	Optimal Value
Model Type	Sequential	-
Batch size	24	-
Max. epochs	(75,100,125)	100
Optimizer	(Adam, SGDM , RMSProp)	ADAM
Learning rate	(0.001,0.0001, 0.00001)	0.001
Activation fn	Softmax	-
Training/ testing data	80% / 20%	-

which included removal of NaNs, conversion of RSSI from dBm to dB, and segmentation into 5, 10 and 15-second windows — the cleaned dataset contained 2,434 usable samples across 7 features: index, time, RSS, SNR, longitude, latitude, and distance. Outlier removal was performed using a ± 3 MAD threshold on RSSI values, ensuring that extreme measurements did not bias the model. This cleaned dataset was used for training and testing the LSTM. Train/test splits followed an 80/20 strategy (later validated with 5-fold cross-validation). The input to the LSTM was a multivariate time series, and the output was a three-class one-hot encoded label corresponding to the estimated location sector.

5 Results and validation

This section details the experimental findings of our study. We begin by examining the model tuning and training process, highlighting the impact of hyperparameter selection on convergence and performance. We then compare two hybrid architectures, the CNN–LSTM and the data-driven LSTM, to determine the benefits of temporal modeling versus convolutional feature extraction. A broader network evaluation follows, where we benchmark the proposed approach against conventional baselines to establish relative performance.

To strengthen rigor, we next employ cross-validation, ensuring that generalization is tested under group-based splits that eliminate potential train–test leakage. We further investigate the robustness of the model to noise, where controlled perturbations of RSSI values quantify stability under realistic measurement conditions. The role of synthetic data augmentation is then assessed through ablation studies, clarifying its contribution to class balance, generalization, and performance gains over raw data. We also analyze sector partition sensitivity by comparing three-sector and four-sector configurations, assessing how angular resolution influences training stability and test accuracy.

Beyond these evaluations, we explicitly examine the contribution of temporal dependencies through a sequence-length ablation study, quantifying how increasing temporal context enhances localization performance. Robustness under adverse channel conditions is further assessed via a packet-loss sensitivity analysis that emulates missing LoRaWAN transmissions. Finally, we evaluate inference-time scalability with respect to sequence length and the number of concurrent users, providing insight into the practical deployability of the proposed framework at the gateway.

Together, these evaluations provide a comprehensive validation of the proposed framework, spanning tuning, architecture comparison, rigorous evaluation protocols, robustness analyses, and sensitivity studies.

5.1 Model tuning & training

Prior to the training, the optimal hyperparameters of the LSTM model were selected after evaluating its performance against various combinations of parameters. The initialized hyperparameters of the model play a crucial role in influencing the solution obtained by the learning algorithm [55]. During experimentation, we have employed different ways to find the highest classification accuracy. To present an unbiased map representation, we ensured in our experimentation the uniform distribution of data collected into 3 sectors. In that sense, we compiled a solution to iteratively consider all angles spanning from 0 to 360, that finds the best equally sectored solution.

We further study and compare the performance of the proposed model using 3 different optimizers that can be used to improve the learning rate of the network performance. The Root Mean Squared Propagation (RMSProp), Stochastic gradient descent (SGDM) and adaptive moment estimation (ADAM). ADAM results in a testing accuracy of 83.8% outperforming both SGDM (56.2 %) and RMSProp (77.9 %) as the gradient optimization process where Adam takes into account the past gradients and momentum. Moreover, learning rate is a critical hyperparameter in model training. If set too high, it may cause the model to converge prematurely, resulting in a sub-optimal solution. We compare the results of using various learning rates of 0.001, 0.0001 and 0.00001, concluding that a learning rate of 0.001 enables the model to achieve convergence at a suitable pace, resulting in optimal performance. During training, a complete iteration of the input data through the NN is defined as an epoch [44]. An epoch comprises one or more training data batches, which the model processes sequentially, after a full epoch, the model's parameters are adjusted based on the error or loss computed from the training data. Table 4 shows the effect of changing the number of epochs on the training and testing accuracy for various number of layers.

The LSTM architecture contains $\approx$63k trainable parameters ($\approx$0.25 MB in 32-bit floating point). The inference cost is $\approx$54k multiplications per time step, which for a sequence of 100 packets corresponds to $\approx$5.4M FLOPs

Table 4 Model Training and Validation accuracy (%) for variable number of layers (ADAM Optimizer and Learning rate = 0.001)

Number of Epochs	[50,100,50]	[100,100,100]	[50,50,50]
75	85.9 %, 60.1%	85.9%, 69.4%	75.2%,56.3%
100	**83.8%, 72.1%**	88.2%, 64.8%	84.4%,61.5%
125	84.8% , 62.1%	85.6% , 61.0%	87.2% , 61.5%

Table 5 Balanced accuracy, sensitivity, and specificity with 95% bootstrap confidence intervals (CI) for each sector

	Q1	Q2	Q3
Balanced Accuracy	0.802 [0.714, 0.877]	0.723 [0.640, 0.802]	0.742 [0.662, 0.819]
Sensitivity (Sn)	0.686	0.672	0.781
Specificity	0.918	0.774	0.703

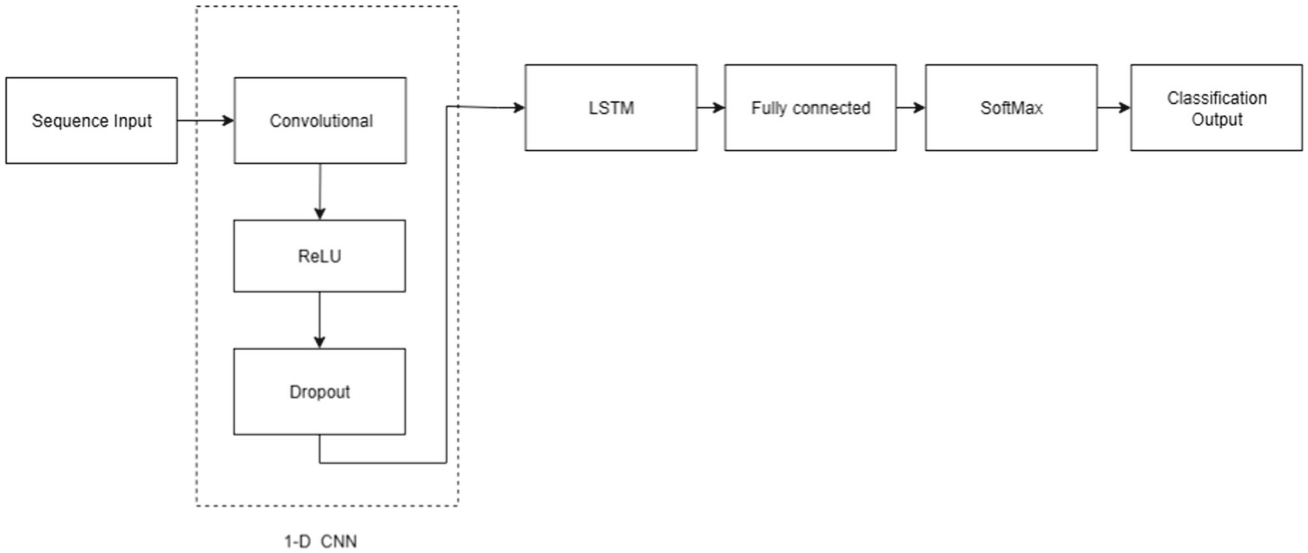


Fig. 12 Baseline 1-D CNN–LSTM architecture used for comparative evaluation with the proposed framework

