future internet

MDPI

Article

Hybrid B5G-DTN Architecture with Federated Learning for Contextual Communication Offloading

Manuel Jesús-Azabal [1,*], Meichun Zheng [1] and Vasco N. G. J. Soares [2,3,4,*]

[1] School of Information Engineering, Wenzhou Business College, Wenzhou 325015, China; 20249054@wzbc.edu.cn
[2] Department of Computer Science, Polytechnic University of Castelo Branco, Av. Pedro Álvares Cabral, n° 12, 6000-084 Castelo Branco, Portugal
[3] Instituto de Telecomunicações, Rua Marquês d'Ávila e Bolama, 6201-001 Covilha, Portugal
[4] AMA—Agência para a Modernização Administrativa, Rua de Santa Marta, n° 55, 1150-294 Lisboa, Portugal
* Correspondence: manuel@wzbc.edu.cn (M.J.-A.); vasco.g.soares@ipcb.pt (V.N.G.J.S.)

Abstract

In dense urban environments and large-scale events, Internet infrastructure often becomes overloaded due to high communication demand. Many of these communications are local and short-lived, exchanged between users in close proximity but still relying on global infrastructure, leading to unnecessary network stress. In this context, delay-tolerant networks (DTNs) offer an alternative by enabling device-to-device (D2D) communication without requiring constant connectivity. However, DTNs face significant challenges in routing due to unpredictable node mobility and intermittent contacts, making reliable delivery difficult. Considering these challenges, this paper presents a hybrid Beyond 5G (B5G) DTN architecture to provide private context-aware routing in dense scenarios. In this proposal, dynamic contextual notifications are shared among relevant local nodes, combining federated learning (FL) and edge artificial intelligence (AI) to estimate the optimal relay paths based on variables such as mobility patterns and contact history. To keep the local FL models updated with the evolving context, edge nodes, integrated as part of the B5G architecture, act as coordinating entities for model aggregation and redistribution. The proposed architecture has been implemented and evaluated in simulation testbeds, studying its performance and sensibility to the node density in a realistic scenario. In high-density scenarios, the architecture outperforms state-of-the-art routing schemes, achieving an average delivery probability of 77%, with limited latency and overhead, demonstrating relevant technical viability.

Keywords: Beyond 5G; delay-tolerant networks; federated learning; communication offloading; edge AI

check for updates

Academic Editor: Paolo Bellavista

Received: 21 July 2025
Revised: 20 August 2025
Accepted: 26 August 2025
Published: 29 August 2025

Citation: Jesús-Azabal, M.; Zheng, M.; Soares, V.N.G.J. Hybrid B5G-DTN Architecture with Federated Learning for Contextual Communication Offloading. *Future Internet* **2025**, *17*, 392. https://doi.org/10.3390/fi17090392

1. Introduction

The rapid proliferation of mobile applications and connected services has led to an unprecedented increase in Internet traffic [1]. Services such as high-definition video streaming, real-time location-based updates, augmented reality guides, and interactive tourist applications now dominate mobile bandwidth consumption [1]. This trend is further driven by user-centric and time-sensitive services that depend heavily on cloud-based infrastructures for content retrieval and interaction.

This surge in demand places considerable strain on communication infrastructure. In high-density areas such as stadiums, transportation hubs, tourist landmarks, and cultural festivals, cellular networks frequently experience congestion, resulting in high latency, packet loss, and degraded quality of service [2,3]. These limitations persist even with the deployment of 5G technologies, particularly when thousands of users attempt to connect simultaneously within a confined geographic area [3].

A notable and under-addressed characteristic of many of these communications is their locality and short-term relevance. For example, information shared between people within the same crowd, such as route changes, local event updates, or ephemeral multimedia, often pertains only to those physically present at that moment [3]. However, these communications are still commonly routed through global infrastructure and cloud servers, introducing unnecessary overhead and increasing dependency on centralized systems [3].

This architectural mismatch between global routing and local communications highlights the need for decentralized, context-aware alternatives that can offload traffic, preserve privacy, and improve responsiveness in dense environments. In this context, Delay Tolerant Networks (DTNs) emerge as a promising paradigm for enabling communication in local environments, operating with limited or intermittent connectivity [4–6]. By relying on opportunistic contacts and a store-carry-forward model, DTNs are well suited to support local communication in urban hotspots and dense crowd scenarios, offering an alternative to alleviate infrastructure congestion [6]. Their decentralized nature and minimal reliance on continuous connectivity make them attractive for offloading traffic from the core Internet infrastructure, especially for local interactions.

Communication in DTNs relies on the asynchronous contact of nodes to transmit information, using dynamic routing [6]. However, despite their potential, the performance of a DTN depends heavily on the ability to select appropriate relay nodes, which is complicated by dynamic mobility patterns, unpredictable contact timings, heterogeneous node behaviors, and variable resource constraints such as battery life or buffer availability [3,6,7]. These challenges require the adoption of strategies to mitigate the effects of these limiting factors. In this regard, the integration of new approaches such as Beyond 5G (B5G) emerges as a promising solution.

The development of B5G architectures emphasizes a shift toward intelligent, distributed, and context-aware communication models [8]. B5G envisions the integration of heterogeneous networks, edge intelligence, and human-centric design to support responsive systems in complex environments [8]. These systems must learn from local interactions, adapt to contextual changes, and make autonomous decisions without compromising privacy or depending on continuous backhaul connectivity [8].

Recent work has already brought several B5G enablers to local communication, including edge computing for on-site analytics that align with distributed, context-aware design [8]; device-to-device (D2D) links for short-range, low-latency proximity services [9]; local breakout via the User Plane Function (UPF) to steer traffic toward edge applications; and edge AI/ML for adaptive behavior [10]. Related efforts have also explored Federated Learning (FL) in DTNs, notably for proactive content caching at the wireless edge [11,12] and for FL-powered routing in opportunistic environments [13]. Nevertheless, gaps remain for dense, ephemeral exchanges: many systems still depend on continuous backhaul connectivity or 5G base-station scheduling and degrade under congestion [8,14]; D2D operation commonly assumes base-station-assisted resource allocation, limiting autonomy in partially connected or overloaded scenarios [15]; and learning-based routing often centralizes training, requiring raw data aggregation with privacy and bandwidth costs, and is misaligned with DTN intermittency [16,17].

Given this scenario, a gap persists between the theoretical potential of DTNs for local, ephemeral exchanges and the practical demands of B5G ecosystems, where congestion, intermittency, and privacy must be handled jointly. We aim to bridge this gap by keeping forwarding local and opportunistic while using edge resources only for lightweight coordination that does not rely on backhaul or centralized data collection.