(<0.01 GFLOPs). This level of complexity is modest and feasible for real-time execution at LoRaWAN gateways or end nodes with low-power CPUs. For ultra-constrained nodes, lighter alternatives such as GRUs (which reduce parameters by $approx25\%$), pruning/quantization, or knowledge distillation could further reduce footprint without significant accuracy loss. The LSTM architecture is intentionally lightweight to ensure compatibility with real-time processing constraints at LoRaWAN gateways, avoiding unnecessary model complexity while effectively exploiting temporal dependencies in packet-level measurements.

Convergence analysis during training (measured over iterations) as shown in Fig.13 shows that validation accuracy stabilizes after 900 updates, corresponding to $\approx$ 10 full epochs and then remains within a narrow range of 63–65%. At the same time, validation loss levels off near 0.9 with only small fluctuations. This indicates that further training mainly captures details specific to the training data rather

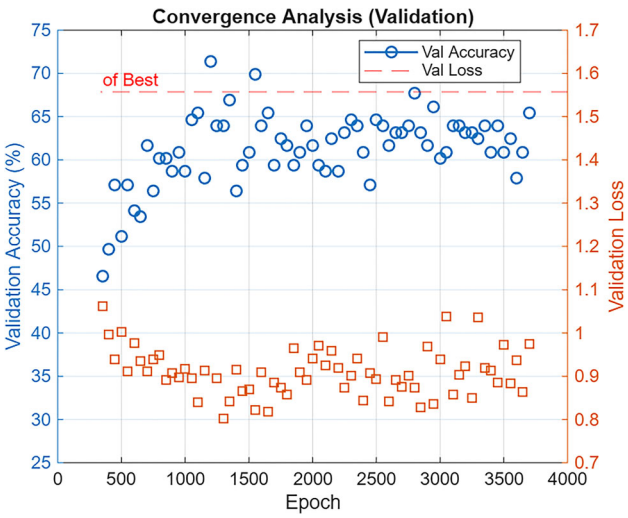


Fig. 13 Evaluation metrics for multi-sector classification for Data-driven LSTM and CNN-LSTM

than improving generalization. To avoid unnecessary computation, early stopping was applied once the plateau was reached. Training completed in approximately 2 minutes on a CPU-only machine, indicating low computational overhead during optimization. Inference was efficient, requiring only 0.039 s for 133 test sequences (0.30 ms per sequence). This demonstrates that the model is feasible for real-time or near–real-time deployment in LoRaWAN localization applications.

5.2 Hybrid LSTM : CNN-LSTM and data-driven LSTM

Convolutional Neural Networks (CNNs) are a highly effective type of deep learning model extensively utilized across a variety of domains, including object detection, speech recognition, and image classification [56]. They have also proven effective in time-series prediction tasks, where their ability to automatically extract hierarchical features from raw input makes them particularly valuable [57]. Since the performance of time-series neural networks significantly relies on the quality of input features [58], CNNs are often combined with sequence models such as Long Short-Term Memory (LSTM) networks. This combination leverages CNNs for feature extraction and LSTMs for temporal dependency modeling [59].

To provide a meaningful baseline for comparison with our proposed LSTM model, we implemented a CNN–LSTM architecture. As illustrated in Figure 12, the model begins with a convolutional feature extraction block (convolution, ReLU activation, and dropout), followed by an LSTM layer that captures temporal patterns. A fully connected layer, softmax, and classification output complete the network. This allows us to benchmark our proposed model against a well-established deep learning paradigm for time-series classification. We compared our proposed LSTM model against the CNN–LSTM baseline using McNemar's test on paired predictions. The test yielded a chi-squared statistic of $\chi^2 = 9.26$ with a p-value of $p = 0.0023$, confirming that the improvement of our model is statistically significant at the 95% confidence level.

5.3 Network evaluation

In this work, we evaluate our proposed model through standard classification metrics. A confusion matrix (M) summarizes the results for three classes, where Q1 = [001], Q2 = [010], and Q3 = [100]. It shows True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN), from which accuracy ($ACC = \frac{TP+TN}{TP+TN+FP+FN}$), precision ($P = \frac{TP}{TP+FP}$), sensitivity ($Sn = \frac{TP}{TP+FN}$), specificity ($Sp = \frac{TN}{TN+FP}$) and F1-score ($F1 == \frac{2\times TP}{2\times TP+FN+FP}$) are derived.

We evaluated the accuracy of the proposed model on both training and testing data, as illustrated in Figure 14. In the training set, the model correctly classified 181 segments of Q1 (87.0%), 140 segments of Q2 (67.3%), and 290 segments of Q3 (92.7%). For the testing set, 44 out of 60 Q1 samples were correctly classified (73.3%), while Q2 achieved approximately 50% accuracy. Q3 showed the strongest performance, with 91% of samples correctly classified and only six misclassified as Q1 or Q2. Overall, the model achieved an accuracy of 72.1% across all test predictions. To provide a more robust statistical assessment, we computed balanced accuracy with 95% bootstrap confidence intervals, along with sensitivity (Sn) and specificity for each class. The results are summarized in Table 5. These values confirm that performance is substantially above chance (0.50) across all sectors, with Q1 showing the highest reliability.

Figure 15a compares the F1-score, recall, and precision of the proposed LSTM model against the CNN–LSTM baseline for the three sectors. For sector Q1 (001), the proposed model achieved an F1-score of 0.80, recall of 0.73, and precision of 0.86, outperforming the CNN–LSTM which obtained 0.74, 0.69, and 0.81, respectively. In sector Q2 (010), the proposed LSTM reached 0.61 F1-score, 0.48 recall, and 0.84 precision, compared to 0.52, 0.41, and 0.72 for CNN–LSTM, indicating notable improvements in all three metrics. For sector Q3 (100), the LSTM produced an F1-score of 0.73, recall of 0.91, and precision of 0.61, slightly higher than the CNN–LSTM values of 0.69, 0.89, and 0.60. Overall, the proposed LSTM demonstrates consistent gains across sectors, with the most substantial improvements observed in recall and F1-score for Q1 and Q2.

To assess spatial performance, we first calculated the hard-label angular error, which measures the deviation between the center of the true sector and the center of the sector predicted by argmax decoding of the softmax output. This resulted in a mean error of 38.8°, a median of 0°, corresponding to 32.3% of the sector width. The distribution was bimodal, with predictions either exactly correct (0° error) or shifted into the neighboring sector (120° error). Since only three sectors are defined, any misclassification necessarily corresponds to one of the two neighbors. For these misclassified samples, the mean distance to the nearest boundary was 39.4°, showing that errors typically occurred well inside neighboring regions rather than at their edges. The model reached 100% Top-2 accuracy, confirming that the true sector was always among the two most likely predictions. We also considered softmax-weighted angular estimates, where instead of taking only the most probable sector, the class probabilities are projected onto the unit circle to yield a continuous angular prediction. This approach better reflects the probabilistic nature of the network's output. Under this measure, the mean error was 49.8° with a median of 28.1°, producing a smoother error distribution compared with the discrete hard-label case.

Fig. 14 Confusion matrix representing the performance of the Data-driven LSTM a) Training data accuracy. b) Testing data

As shown in Figure 15b, the predictions remain clustered around the true sector directions with moderate shifts into neighboring regions. This illustrates that probability mass is consistently shared across adjacent sectors, pulling the angular estimate closer to the true direction and preventing errors from spreading randomly across the angular space.

5.4 Temporal dependency analysis via sequence-length ablation

To explicitly quantify the contribution of temporal dependencies to single-gateway sector localization, we conduct a sequence-length ablation study in which the number of consecutive packets provided to the LSTM is systematically varied.

As shown in Table 6, localization performance improves as longer temporal context is incorporated. Using a single packet ($L = 1$) yields 57.9% accuracy, while increasing the context to $L = 5$ improves accuracy to 63.3%. The best performance is achieved at $L = 8$, reaching 72.5% accuracy and a Macro-F1 score of 0.719. These results demonstrate that temporal modeling plays a critical role in this setting: aggregating multiple consecutive transmissions allows the network to suppress short-term fading and measurement noise while capturing mobility-induced signal evolution, resulting in more reliable sector estimation than memoryless (single-packet) inference. Longer temporal contexts ($L \geq 10$) were not evaluated due to insufficient sequence availability under the adopted data split protocol, reflecting realistic LoRaWAN sampling sparsity rather than a limitation of the proposed model.

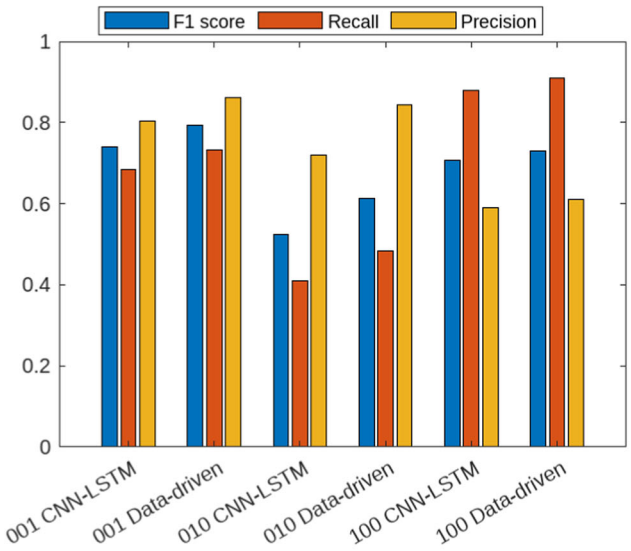
Table 6 Sequence-length ablation study: effect of temporal context length L (number of consecutive packets) on localization performance. Accuracy and Macro-F1 are reported on the held-out test set

Seq. length L	Test samples	Accuracy	Macro-F1
1	133	0.579	0.530
2	133	0.534	0.511
3	133	0.534	0.513
5	60	0.633	0.622
8	40	**0.725**	**0.719**

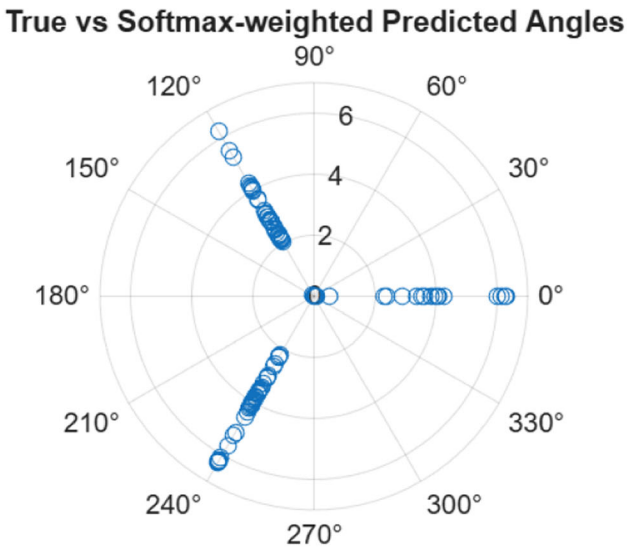
5.5 Sensitivity to packet Loss

To evaluate robustness under varying channel conditions, we conduct a packet-loss sensitivity analysis in which a fraction of timesteps is randomly dropped from each test sequence. This procedure emulates missing LoRaWAN packets caused by fading, interference, or duty-cycle constraints. For each packet-loss rate, results are averaged over 20 stochastic trials.

Table 7 and Fig. 16 show that localization accuracy degrades gradually as packet loss increases, indicating graceful performance degradation. These results demonstrate that the proposed temporal inference framework is able to exploit



(a) Evaluation metrics for multi-sector classification for Data-driven LSTM and CNN-LSTM.



(b) True vs. softmax-weighted predicted angles, showing clustering near sector directions with moderate deviations into adjacent regions.

Fig. 15 Comparison of evaluation metrics and angular prediction behavior of the proposed model

Table 7 Sensitivity to packet loss. Test accuracy is reported as mean ± standard deviation over 20 random trials, obtained by randomly dropping timesteps from each test sequence.

Packet loss rate	Accuracy (mean ± std)
0%	0.684 ± 0.000
10%	0.677 ± 0.010
20%	0.623 ± 0.009
30%	0.582 ± 0.011
40%	0.538 ± 0.012
50%	0.496 ± 0.012

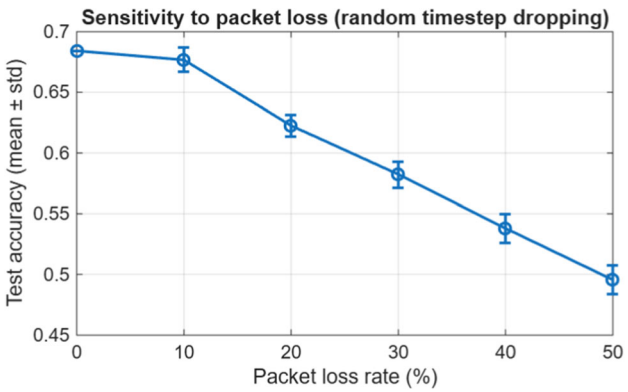


Fig. 16 Accuracy versus packet loss rate (mean ± standard deviation over 20 trials). Packet loss is simulated by randomly dropping a fraction of timesteps from each test sequence to emulate missing LoRaWAN packets under varying channel conditions

remaining sequential structure even under moderate packet loss, highlighting its robustness to realistic channel impairments.