DTN routing needs a model that can be trained from local encounters, tolerate intermittent connectivity, and avoid sharing raw data. FL satisfies these constraints: training happens on-device, and only model weights are exchanged during brief contacts with the serving smart pole; aggregation at the pole provides a crowd-level prior that shortens warm-up and reduces flooding; periodic aggregation tracks crowd drift without centralized logs; and privacy is preserved because encounter vectors never leave devices [18–21]. In contrast, alternative techniques are less suitable in this setting: centralized ML requires continuous backhaul and raw data collection [20]; purely local learning lacks knowledge transfer and converges slowly [21]; and split learning assumes stable links between clients and a coordinating server [22].

Building on this, this paper presents a hybrid B5G–DTN architecture for contextual offloading in dense scenarios. DTN forwarding remains D2D. Smart poles act as edge coordinators: they aggregate model updates and dispatch delivery feedback without collecting raw encounter data. We design a destination-agnostic FL routing mechanism. Each node runs a lightweight regression model that scores encounters from contact-level features. Labels are obtained retrospectively from delivery paths reported at the edge. Models are aggregated with the standard FedAvg algorithm. We implement a proof of concept by integrating The ONE simulator with an external Python 3.13.7 FL core via a REST interface, linking per-node models with the aggregated model. We evaluate the approach in a Macau district scenario, varying density to assess performance against state-of-the-art routing.

Considering the described content, the paper is organized as follows. The Section 1 explains in detail the relevance of local communications and their impact on Internet infrastructure. Then, the Section 2 summarizes the most relevant approaches aimed at enhancing local transmissions, addressing the role of FL and B5G in local contexts. After that, the architecture is described in detail, including an overview of the proposal, the components of the architecture, and the normalization of the FL model. Next, the Section 4 presents the implemented proof of concept, the testbed scenarios, and the obtained results. Finally, the Section 5 provides the main findings of the proposal and outlines future work.

2. Related Work

The proposed work presents an architecture to support ephemeral communications in local contexts, where all entities involved are physically present in the environment. For this purpose, disciplines such as DTN, B5G, and FL are combined to explore their potential for local communications. Next, the most relevant approaches within this scope are discussed.

2.1. Paradigms for Local Communications

In scenarios characterized by high user density, limited infrastructure availability, and short-lived information needs, several communication paradigms can support local exchanges [23]. These include infrastructure-based networking [8], peer-to-peer (P2P) overlays [24], D2D communication [9], and DTNs [7].

Infrastructure-based approaches, such as cloud messaging or centralized publish and subscribe systems, offer efficient message delivery for local interchanges [8]. However, works such as [25] propose alternatives that become highly dependent on stable backhaul connections and centralized resources. In crowded environments, such as festivals or tourist zones, they often suffer from congestion and fail to meet real-time requirements for local dissemination [26,27].

P2P networks enable localized exchanges in a variety of contexts. Works such as [28] successfully explore the role of peer-to-peer connections for local energy management. Other proposals extend the application of these networks to dynamic scenarios, such as coordinating autonomous vehicles [29]. However, P2P networks often assume continuous connectivity and suffer from low routing efficiency under mobility [24]. Works such as [30] present a system designed for unstructured P2P over ad hoc networks, highlighting how peer mobility, battery constraints, and network dynamics significantly impair routing effectiveness. Similarly, D2D communication in 5G provides low latency, short-range links between nearby devices [31]. Studies such as [9] have achieved notable progress in this area, demonstrating successful integration of D2D communication in demanding local environments. However, there remains a strong dependence on base station coordination [14], which limits its autonomy in partially connected or overloaded scenarios [15].

In the case of DTNs, they operate under the assumption of intermittent connectivity and leverage the store-carry-forward paradigm to transfer data between nodes through physical encounters [8,32]. This approach aligns well with local, contextual communications in urban or crowd-based settings, where information relevance is both location and time bound, helping to avoid infrastructure saturation [33]. Works such as [34,35] demonstrate the potential of DTNs in isolated rural areas, including use cases like video streaming [36,37]. DTNs have also been evaluated in real-life testbeds [6,38], showing promising performance. Moreover, studies such as [5,39] have successfully integrated these networks with demanding paradigms such as Digital Twins. Considering this, DTNs emerge as a suitable alternative for local communications, although they still face routing challenges due to mobility uncertainty [32,40]. In this regard, the application of ML techniques to routing decisions represents a significant advancement.

2.2. ML in Local Networking

The use of ML in networking has gained considerable attention, particularly for routing in mobile and dynamic environments [17]. In the context of DTNs, several studies have explored centralized ML-based routing, where a central server collects mobility or contact data from nodes and trains a model to predict suitable relays [16,17,41]. These models typically employ supervised or reinforcement learning to capture patterns in node encounters, buffer usage, or delivery success rates.

While centralized ML approaches can yield improved performance over heuristic-based methods, they face critical limitations [42]. First, they require continuous data collection and transmission to a central controller, introducing significant privacy concerns and bandwidth overhead [16]. Second, they suffer from generalization issues: a model trained in one scenario may perform poorly in another environment with different mobility dynamics [41]. Finally, centralized solutions represent a single point of failure and are incompatible with the disconnected and distributed nature of DTNs. Considering this, FL emerges as a privacy-preserving alternative.

FL allows model training to be carried out locally on devices without transferring raw data. In networking, FL has been applied to mobile edge computing, Internet of Things (IoT) environments, and vehicular networks, where devices collaboratively train models while relying on edge nodes for aggregation [10]. The main advantages of FL include privacy preservation, as local data remains on the device; adaptability to local context, allowing per-device personalization; and scalability, since training is distributed and does not depend on a constant connection with a central controller.

Recent advances improve the communication efficiency of FL through compression and server-side optimization. Distillation with quantization reduces uplink payloads while preserving accuracy. Works such as FedDT combine knowledge distillation with ternary compression [43]. Furthermore, other works focus on server-centric sharpness optimization, cutting client communication by exploiting global gradients, such as FedGloSS [44]. These techniques are complementary to our design: the edge poles that coordinate aggregation can incorporate compressed updates or sharpness-aware routines to shrink air-interface costs in dense deployments.

Personalization for heterogeneous devices has also progressed, especially for personalized IoT. Cedar integrates FL with layer-wise adaptive uploads to deliver domain-adaptive, privacy-preserving personalization across personalized IoT tasks [45]. Closer to classic FL, dynamic-weight aggregation tailors global models to per-client data without sharing raw samples [46]. These mechanisms become disruptive techniques that enhance the coordination at the edge. In the case of our proposal, these methods enable keeping all encounter data on devices, aligning with the DTN setting's privacy and intermittency needs.

Furthermore, several works optimize the client–server interaction layer itself. Information-centric networking has been shown to accelerate FL exchanges versus host-centric transports and to improve the robustness of model dissemination [47]. Modular, resilient edge frameworks report lower overheads with protocols such as Zenoh and maintain convergence under worker churn [48]. Mobility and resource-aware client selection improve round completion and stability in dynamic IoT [49]. These directions fit naturally with B5G strategies as local coordinators in crowded areas.