5.6 Scalability analysis

We evaluate inference-time scalability with respect to the sequence length L and the number of concurrent inference requests U. All measurements are conducted on a CPU-only platform. For practical sequence lengths ($L \geq 3$), inference latency remains below 0.15 ms per sequence.

As shown in Table 8, total inference time scales approximately linearly with U, while the average per-sequence latency decreases for larger batch sizes due to amortized overhead. These results indicate that the proposed framework can support thousands of users per second at the gateway, making it suitable for real-time deployment in large-scale LoRaWAN scenarios.

5.7 Cross-validation

In our initial split-based evaluation, the proposed model achieved an overall accuracy of 72.1%. To address the possibility of train–test leakage, we employed a leave-one-group-out evaluation strategy. Since the dataset was aggregated across multiple measurement campaigns, explicit

Table 8 Inference-time scalability with sequence length and number of sequences (CPU-only)

Setting	Total time (s)	Latency (ms/seq)
$L = 3$ (133 seq)	0.0065	0.049
$L = 5$ (60 seq)	0.0048	0.079
$L = 8$ (40 seq)	0.0049	0.122
$U = 50$ (seq)	0.0108	0.216
$U = 100$ (seq)	0.0140	0.140
$U = 133$ (seq)	0.0160	0.120

Table 9 Mean confusion matrix across 5 folds (values are mean ± std)

	Pred Q1	Pred Q2	Pred Q3
True Q1	34.2 ± 2.7	2.8 ± 1.7	4.4 ± 0.8
True Q2	6.0 ± 2.7	31.6 ± 2.9	16.2 ± 3.5
True Q3	7.8 ± 2.9	18.0 ± 2.8	56.0 ± 7.2

trajectory/day identifiers were not retained. Instead, we partitioned the dataset into five non-overlapping groups, each containing distinct measurement sequences. In each fold, one group was completely excluded for testing, while the remaining four groups were used for training and validation. This guarantees that no packets from the same group appear in both training and testing.

Table 9 reports the mean confusion matrix across the five folds, with values shown as mean ± standard deviation. Under this stricter protocol, the proposed LSTM achieved an average accuracy of **68.8%** ± **2.2**, a balanced accuracy of **76.9%** ± **1.6**, and a macro-F1 score of **68.8%** ± **1.9**. While the overall accuracy is slightly lower than the original split-based evaluation, this reduction is expected, as the group-based approach eliminates overlap between training and test data. Importantly, the balanced accuracy and macro-F1 confirm that the model generalizes reliably across all classes when tested under more rigorous, leakage-free conditions.

5.8 Robustness to noise

We assess noise robustness via post-hoc test-time corruption of the held-out real test set. RSSI values are perturbed with zero-mean Gaussian noise at controlled levels (fixed magnitudes of $\{\pm1, \pm2, \pm3, \pm5\}$ dB). The results in Fig. 17 shows that accuracy remained stable (58%) across all noise levels, indicating that the LSTM is resilient to measurement fluctuations at inference time. The trained model is kept fixed and evaluated on $R = 20$ stochastic corruptions per noise level. This protocol avoids circularity because no model-based synthesis is used at test time, and it directly quantifies degradation under realistic measurement perturbations. For

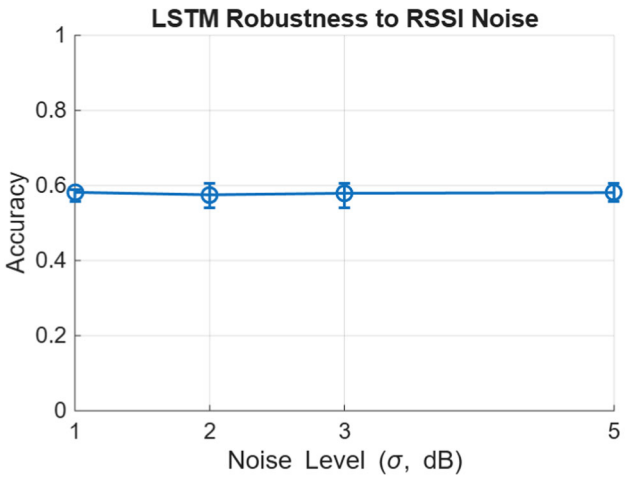


Fig. 17 Robustness of model under post-hoc Gaussian noise injection to RSSI values

clarity, the robustness-to-noise analysis was performed using only the original experimental dataset, without augmentation or synthetic data. This ensures that the reported stability under noise reflects the inherent generalization ability of the LSTM on real-world measurements.

5.9 Synthetic data & ablation studies

To evaluate the effect of synthetic augmentation, we considered two cases: (i) synthetic samples were restricted to the training phase, with testing performed only on real experimental data, and (ii) the model was trained and tested solely on the original dataset without any synthetic augmentation. These comparisons isolate the role of synthetic data and demonstrate how augmentation influences model balance and generalization.

We retrained the model with synthetic augmentation but evaluated strictly on the 20% of the original collected dataset. The balanced accuracy in this case was 71.6%, compared to 72.1% when testing on the combined original+augmented set in fig. 18a. The small difference confirms that augmentation primarily improves robustness during training without artificially inflating performance at test time. We note that quadrant 2 was absent from the test set, which may also inflate reported accuracy.

When trained without synthetic data, the LSTM reflects true performance on real measurements but suffers from class imbalance, leading to weaker recognition of minority classes. With synthetic augmentation, class balance improves and testing accuracy increases, particularly for under-represented sectors, though at the cost of potential reliance on artificial patterns. Training and testing results are shown in Fig 18b and 18c.