In addition to these contributions, prior work has already explored FL within DTN. Early efforts studied FL-based proactive caching at the edge, where models guide which content to pre-position near users [12]. Building on this direction, [11] propose a feedback delay-tolerant proactive caching scheme that uses federated training at the wireless edge to update caching decisions under intermittent connectivity. Closer to forwarding, the work [50] investigates FL-based routing for opportunistic network environments, showing that collaborative on-device training can inform relay selection without centralizing raw contact data. Our work complements these studies but targets a different axis: we use a destination-agnostic, contact-level regression that scores each encounter for relay usefulness; we obtain labels retrospectively from delivery paths reported at the edge; and we keep D2D forwarding autonomous, using B5G smart poles only for short model uploads/downloads and aggregation. This design fits dense, ephemeral exchanges while preserving privacy and minimizing backhaul dependence.

Considering this, FL fits as a router enhancer for DTNs. However, in spite of its distributed nature, the aggregation of the models requires a coordination with a core entity in charge of handling the addition of new nodes and their training. For this, B5G becomes a potential middleware to combine FL with DTNs.

2.3. Possibilities of B5G in Local Communications

The evolution toward B5G networks introduces new architectural and functional paradigms that are particularly well suited for supporting localized, contextual, and ephemeral communications. Central to B5G's potential in local communication is the deployment of edge computing nodes in the form of smart poles or street cabinets. These nodes not only provide coverage and routing functions but also offer on-site computing capabilities, enabling context-aware services such as real-time processing and localized alert dissemination.

This paper situates its proposed architecture within the B5G vision, leveraging static edge nodes as FL coordinators and using mobile devices as opportunistic nodes to consume local information.

2.4. Contributions

Despite the promise of ML and FL in communication systems, significant gaps remain—particularly in applying FL to routing for ephemeral communications and B5G-localized services. To address these gaps, this manuscript proposes a hybrid B5G–DTN architecture with FL-based routing. The system enables local, context-aware communication offloading while preserving privacy and adapting to mobility dynamics. Our specific contributions are:

- Hybrid B5G–DTN architecture. We preserve D2D forwarding and use smart poles solely to aggregate federated model updates and to dispatch binary delivery feedback, avoiding uploads of raw encounter data.
- Routing mechanism. A lightweight, destination-agnostic FL scheme that scores encounters via regression on contact-level features; labels are derived retrospectively from successful delivery paths; aggregation uses FedAvg.
- Implementation. A proof-of-concept integrating The ONE simulator with an external Python FL core via a REST API, synchronizing per-node models with the aggregated model.
- Evaluation. An empirical study in a realistic Macau tourist district scenario against state-of-the-art routing protocols, reporting delivery probability (up to 92%, average 77%), latency, overhead, hop count, and FL model metrics.

Considering the contributions of the proposed work, the next section provides the details of the proposal.

3. Hybrid B5G-DTN Architecture for Contextual Communication Offloading

3.1. System Overview

As Figure 1 shows, the architecture identifies two main components: (1) the DTN, which includes the set of dynamic nodes in the scenario, such as senders, intermediate nodes, and destinations; and (2) the B5G pole, which hosts the edge computing node responsible for aggregating the FL model.

- DTN. During crowded scenarios, the DTN comprises the set of devices involved in local communications. In this way, it is possible to identify four distinct roles:
  - Sender node. Responsible for generating punctual local information and transmitting it to the relevant destination at the current moment. To do this, the sender can define criteria to select destination nodes within the crowd, such as their intended path, in order to notify them about route changes.
  - Destination node. These are the devices identified as relevant for receiving the information. They are dynamically selected based on a specific filter defined by the sender, such as their direction of travel. This allows them to receive punctual contextual communications (e.g., notifications about road closures, contextual advertisements, or changes in the metro schedule).
  - Intermediate nodes. These devices act as relays for information between the sender and the destination. They are dynamically identified as non-destination nodes because they do not meet the filter criteria set by the sender. Therefore, they can serve as intermediate nodes, forwarding information using the FL model integrated into their device. For this purpose, information such as previous contacts and the destination path is considered to estimate a relevance score for data forwarding.

   – Provisional intermediate nodes. These nodes are not classified as destination or intermediate nodes due to the low reliability of their destination path estimation. As a result, the architecture considers them provisional intermediate nodes.

- B5G pole. This infrastructure is responsible for running the edge node that aggregates the weights of the FL models executed by the network nodes. To achieve this, devices communicate with the pole to stay updated with the latest model, while the pole also keeps track of the most relevant events in the network. It becomes a critical element for maintaining model updates and enhancing the training of FL models. To support this process, the edge node feeds the DTN with communication outcomes by tagging whether messages were delivered to their destinations or not, enabling autonomous training of the models on each node.

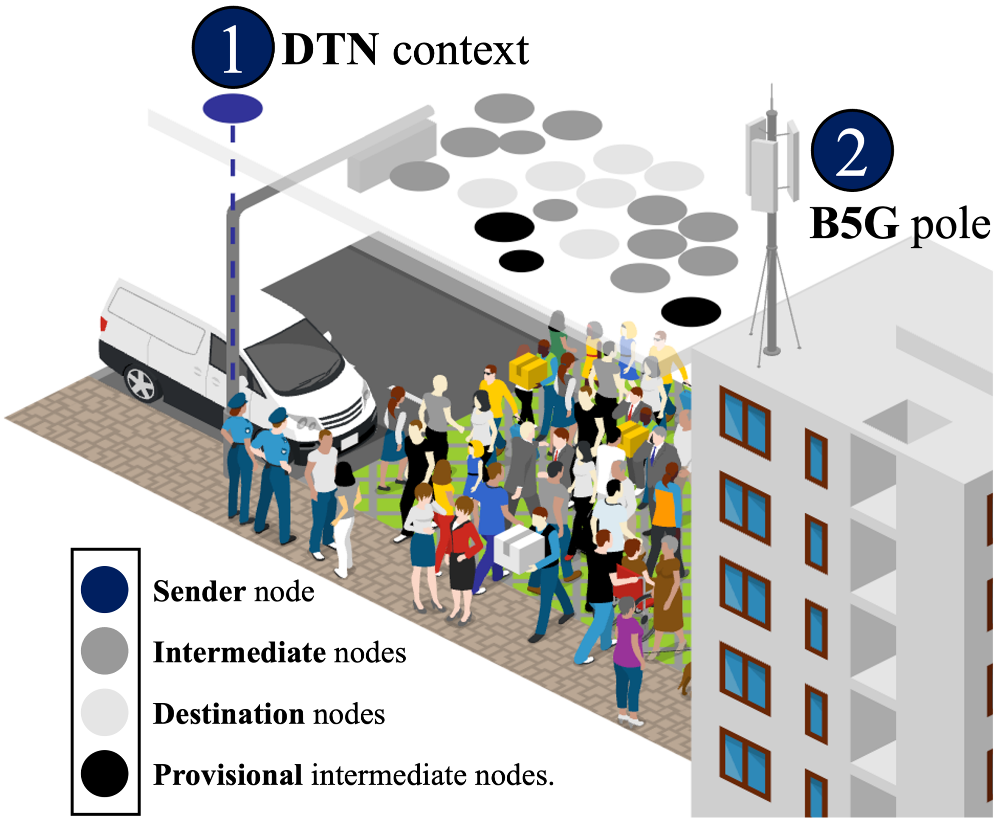


Figure 1. Overview of the B5G-DTN architecture.

This approach enables the transmission of punctual communications to relevant destinations in local contexts, avoiding the involvement of the entire communication infrastructure stack. Considering the different steps involved in the architecture, the next subsection outlines the operational workflow of the proposal.

3.2. Operational Workflow

The architecture is based on a set of phases that enable the application of FL to the routing process of contextual information in a local setting. To achieve this, the model must undergo training, supported by the intervention of the edge node running on the B5G pole. The phases are described below.

- Phase 0. Initialization. This phase begins when mobile devices enter the coverage area of the B5G smart pole, becoming nodes in the DTN. To enable communication, each node receives an anonymous identifier. If the pole has a global FL model available, nodes load it locally and are ready to use it for routing. However, if the pole does not have a global model, nodes operate without one.

- Phase 1. Message creation and role assignment. Once nodes are present in the local context, a relevant profile (e.g., a verified source or official institution) acts as the sender node, generating a contextual message. This message includes both content and a receiver filter based on destination segmentation (e.g., nodes heading toward a landmark). As a result, four groups of nodes are identified: the sender, as the origin of the message; destination nodes, which meet the filter criteria; intermediate nodes, which clearly do not meet the criteria; and provisional intermediate nodes, whose destination is unknown and therefore cannot be confidently classified.
- Phase 2. Blind dissemination and data gathering. After generating the message and segmenting the nodes, opportunistic routing begins. Initially, as the nodes lack a trained model, the message is disseminated indiscriminately, flooding the network. Meanwhile, each node logs contact information locally, including the identifier of the encountered node, timestamp, contact duration, mobility vector, buffer state, battery level, and whether the destination matched. To support this, nodes exchange context vectors at each contact, forming the basis for local FL training data. Since delivery success is not yet known, these samples remain unlabeled.
- Phase 3. Retrospective labeling and local FL training. When a destination node receives the message, it sends a delivery report to the B5G pole, including the anonymized relay path with a list of node identifiers and the message identifier. The B5G pole then notifies each node on the path with a success label for that contact/message pair. Nodes not involved in the delivery path provisionally label their related encounters as failed. In this way, each node builds a labeled dataset of context features and associated success or failure labels. Once enough examples are collected, local training begins.
- Phase 4. FL-based routing begins. After training their local model, nodes begin using it to score future encounters with provisional or intermediate nodes. Messages are forwarded only if the estimated score exceeds a predefined threshold (e.g., 0.7). If a destination node is identified, the message is forwarded directly, bypassing the model. Meanwhile, nodes track whether their predictions match the actual delivery outcomes, as later reported by the B5G pole. This enables nodes to monitor the accuracy of their models.
- Phase 5. Model aggregation and dissemination. Once a node's model reaches a sufficient accuracy threshold, it transmits its model weights to the B5G pole. The edge node on the pole performs federated aggregation to produce an updated global model. This global model is then distributed to nodes currently connected to the B5G pole, as well as to newly arriving nodes, allowing them to begin with a pretrained model rather than training from scratch.

In this workflow, the architecture combines DTNs with B5G and leverages FL for routing. The FL models compute a local relay score for each encounter, enabling nodes to share information directly without relying on intermediate infrastructure. B5G smart poles act as edge nodes that facilitate effective training of local models rather than serving as intermediate nodes for cloud-based communication. Consequently, transmissions that might otherwise involve remote servers are executed locally when they are relevant to a specific moment and place, shifting traffic away from the Internet toward local links. Each B5G pole coordinates FL training and aggregation, providing nodes with delivery labels to enrich individual records for supervised learning. After local training, nodes briefly contact an edge node to upload weights; the pole produces an aggregated model for nodes to download. As a result, forwarding decisions are made locally and selectively, redundant copies are reduced, and airtime shrinks—offloading the infrastructure. Crucially, nodes continue exchanging messages autonomously even when a pole is temporarily unreachable; they resynchronize the model at the next brief contact.

The next subsection details each architectural component, its role, and its responsibilities.

3.3. Architecture Components

The architecture presents a hybrid B5G-DTN system in which mobile DTN nodes interact locally through wireless interfaces. These devices periodically synchronize with a B5G pole via 5G for FL coordination and feedback exchange. As shown in Figure 2, each device includes a set of internal components that communicate for different purposes.

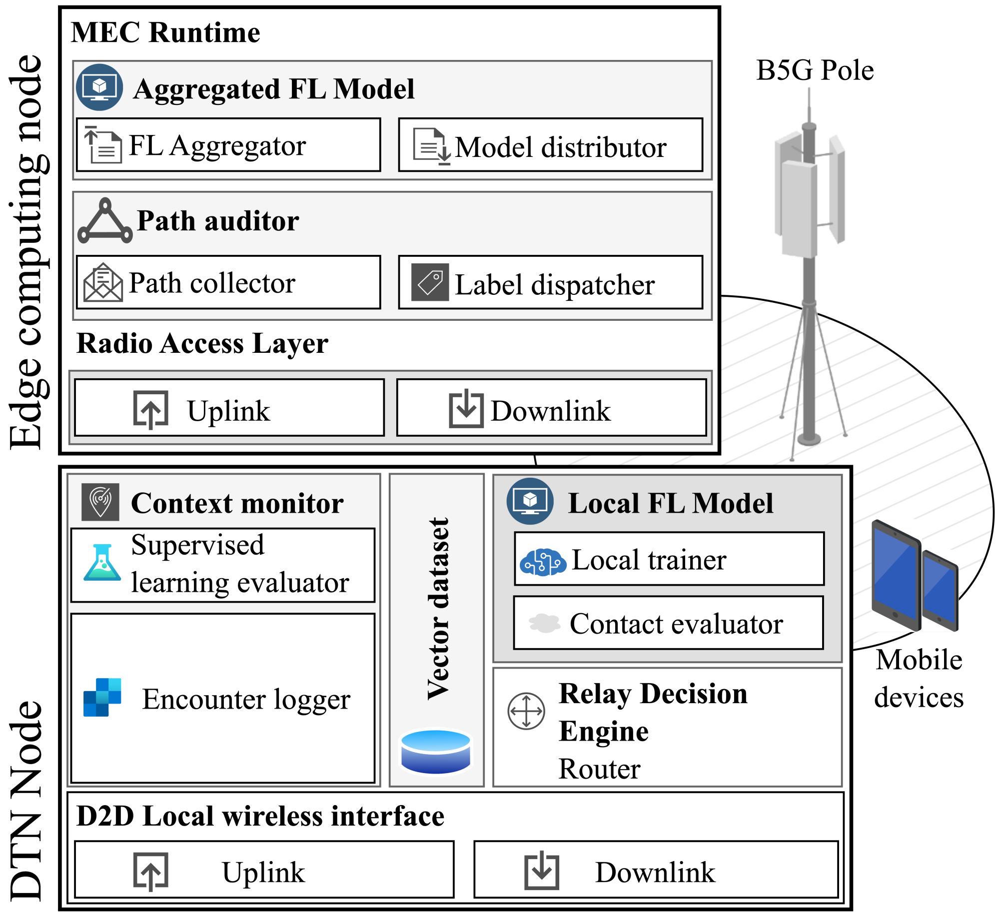


Figure 2. Architecture components in the hybrid B5G-DTN approach.