We have now explicitly reported the results for both cases. Without any augmentation or synthetic data, the LSTM

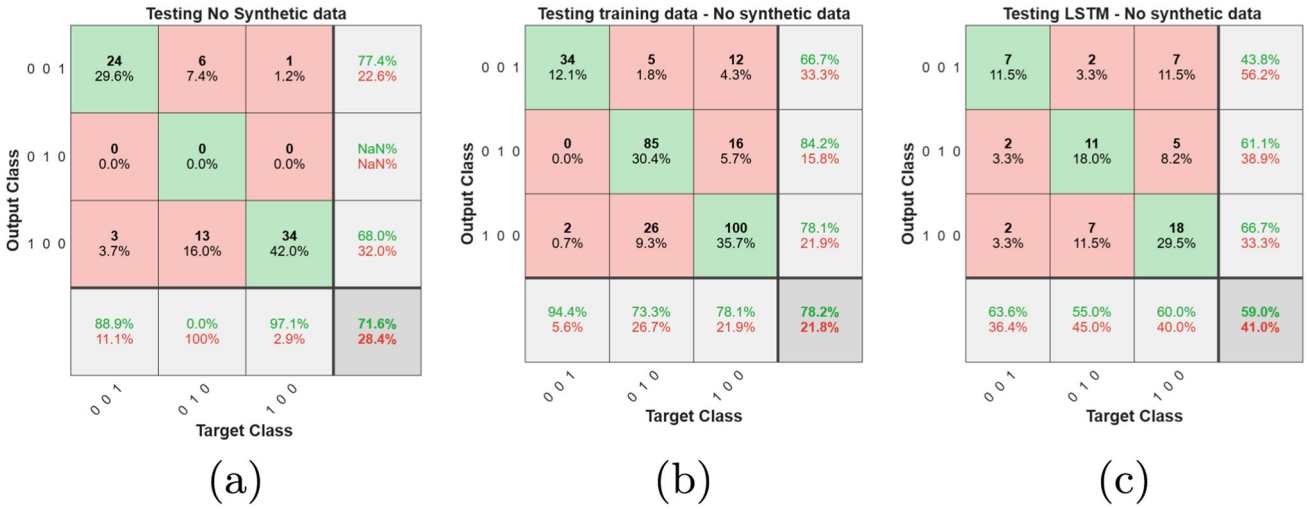


Fig. 18 Confusion matrices for different training scenarios: (a) LSTM trained on augmented data but tested without any, (b) LSTM trained without any augmented data, and (c) LSTM trained and tested without any augmented data

Table 10 Performance comparison of baseline models and proposed LSTM variants

Model	Accuracy (%)
Majority-class predictor	42.8
Logistic Regression (static RSSI/SNR/dist.)	30.0
SVM (static RSSI/SNR/dist.)	30.9
kNN (static RSSI/SNR/dist.)	27.0
HMM (discretized RSSI)	46.1

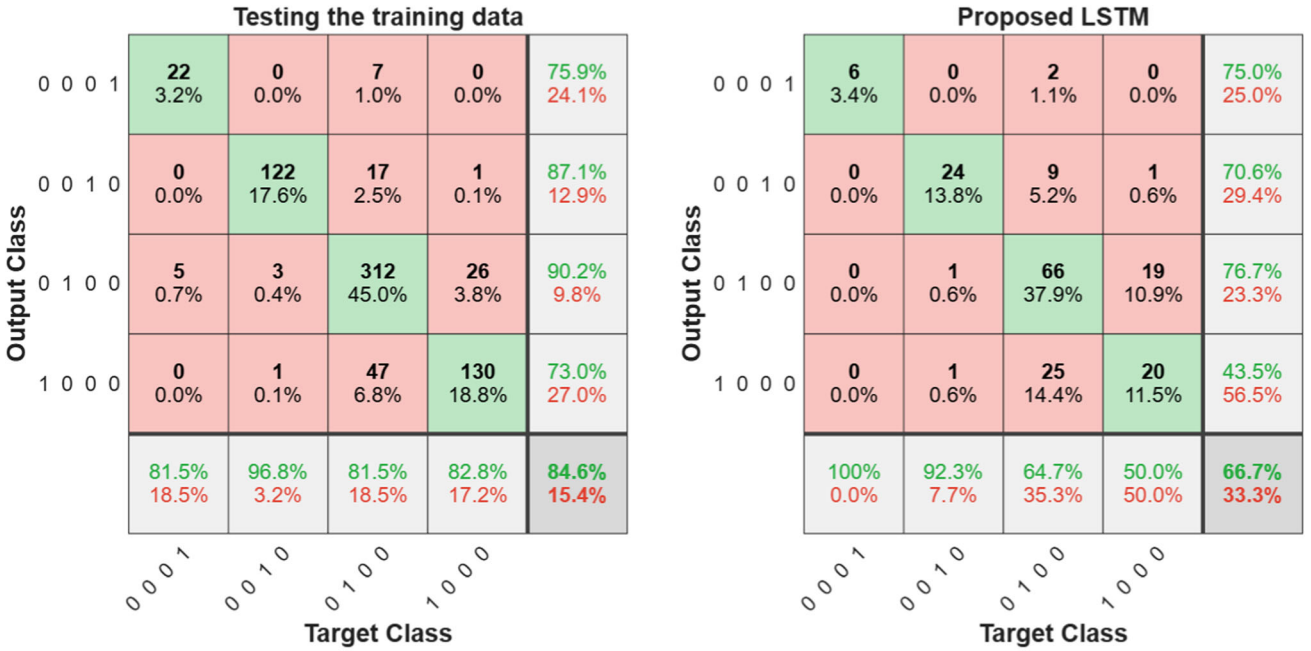
achieves 59% test accuracy. When distribution-based augmentation (synthetic channel-model samples and jittering using RSS/SNR distributions) is included, test accuracy improves to 72%. This 13% absolute gain reflects the limited size and imbalance of the raw dataset: the model underfits when trained solely on real data. Augmentation expands the training space with statistically consistent samples, leading to better generalization. Such improvements are in line with prior studies in RF-based localization, where augmentation is known to mitigate data scarcity and class imbalance.

To ensure fairness and rigor, we extended our evaluation to include several baseline models commonly used in wireless localization. These baselines (kNN, Logistic Regression, SVM, and HMM) were trained on static features such as RSSI, SNR, and estimated distance, without temporal modeling. As shown in Table 10, these methods achieved modest performance (27–46% accuracy), with some collapsing to majority-class predictions, reflecting their limited ability to capture LoRaWAN's temporal variability. To further assess the role of channel-model features, we implemented an ablation study by training the LSTM using only raw sequential RSSI/SNR values. This reduced-variant achieved 42% accuracy, already surpassing most static baselines.

5.10 Sector partition sensitivity (3Q vs. 4Q)

To assess the sensitivity of the proposed approach to sector angle partitioning, we compared three-sector (3Q) and four-sector (4Q) configurations. Angles were fixed using the training data and preserved during testing, thus avoiding any test leakage. The comparative evaluation between the 3Q and 4Q sector configurations reveals notable differences in sensitivity and generalization. In the four-sector case shown in Figure 19, the model achieved relatively high training accuracy across all classes (overall 84.6%), but this performance did not translate to the testing stage, where the overall accuracy dropped sharply to 66.7%. This degradation was particularly pronounced in Classes 3 and 4, where the accuracy fell to 64.7% and 50%, respectively. Such a significant gap between training and testing indicates overfitting, suggesting that the model captured sector-specific patterns during training but failed to generalize reliably to unseen data.