3.3.1. Edge Computing Node

The edge computing node runs on a B5G pole. This component serves as the central aggregation and feedback point for the federated system, equipped with 5G interfaces and edge computing capabilities. The relevant components are described below:

- Mobile Edge Computing (MEC) Runtime. This layer contains the software environment running on a B5G pole. It is responsible for executing the aggregated FL model and the path auditor, providing their services through an Application Programming Interface (API).
    - Aggregated FL Model. This component forms the core of the federated system, providing aggregation and distribution functions. The FL aggregator receives updated model weights from mobile nodes and performs the aggregation process using a customized technique (e.g., FedAvg). In addition, the model distributor sends the aggregated model back to participating nodes, enabling them to improve their local inference capabilities.

- Path Auditor. This component supports the local topology by maintaining updated information about communication performance. The path collector receives details about successful message delivery paths, including the identifiers of the involved nodes. The label dispatcher then notifies those nodes about successful deliveries. Together, these functions allow tagging of interactions to support supervised learning in mobile nodes.

- Radio Access Layer. This layer handles the physical communication between the pole and mobile nodes. The 5G interface manages uplink communication from DTN nodes to the pole, as well as downlink communication from the pole to the DTN nodes.
- Control and UPF. This component is responsible for forwarding user traffic to the edge application. Specifically, the UPF component routes selected traffic from DTN nodes to the local MEC node, which runs the FL model and the path auditor. This layer also manages IP addressing and DNS processes, enabling the execution of services deployed within the local environment of the pole.

3.3.2. DTN Nodes

Along with the B5G pole, the mobile devices within its coverage area rely on a component stack to operate within the proposed architecture. The components of the DTN nodes are described below:

- Context Monitor. This component runs on mobile devices acting as DTN nodes in the local environment. It uses the local wireless interface to gather information about the node's physical encounters and is also responsible for notifying the edge node about message receptions.
  - Encounter Logger. This module monitors the interactions over the local interface, collecting details about contacts, including message receptions. It keeps the vector dataset updated with new interaction records.
  - Supervised Learning Evaluator. This module communicates with the B5G pole to report message receptions, including their routing paths. This allows the edge node to remain informed about significant encounters, while intermediate nodes receive feedback to label their interactions as effective or not, depending on their role in the message delivery.
- Vector Dataset. This internal database stores information about node interactions and contextual variables. The received vectors from encountered nodes are saved and labeled according to their contribution to successful message delivery.
- Local FL Model. This layer represents the FL model running on the device. To keep it updated and applicable for routing decisions, it includes two main components: the local trainer and the contact evaluator.
  - Local Trainer. This component handles training operations for the FL model. Using the received weights from the B5G pole and the device's internal dataset, it maintains the model's accuracy for encounter evaluation.
  - Contact Evaluator. This module processes the variables from the training dataset and outputs a continuous score for each encounter. The resulting value, ranging from 0 to 1, is used in the routing process to assess the suitability of potential intermediate nodes.
- Relay Decision Engine. This component is responsible for defining the routing logic in the local context. Based on the local model, it can use the contact evaluator to predict encounter scores and select the next hop accordingly. A custom threshold can be configured for this decision.

- D2D Local Wireless Interface. This component represents the physical communication layer for local data exchange. For D2D transmissions, limited-range technologies such as Bluetooth, Bluetooth Low Energy (BLE), or WiFi may be used. Two main data flows are supported: uplink, to receive data vectors from encountered nodes, and downlink, to transmit the node's contextual variables.

3.3.3. Communication Links

Considering the components described in the architecture, a set of communications is required to operate the FL model and apply it to support local transmissions. Table 1 presents these communication flows.

Table 1. Communications involved in the proposed architecture.

Source Component	Target Component	Interface Type	Data Transmitted	Purpose
Encounter Logger (DTN Node)	Local FL Trainer (same node)	Internal (device)	Context vector	Collect and prepare data for local model training
Local FL Model (DTN Node)	FL Aggregator (MEC Runtime)	5G Uplink	Local model weights	Send weights for federated aggregation at the edge node
Model Distributor (MEC Runtime)	Local FL Model (DTN Node)	5G Downlink	Aggregated model weights	Provide updated global model to DTN nodes
Contact Evaluator (DTN Node)	Relay Decision Engine (same node)	Internal (device)	Relay score	Evaluate whether to forward the message
Relay Decision Engine (DTN Node)	Peer DTN Node	D2D Wireless	Message payload	Transmit message opportunistically via local wireless
Destination Node (DTN Node)	Path Collector (MEC Runtime)	5G Uplink	Anonymized message path	Report successful message delivery for feedback
Label Dispatcher (MEC Runtime)	Supervised Learning Evaluator (DTN Node)	5G Downlink	Delivery labels (success/failure)	Tag encounters to enable supervised training

The communications carried out in the architecture enable the operational phases of the workflow. As a result, the internal processes within the devices are essential for functioning in the local context, while the external communications between nodes and B5G poles support the application of the FL model.

In addition to the components involved in the architecture, several important factors must be considered in the FL model, as its application to encounter evaluation involves strict requirements. The following section addresses these aspects.

3.4. FL-Based and Edge AI Routing for Opportunistic Relay Selection

The proposed routing strategy relies on a lightweight FL model deployed on edge AI nodes to optimize relay decisions. To avoid blindly broadcasting messages, intermediate and provisional nodes use the trained model to assess whether an encountered node is a suitable relay based on contextual features. The model is trained both locally and collaboratively using supervised data collected during message dissemination events. This subsection details the node segmentation process, the model's input variables, the regression approach, and the scalability and effectiveness of the proposed solution.

3.4.1. Decision Tree for Node Segmentation

Routing is applied only to non-destination nodes. To simplify decision boundaries, the network is logically segmented by the B5G edge infrastructure at the time of message injection. The sender node specifies a criterion for message relevance (e.g., destination location). Based on this filter, the B5G pole classifies all active nodes into three categories:

- Destination nodes. These nodes match the defined criterion and always receive the message.
- Intermediate nodes. These nodes clearly do not match the criterion.
- Provisional nodes. These nodes provide insufficient information, as they do not declare a destination.

Only intermediate and provisional nodes apply the FL model. This segmentation simplifies the routing decision process, as the model focuses exclusively on relay selection rather than destination prediction.

3.4.2. Model Input Variables

The FL model is trained to estimate the likelihood that an intermediate node will contribute to successful message delivery, using only encounter-level context. The goal is to identify reliable relays without embedding assumptions about specific messages or targeting logic. Therefore, the input vector is constructed using features that remain valid regardless of the sender's segmentation criteria, which may vary across transmissions. This design ensures that the model remains relevant and effective over time, even as the sender's rules change.

The input vector was defined by domain expertise, guided by prior DTN and routing ML literature and by the operational constraints of our setting. In delay-tolerant networks, encounter-driven signals, such as frequency, recency of contacts or contact duration; resource state, such as buffer availability; and mobility stability are established predictors or surrogates for deliverability and centrality under intermittency [6,17,32,40]. We therefore select features that (i) are measurable on-device without external infrastructure, (ii) remain available under short, opportunistic contacts, (iii) are invariant to the sender's transient segmentation, and (iv) fit small context vectors suitable for brief exchanges and FL aggregation at the edge. This yields the set in Table 2: contact duration as a proxy for successful exchange windows; buffer and battery as availability indicators; path fidelity and mobility flag as stability cues; encounter frequency and time since last contact as connectivity indicators; and a provisional status flag to capture uncertain path intent.

To preserve privacy and reusability across broadcasts, we intentionally exclude variables tied to message semantics or explicit recipient profiles, such as declared destination or interests, and avoid absolute geolocation traces. These choices ensure the model can generalize across different sender filters, keep sensitive data on devices, and operate with minimal uplink/downlink overhead when synchronizing with the serving pole.

All features are extracted locally during node encounters, individually normalized, and periodically updated as nodes move. Importantly, no information related to the message's intended recipient profile (e.g., declared destination or interest category) is used as input to the model. Instead, the model relies on structural, temporal, and behavioral characteristics observable during encounters, which both preserve privacy and support model reusability. These variables are described in Table 2.

This feature design supports a model that is agnostic to destination criteria, focusing on persistent contextual variables that enable generalization without requiring retraining from scratch, thus promoting model reusability. Once these variables are input into the model, a regression calculation is performed to estimate the relevance of the relay in the communication. The following subsection details this process.

Table 2. Input features for FL-based relay scoring model.

Feature	Justification	Data Type
Average contact duration	Longer contact windows increase the likelihood of successful data exchange.	Float
Buffer availability	Indicates the node's capacity to store messages and forward them later. A limited buffer may restrict acceptance of new messages.	Float
Battery level	A low battery level may lead to disconnection or shutdown, reducing node reliability.	Float
Fidelity to path	Measures whether the node is following a predictable and stable local trajectory.	Float
Encounter frequency	A high number of previous encounters may indicate strong connectivity and centrality within the network.	Integer
Time since last contact	Provides insight into the stability and regularity of node interactions.	Float
Mobility flag	Identifies the type of movement (e.g., pedestrian, static, vehicle) to estimate relay stability.	Categorical
Provisional flag	Marks nodes with undefined paths or ambiguous status that may still function as opportunistic relays.	Boolean

3.4.3. Regression Model over Classification

The architecture uses a regression approach to predict a continuous score representing the estimated usefulness of a contact for relaying. This technique is preferred over binary classification, as it allows for flexible thresholding that can adapt to network conditions or message priority. Additionally, it avoids rigid binary decisions, which may be unreliable in noisy, real-world environments. The regression output $\epsilon \in [0,1]$ represents the expected probability of successful message delivery if relayed through the encountered node. The following subsection provides a formalization of the model.

3.4.4. Model Formalization

Let each node n_i observe a set of encounters $E_i = \{e_1, e_2, \ldots, e_k\}$, where each encounter e_j is represented by a feature vector in Equation (1):

$$x_j = [x_{j1}, x_{j2}, \ldots, x_{jm}] \in \mathbb{R}^m \tag{1}$$

These features include contextual variables such as contact duration, buffer availability, battery level, path fidelity, or encounter frequency, all normalized to the range $[0,1]$. Next, Equation (2) represents this:

$$x_{jk}^{(\text{norm})} = \frac{x_{jk} - \min(x_k)}{\max(x_k) - \min(x_k)} \tag{2}$$

The goal is to learn a function $f_\theta : \mathbb{R}^m \to [0,1]$ parameterized by θ, as Equation (3) describes:

$$\hat{y}_j = f_\theta(\mathbf{x}_j) \tag{3}$$

where $\hat{y}_j$ is the predicted relay usefulness score, the probability that the encounter leads to a successful delivery.

Each sample is labeled using retrospective feedback from the B5G edge node, as Equation (4) describes:

$$y_j = \begin{cases} 1 & \text{if encounter } e_j \text{ was on a successful delivery path} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

The local loss function for node n_i is the mean squared error (MSE), as Equation (5) describes:

$$\mathcal{L}_i(\theta) = \frac{1}{k} \sum_{j=1}^{k} \left(f_\theta(\mathbf{x}_j) - y_j \right)^2 \quad (5)$$

Each node updates its local model f_θ by minimizing $\mathcal{L}_i$ and periodically sends its weight update θ_i to the B5G edge node. The edge node performs federated aggregation under an elected algorithm, such as Federated Averaging (FedAvg), represented in Equation (6):

$$\theta^{(\text{global})} = \sum_{i=1}^{N} \frac{|D_i|}{\sum_{j=1}^{N} |D_j|} \, \theta_i \quad (6)$$

where D_i is the local training set size at node n_i, and N is the number of contributing nodes.

The updated global model $\theta^{(\text{global})}$ is then broadcast to all participating nodes in the coverage area.

The decision rule at runtime is done when a node n_i encounters another node n_j, it computes $\hat{y} = f_\theta(\mathbf{x}_{ij})$, and transmits the message only if $\hat{y} > \tau$, where $\tau \in [0, 1]$ is a tunable threshold, e.g., $\tau = 0.7$, chosen to balance delivery ratio and redundancy.

3.4.5. Scalability and Effectiveness in Persistent Crowds

Considering the functioning of the architecture, the proposed model is particularly well suited for temporally extended, high-density environments (e.g., crowded public spaces, holiday events, tourist weekends). These scenarios offer favorable conditions in which node behavior tends to follow repetitive and predictable mobility patterns (e.g., back and forth movement along pedestrian corridors). Similarly, encounter patterns often show temporal regularity, enhancing the model's ability to learn and generalize relay suitability.

A larger population of mobile nodes contributes to a more diverse and abundant set of federated training instances, enabling improved local adaptation and more robust global models. As the number of interactions grows, the system becomes increasingly effective at distinguishing reliable relays from unreliable ones, without requiring central orchestration.