In contrast, the three-sector configuration exhibited a more balanced performance. While training accuracy was lower (72.1%), the testing accuracy improved to 83.8%, reflecting stronger generalization capability. Class-level performance was more consistent, with all sectors achieving reasonably high accuracies, particularly a recovery in Class 3 (61.0% in training versus 78.4% in testing). This indicates lower sensitivity to deployment conditions and a more stable model across varying operating environments.



(a) Training results (b) Testing results

Fig. 19 Comparison of training and testing results

Overall, the three-sector setup emerges as the more optimal configuration for this dataset, offering reduced sensitivity between training and testing and providing more robust predictive performance. The four-sector design, while attractive for finer granularity, may require additional data augmentation, regularization, or optimization strategies to mitigate overfitting and achieve comparable robustness.

6 Discussion & practical implications

The clustering approaches demonstrate improvements that are not only statistically significant but also operationally meaningful. Specifically, K-means reduces the mean absolute error by about 28.6% (≈2.9 dB) and the distribution-based clustering (DBC) achieves a 16% reduction (≈1.6 dB) relative to the Birmingham baseline model. In practical terms, such reductions shrink distance uncertainty by 10–30% depending on the urban path-loss exponent, directly improving localization accuracy and reducing misclassification between sectors. From a network planning perspective, these gains can either be translated into smaller fade margins—saving 2–3 dB in the link budget—or into extended coverage radius. This has tangible benefits for gateway placement, ensuring more reliable QoS in dense urban deployments.

Beyond clustering, broader validation experiments highlight several practical insights. Cross-validation confirms that the model generalizes across independent campaigns, minimizing train-test leakage and supporting real-world deployment where signal statistics vary by day or mobility pattern. Robustness-to-noise analysis shows stable accuracy under RSSI perturbations up to ±5 dB, indicating resilience to typical interference and multipath effects.

Augmentation and ablation studies further demonstrate the value of synthetic data. Without augmentation, class imbalance leads to degraded accuracy, whereas augmentation improves generalization by up to 13% absolute. Crucially, augmented samples were excluded from testing, ensuring unbiased performance estimates. This approach enables model bootstrapping in scenarios with limited real-world data.

Sector-partition sensitivity analysis reveals that finer angular resolution (four sectors) leads to overfitting, while a coarser three-sector division yields better robustness and generalization. Misclassifications remain spatially adjacent, preserving directional consistency. This suggests partition granularity should align with coverage goals, training data availability, and desired trade-offs between accuracy and reliability.

In summary, the validation studies demonstrate that the proposed framework is both statistically sound and practically deployable. The improvements translate into measurable savings in link budgets, enhanced sector classification stability, and more reliable coverage in urban environments. For practitioners, these results suggest that data-driven local-

ization models can be effectively integrated into LP-WAN planning workflows, provided that careful attention is given to training data diversity, augmentation strategies, and sector partitioning choices.

7 Conclusion and future directions

The presented work characterized and implemented an outdoor, LoRa-based, single gateway coverage network. The deployment resulted in a LoRa coverage map in city center Birmingham, the data collected was used to study the distance-power relation between the gateway and the mobile node and presented a well-expressed path-loss model for Birmingham city center. The path-loss model was established after enhancements including repetition analysis, regression, and clustering.

Most importantly, this paper introduces a novel data-driven framework for location estimation using time-series data, where the path-loss model is explicitly integrated into LSTM layers during training. This combination of statistical modeling and deep sequential learning enables reliable sector-level localization of transmitting nodes from consecutive LoRaWAN packets. Performance is benchmarked against the CNN–LSTM baseline, with the proposed LSTM framework achieving superior accuracy and robustness.

This work focuses intentionally on the single-gateway case as a baseline, motivated by low-power and infrastructure-constrained scenarios where multi-gateway deployment is impractical. That said, the proposed approach is inherently extensible: as the number of nodes or gateways increases, the LSTM framework can be retrained with richer input features such as, per-gateway RSSI/SNR. Therefore, this study is viewed as a foundation, demonstrating feasibility under constrained conditions — while leaving multi-node/multi-gateway scalability as a natural and valuable direction for future work.

Future research can extend this work in several promising directions. Federated learning offers the potential to train localization models collaboratively across multiple gateways while preserving data privacy and minimizing communication overhead. Finally, reinforcement learning–based adaptive strategies may allow real-time adjustment of localization parameters, enhancing robustness under dynamic channel conditions. Together, these directions highlight the broader applicability of the proposed framework in next-generation IoT networks.

Author Contributions All authors contributed to the study conception. Data collection and analysis were performed by AlaaAllah ElSabaa. The first draft of the manuscript was written by AlaaAllah ELSabaa and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding The authors received no financial support for the research, authorship, and/or publication of this article.

Data Availability Data collected through experimentation that is used in this manuscript is available upon request.

Declarations

Competing interests The authors declare no competing interests.

References

1. Markit, I. (2017). The internet of things: A movement, not a market. *IHS Market, 1*(1), 1.
2. Li, X., Pang, H., Li, G., Jiang, J., Zhang, H., Gu, C., & Yuan, D. (2024). Wireless positioning: Technologies, applications, challenges, and future development trends. *CMES-Computer Modeling in Engineering & Sciences,138*(3).
3. Shahbazian, R., Macrina, G., Scalzo, E., & Guerriero, F. (2023). Machine learning assists iot localization: A review of current challenges and future trends. *Sensors, 23*(7), 3551.
4. Nor, R.F.A.M., Zaman, F.H., & Mubdi, S. (2017). Smart traffic light for congestion monitoring using lorawan. In: 2017 IEEE 8th Control and System Graduate Research Colloquium (ICSGRC), pp. 132–137 . IEEE.
5. Mdhaffar, A., Chaari, T., Larbi, K., Jmaiel, M., & Freisleben, B. (2017). Iot-based health monitoring via lorawan. In: IEEE EUROCON 2017–17th International Conference on Smart Technologies, pp. 519–524 . IEEE.
6. Semtech, A. (2022). LoRa^{TM} Modulation Basics, Application Note .
7. Popli, S., Jha, R. K., & Jain, S. (2018). A survey on energy efficient narrowband internet of things (nbiot): architecture, application and challenges. *IEEE Access, 7*, 16739–16776.
8. Bagariang, Y., Nashiruddin, M.I., & Adriansyah, N.M. (2019). Lora-based iot network planning for advanced metering infrastructure in urban, suburban and rural scenario. In: 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), pp. 188–193 . IEEE.
9. El Chall, R., Lahoud, S., & El Helou, M. (2019). Lorawan network: Radio propagation models and performance evaluation in various environments in lebanon. *IEEE Internet of Things Journal, 6*(2), 2366–2378.
10. Almuhaya, M. A., Jabbar, W. A., Sulaiman, N., & Abdulmalek, S. (2022). A survey on lorawan technology: Recent trends, opportunities, simulation tools and future directions. *Electronics, 11*(1), 164.
11. Pimpinella, A., Redondi, A.E., Nicoli, M., & Cesana, M. (2020). Machine learning based localization of lorawan devices via inter-