Importantly, to maintain reusability and scalability, the model must exclude any variables directly tied to the segregation or destination criteria defined by the sender. This is essential because segregation rules may vary from one broadcast to another, and embedding them in the model would result in obsolescence and reduced generalization. Instead, the model focuses on contact-level and contextual metrics that remain valid across different messages and operational conditions. This approach ensures that the federated learning process remains specific to the local context but agnostic to the defined criteria, making the system effective and resilient across varying broadcast intents within the same crowd environment.

Considering these design variables, the architecture applies the FL model as an enhancement to DTN routing for local communications. To further explore the role of edge computing and B5G infrastructure in this process, the following subsection examines the function of this component in detail.

3.5. Edge Computing in the B5G-Enhanced Architecture

The edge node plays a critical orchestration role in the proposed hybrid B5G-DTN architecture, acting as the computational and coordination backbone for FL and context dissemination. Deployed within a smart pole of the B5G infrastructure, the edge node is positioned close to end users, enabling real-time responsiveness with minimal dependence on the backhaul. This subsection addresses the physical deployment considerations, the node's contribution to the FL model, its role as a feedback dispatcher, and key aspects related to scalability.

3.5.1. Deployment in B5G Poles

Edge nodes are integrated into B5G poles located in high-density urban areas, such as tourist hotspots or large event venues. These poles not only provide high-speed wireless connectivity but also host lightweight computing environments capable of executing federated learning aggregation, contact validation, and message path tracing. Their geographical distribution enables localized model training and deployment while preserving isolation between distinct communication contexts.

Within the architecture, the coordinator is the smart pole serving the area where the crowd forms. Mobile devices attach to the serving 5G cell as usual; the pole associated with that cell provides the edge computing runtime that aggregates model updates and dispatches delivery feedback within its coverage area. If a device hands over to a different cell, it resynchronizes with the new pole and obtains the latest aggregated model. The coordinator, running on the B5G smart pole, handles only the reception of model weights, federated aggregation, distribution of the updated model, and delivery-label notifications. As a result, the coordinating pole neither routes payload messages nor collects raw encounter data.

3.5.2. FL Aggregator and Feedback Dispatcher

The edge node serves as the aggregation point for FL models trained locally on mobile devices. Once nodes accumulate sufficient encounter data and complete local training, their model weights are transmitted to the edge node. Using techniques such as FedAvg, the node aggregates these updates into a global model, which is then redistributed to both newly joined and existing participants. This process enables learning from contextual interaction histories without transferring sensitive personal data or location information.

Another crucial function of the edge node is to receive path success reports from destination nodes. When a destination node receives a message, it sends an anonymized record of the encounter path to the edge node. This allows the edge to notify all participating nodes whether their forwarding decisions contributed to a successful delivery. Such retrospective labeling is essential to ensure that local models are accurately updated with reward signals, thereby improving routing quality over time.

The edge node is also responsible for distributing the most recent global model to newly arrived nodes within its coverage area. When a node enters the local context, it checks for the availability of a trained global model. If one is available, the model is downloaded and integrated into the node's local inference loop. This ensures that routing decisions are informed by prior experiences in the same context, significantly improving communication efficiency from the outset.

3.5.3. Scalability and Applicability

The deployment of edge nodes in B5G poles introduces both significant opportunities and practical limitations in terms of scalability and applicability across different contexts. The proposed architecture inherently supports horizontal scalability through the spatial distribution of B5G poles. Each edge node operates autonomously within its coverage area, performing localized aggregation and coordination. This design enables the system to scale geographically without introducing a centralized bottleneck.

In this context, edge-based orchestration is particularly effective in persistent, high-density environments such as:

- Tourist districts with recurring mobility flows (e.g., two clearly defined directions).

- Transportation hubs (e.g., airports, metro stations).
- Large event venues with well-defined temporal boundaries.

Despite these applicability scenarios, the potential of the system is reduced in transient, low-density, or rural environments. This is especially due to the potentially insufficient node density to generate reliable FL training data, as well as the limited presence of B5G poles. These conditions can lead to delayed encounter feedback, hindering meaningful retrospective labeling.

Based on the explanations provided in this section, the proposed architecture offers a viable alternative for enabling local communications in crowded environments. To evaluate its effectiveness, the next section presents the implementation of a proof of concept and discusses the outcomes obtained in a realistic scenario.

4. Results

This section evaluates the technical viability and operational performance of the proposed architecture. To this end, a proof of concept has been implemented, focusing on the assessment of the FL model and its integration as the routing core. Additionally, two realistic scenarios have been simulated to demonstrate a potential application of the system under two conditions: low node density and high node density. The following subsections present the details of the proof of concept, the simulation setup, the obtained results, and a discussion of the findings.

4.1. Proof-of-Concept Implementation

To evaluate the proposed hybrid B5G DTN architecture, a proof of concept implementation has been developed (https://github.com/Bear-the-box/Hybrid-B5G-DTN-arch, accessed on 28 August 2025). For this purpose, a modified version of The ONE (Opportunistic Network Environment) simulator [51] has been used, along with a Python-based FL core responsible for routing model training, local instantiation of FL models, and aggregation. As shown in Figure 3, the implementation is based on the integration of these two components.

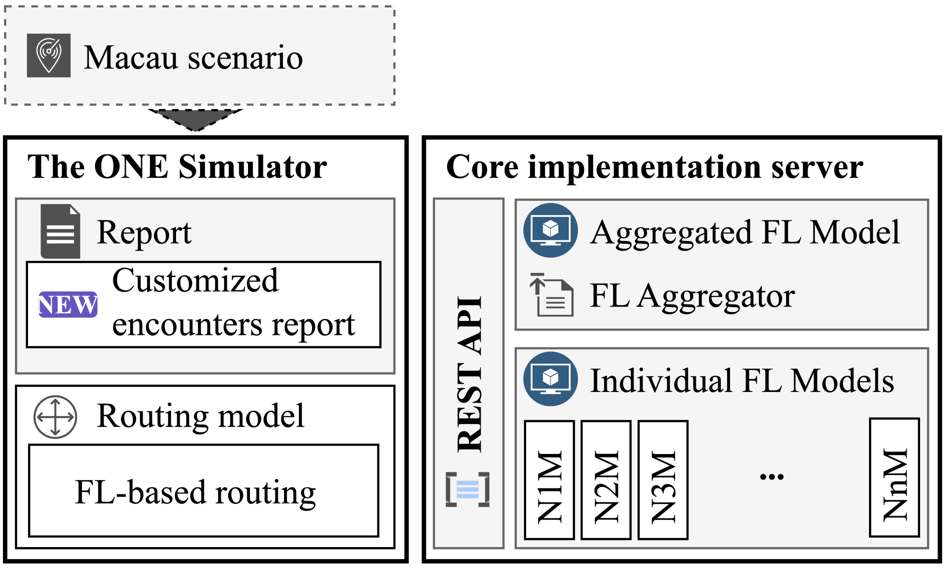


Figure 3. Architecture components integrated into the proof-of-concept.

In the case of The ONE simulator, two main integrations have been implemented: logging encounters between nodes and incorporating FL models into the routing decisions.