technology knowledge transfer. In: 2020 IEEE International Conference on Communications Workshops (ICC Workshops), pp. 1–6 . IEEE.
12. Deese, A. S., Jesson, J., Brennan, T., Hollain, S., Stefanacci, P., Driscoll, E., Dick, C., Garcia, K., Mosher, R., Rentsch, B., et al. (2020). Long-term monitoring of smart city assets via internet of things and low-power wide-area networks. *IEEE Internet of Things Journal, 8*(1), 222–231.
13. Piechowiak, M., Zwierzykowski, P., & Musznicki, B. (2023). Lorawan metering infrastructure planning in smart cities. *Applied Sciences, 13*(14), 8431.
14. Anjum, M., Khan, M. A., Hassan, S. A., Mahmood, A., Qureshi, H. K., & Gidlund, M. (2020). Rssi fingerprinting-based localization using machine learning in lora networks. *IEEE Internet of Things Magazine, 3*(4), 53–59.
15. Moradbeikie, A., Keshavarz, A., Rostami, H., Paiva, S., & Lopes, S. I. (2021). Gnss-free outdoor localization techniques for resource-constrained iot architectures: A literature review. *Applied Sciences, 11*(22), 10793.
16. Yang, Z., Zhou, Z., & Liu, Y. (2013). From rssi to csi: Indoor localization via channel response. *ACM Computing Surveys (CSUR), 46*(2), 1–32.
17. Bécue, A., Praça, I., & Gama, J. (2021). Artificial intelligence, cyber-threats and industry 4.0: Challenges and opportunities. *Artificial Intelligence Review, 54*(5), 3849–3886.
18. Janssen, T., Berkvens, R., & Weyn, M. (2019). Comparing machine learning algorithms for rss-based localization in lpwan. In: International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp. 726–735 . Springer.
19. Zhang, Y., Ma, L., Xu, Y., & Sun, Y. (2019). An rss pathloss considered distance metric learning for fingerprinting indoor localization. In: 2019 IEEE Global Communications Conference (GLOBECOM), pp. 1–6 . IEEE.
20. Van Komen, D. F., Neilsen, T. B., Howarth, K., Knobles, D. P., & Dahl, P. H. (2020). Seabed and range estimation of impulsive time series using a convolutional neural network. *The Journal of the Acoustical Society of America, 147*(5), 403–408.
21. Sagheer, A., & Kotb, M. (2019). Time series forecasting of petroleum production using deep lstm recurrent networks. *Neurocomputing, 323*, 203–213.
22. Ngamakeur, K., Yongchareon, S., Yu, J., & Sheng, Q. Z. (2022). Deep cnn-lstm network for indoor location estimation using analog signals of passive infrared sensors. *IEEE Internet of Things Journal, 9*(22), 22582–22594.
23. Wu, W., Shen, L., Zhao, Z., Li, M., & Huang, G. Q. (2022). Industrial iot and long short-term memory network-enabled genetic indoor-tracking for factory logistics. *IEEE Transactions on Industrial Informatics, 18*(11), 7537–7548.
24. Mahjoub, T., Mnaouer, A. B., Said, M. B., & Boujemaa, H. (2024). Lora signal propagation and path loss prediction in tunisian date palm oases. *Computers and Electronics in Agriculture, 222*, Article 109027.
25. Lima, W. G., Lopes, A. V., Cardoso, C. M., Araújo, J. P., Neto, M. C., Tostes, M. E., Nascimento, A. A., Rodriguez, M., & Barros, F. J. (2024). Lora technology propagation models for iot network planning in the amazon regions. *Sensors, 24*(5), 1621.
26. Krishnan, S., & Mendoza Santos, R.X. (2021). Real-time asset tracking for smart manufacturing. Implementing Industry 4.0: The Model Factory as the Key Enabler for the Future of Manufacturing, 25–53 .
27. Vo, Q. D., & De, P. (2015). A survey of fingerprint-based outdoor localization. *IEEE Communications Surveys & Tutorials, 18*(1), 491–506.
28. Cao, X., Zhuang, Y., Yang, X., Sun, X., & Wang, X. (2021). A universal wi-fi fingerprint localization method based on machine learning and sample differences. *Satellite Navigation, 2*, 1–15.
29. Chen, Z., Xia, F., Huang, T., Bu, F., & Wang, H. (2013). A localization method for the internet of things. *The Journal of Supercomputing, 63*, 657–674.
30. Tarekegn, G.B., Lin, H.-P., Adege, A.B., Munaye, Y.Y., & Jeng, S.-S. (2019). Applying long short-term memory (lstm) mechanisms for fingerprinting outdoor positioning in hybrid networks. In: 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall), pp. 1–5 . IEEE.
31. Anjum, M., Khan, M.A., Hassan, S.A., Mahmood, A., & Gidlund, M. (2019). Analysis of rssi fingerprinting in lora networks. In: 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), pp. 1178–1183 . IEEE.
32. Kufakunesu, R., Hancke, G. P., & Abu-Mahfouz, A. M. (2020). A survey on adaptive data rate optimization in lorawan: Recent solutions and major challenges. *Sensors, 20*(18), 5044.
33. Carrino, F., Janka, A., Abou Khaled, O., & Mugellini, E. (2019). Loraloc: Machine learning-based fingerprinting for outdoor geolocation using lora. In: 2019 6th Swiss Conference on Data Science (SDS), pp. 82–86 . IEEE.
34. Zakaria, Y., & Ivanek, L. (2018). Analysis of channel propagation models based on calculated and measured data. In: 2018 ELEKTRO, pp. 1–5 . IEEE.
35. Linka, H., Rademacher, M., Aliu, O.G., & Jonas, K. (2018). Path loss models for low-power wide-area networks: Experimental results using lora .
36. Jörke, P., Böcker, S., Liedmann, F., & Wietfeld, C. (2017). Urban channel models for smart city iot-networks based on empirical measurements of lora-links at 433 and 868 mhz. In: 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), pp. 1–6 . IEEE.
37. Harinda, E., Hosseinzadeh, S., Larijani, H., & Gibson, R.M. (2019). Comparative performance analysis of empirical propagation models for lorawan 868mhz in an urban scenario. In: 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), pp. 154–159 . IEEE.
38. ElSabaa, A., Guéniat, F., Wu, W., & Ward, M. (2022). Enhanced data-driven lora lp-wan channel model in birmingham. In: 2022 IEEE World AI IoT Congress (AIIoT), pp. 766–772 . IEEE.
39. Van Houdt, G., Mosquera, C., & Nápoles, G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review, 53*(8), 5929–5955.
40. Selmy, H. A., Mohamed, H. K., & Medhat, W. (2024). A predictive analytics framework for sensor data using time series and deep learning techniques. *Neural Computing and Applications, 36*(11), 1–14.
41. Srivastava, A., & Anto, S. (2022). Weather prediction using lstm neural networks. In: 2022 IEEE 7th International Conference for Convergence in Technology (I2CT), pp. 1–4 . IEEE.
42. Han, C., Park, H., Kim, Y., & Gim, G. (2022). Hybrid cnn-lstm based time series data prediction model study. In: IEEE/ACIS International Conference on Big Data, Cloud Computing, and Data Science Engineering, pp. 43–54 . Springer.
43. Lara-Benítez, P., Gallego-Ledesma, L., Carranza-García, M., Luna-& Romera, J.M. (2021). Evaluation of the transformer architecture for univariate time series forecasting. In: Advances in Artificial Intelligence: 19th Conference of the Spanish Association for Artificial Intelligence, CAEPIA 2020/2021, Málaga, Spain, September 22–24, 2021, Proceedings 19, pp. 106–115 . Springer.
44. Abubakr, T., & Nasr, O.A. (2023). Novel lstm-based approaches for enhancing outdoor localization accuracy in 4g networks. IEEE Access .
45. Technology, M. (2015–2020). Low-Power Long Range LoRa®Technology Transceiver Module. Microchip Technology, Microchip Technology. https://ww1.microchip.com/downloads/en/DeviceDoc/RN2483-Data-Sheet-DS50002346E.pdf.
46. Mackey, A., & Spachos, P. (2019). Lora-based localization system for emergency services in gps-less environments. In: IEEE

INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 939–944 . IEEE.
47. Lagarias, J. C., Reeds, J. A., Wright, M. H., & Wright, P. E. (1998). Convergence properties of the nelder-mead simplex method in low dimensions. *SIAM Journal on optimization,9*(1), 112–147.
48. Leys, C., Ley, C., Klein, O., Bernard, P., & Licata, L. (2013). Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology, 49*(4), 764–766.
49. Hu, L., Bao, X., & Wang, Q. (2011). The repetition principle in scientific research. *Zhong xi yi jie he xue bao=Journal of Chinese Integrative Medicine, 9*(9), 937–940.
50. Virmani, D., Taneja, S., & Malhotra, G. (2015). Normalization based k means clustering algorithm. arXiv:1503.00900 .
51. Rizk, H., Shokry, A., & Youssef, M. (2019). Effectiveness of data augmentation in cellular-based localization using deep learning. In: 2019 IEEE Wireless Communications and Networking Conference (WCNC), pp. 1–6 . IEEE.
52. Iwana, B. K., & Uchida, S. (2021). An empirical survey of data augmentation for time series classification with neural networks. *Plos one, 16*(7), 0254841.
53. Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing, 45*(11), 2673–2681.
54. Tan, K. L., Lee, C. P., Anbananthen, K. S. M., & Lim, K. M. (2022). Roberta-lstm: a hybrid model for sentiment analysis with transformer and recurrent neural network. *IEEE Access, 10*, 21517–21525.
55. Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2016). Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems, 28*(10), 2222–2232.
56. Shiri, F.M., Perumal, T., Mustapha, N., & Mohamed, R. (2023). A comprehensive overview and comparative analysis on deep learning models: Cnn, rnn, lstm, gru. arXiv:2305.17473 .
57. Qin, L., Yu, N., & Zhao, D. (2018). Applying the convolutional neural network deep learning technology to behavioural recognition in intelligent video. *Tehnički vjesnik, 25*(2), 528–535.
58. Gamboa, J.C.B. (2017). Deep learning for time-series analysis.arXiv:1701.01887 .
59. Längkvist, M. (2014). Modeling time-series with deep networks. PhD thesis, Örebro university .

AlaaAllah ElSabaa is currently pursuing a PhD in Engineering at Birmingham City University (BCU), UK. Her research focuses on LoRaWAN networks, wireless propagation modeling, and the application of machine learning techniques for time-series data in localization systems. She received her BSc (Hons) in Electronics and Communications Engineering in 2014 from the Arab Academy for Science and Technology and Maritime Transport (AASTMT), Egypt. From 2014 to 2018, she worked as a full-time teaching assistant at AASTMT. In 2017, she was awarded an Erasmus scholarship to spend a semester at Staffordshire University, UK. She has authored publications in wireless communications and IoT systems and has participated in international research exchange and summer training programs, including programs in Canada.

Florimond Guéniat received the Ph.D. degree in Physics from Paris Saclay University, France, in 2013. Following postdoctoral appointments in the department of Mathematics at Florida State University and in the mechanical engineering department at the University of Illinois at Urbana-Champaign, he joined Birmingham City University, Birmingham, U.K., where he is currently an Associate Professor. He was previously the Academic Lead for Mechanical and Automotive Engineering. His research interests span complex systems control, microgrid optimization, and artificial intelligence, mainly applied to energy conservation in transportation and energy production. Dr. Gueniat has authored over 50 papers in international journals and conference proceedings. He is the Co-PI and technical lead of the Ayrton funded SMART-SIP+ project, and is actively involved in other key technology initiatives including ArtpackAI. Furthermore, he serves as a committee member for the Artificial Intelligence Technical Network of the Institution of Engineering and Technology (IET).

Wenyan Wu is currently a Professor in Smart Sensor and Advanced System Engineering and the Chair in the research group of Sensor and Control with Birmingham City University, Birmingham, U.K. She received the B.Eng. and M.Eng. degrees in Electronic Engineering from Dalian University of Technology, China, obtained Ph.D. degree in water quality modelling and optimization in water distribution systems from the Harbin Institute of Technology, Harbin, China, and gained the second Ph.D. degree in Reconfigurable Virtual environment Design for virtual testing from the University of Derby, Derby, U.K. She was a Principal Investigator (PI) and a Project Coordinator for EU FP7-WatERP, EU FP7- SmartWater, and EU Horizon 2020-IoT4Win. She has extensive research experiences in smart sensors and sensor networks, the IoT, intelligent monitoring and Artificial intelligence, digital design and processing, data visualization, modelling and optimization in water distribution system, and water resource management. She has authored or coauthored more than 150 peered reviewed journal articles and conference papers. She was a Professor in Digital Design and Technologies with Staffordshire University, U.K. a Senior Lecturer in Computing, Harbin Institute of Technology, China and a Research Fellow in Water Software Systems, De Montfort University, U.K., and the National Key CAD Laboratory, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. Prof. Wu organized special issue in IEEE journals and regular IEEE and Springer Journals reviewer and was member of IEEE CIS and IEEE sensor council.