- Interaction report. To support the variables required by the model for training and generalization, a monitoring component has been integrated into The ONE. It logs each encounter along with the contextual information vector exchanged between nodes.
- FL-based routing. A routing algorithm has been implemented to perform two main tasks: while the local FL model is not yet trained, the node floods the network upon encountering intermediate nodes. Once the model becomes available, the corresponding FL model instance is invoked to compute a relevance score for the encounter.

On the other hand, a Python-based core implementation has been integrated to represent the core implementation server. This involves two main entities: the FL core model representing the B5G pole and the individual FL models running on the simulated nodes. Additionally, a REST API is provided to facilitate interactions between the simulation and the learning models.

- REST API. This component enables interaction between the model core and the simulation, allowing nodes to invoke their individual FL models to estimate encounter scores.
- Individual models. This array of models corresponds to the set of nodes involved in the simulation, ranging from Node 1 Model (N1M) to Node *n* Model (NnM). Each node is able to call its corresponding FL model instance for inference.
- Aggregated model. The core model of the implementation performs aggregation and distribution operations involving the individual models. It updates its weights by incorporating the training results from participating nodes. The FedAvg algorithm is used for this purpose due to its simplicity, compatibility, and proven effectiveness [52].

Based on this proof of concept, the implementation has been evaluated in an experimental setup using a simulated scenario of the Macau tourist district. The following subsection presents the execution details, simulation conditions, and the outcome variables considered for analysis.

4.2. Experimental Setup

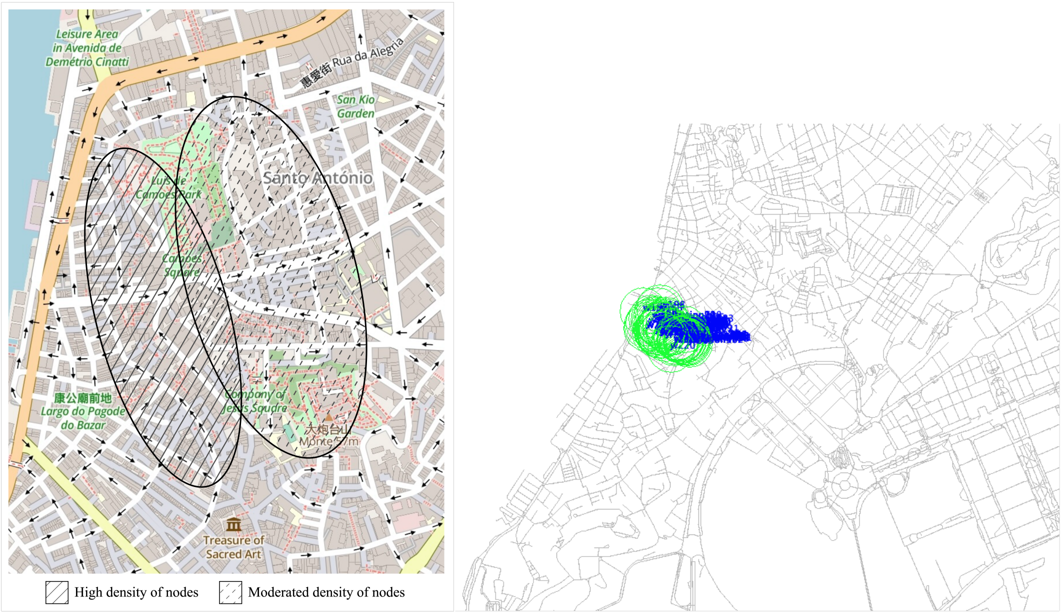
The experimental setup description focuses on the variables defined for the scenario and the evaluation metrics selected for performance assessment.

4.2.1. Scenario Variables

The experimental setup corresponds with a simulation based on the touristic district of the city of Macau. This city is popular worldwide because of its unique history and cultural mixture, factors that make Macau a top touristic destination for many travelers. However, this situation drives to high requirements for touristic planification to handle traffic, crowd flow, and notification about route changes. In this context, the proposed architecture aligns with the problematic, becoming a potential resource to deal with local contextual communications. To evaluate the impact of node density on the proposal's performance, we consider two realistic urban-mobility scenarios in which crowd flows naturally converge around key tourist landmarks (e.g., the Ruins of St. Paul's): one low-density and one high-density. Both contexts, shown in Figure 4, simulate low and high density events with varying pedestrian behaviors, including:

- Destination goers: Nodes actively navigating toward a specific landmark.
- Passersby: Users moving randomly without a defined destination. This category also includes stationary individuals or those engaged in other activities (e.g., shopkeepers).

Considering these specifications, Table 3 presents the most relevant variables used in the simulation. The scenario is designed to virtually represent a busy day in the tourist district of Macau. To reflect this, the simulation runs for 12 h, corresponding to the typical commercial schedule of the area. Regarding communication, DTN nodes are equipped with a Bluetooth interface with a theoretical range of 100 m [53]. This value also enhances the likelihood of node interactions during the experiment.



(**a**) (**b**)

Figure 4. Simulation scenario: (**a**) Node distribution heatmap indicating areas of higher density. (**b**) Simulation environment in The ONE, illustrating node mobility patterns around the tourist district.

Table 3. Simulation variables used in the experimental setup.

Variable	Value
Simulation length	43,200 s (12 h)
DTN interface type	Bluetooth
Bluetooth range	100 m
Number of node groups	3
Mobility model	MapRouteMovement
Mobility speed	0.5–1.5 m/s
Message generation interval	3600/7200/14,400 s
Validation threshold	Score > 70%

In addition to these variables, the scenario includes three node groups whose distributions emulate a busy urban environment as well as a lower-density setting for a fair comparison. After a sensitivity analysis, we selected the following group sizes.

In the low-density scenario, we use one static sender, 50 destination nodes, and 100 intermediate nodes. In the high-density scenario, we use one static sender, 126 destination nodes, and 300 intermediate nodes. Regarding mobility, all nodes, except the sender, follow the MapRouteMovement model in both scenarios, strictly adhering to street paths. Walking speed reflects a standard pedestrian pace.

Finally, the simulation considers two key variables that significantly influence performance: the message generation interval and the validation threshold for the FL model.

- The message generation interval includes three different values: 3600 s, 7200 s, and 14,400 s. This means that the sender generates a new message over the network each time the specified interval elapses. These values align with potential notifications related to traffic updates and route limitations in the area. Additionally, the use of varying intervals introduces different levels of network traffic, allowing for an assessment of its impact on performance.

- The validation threshold refers to the minimum score a contacted node must reach to be considered a potential intermediate node. This value acts as a selective criterion, enforcing a stringent threshold for routing decisions.

In addition to this configuration, simulations are also performed using alternative well-known protocols to compare the baseline behavior of the proposed algorithm:

- Epidemic routing [54]. This algorithm serves as the standard flooding technique for broadcasting information in local contexts, providing a density-sensible approach.
- First contact [55]. This strategy provides a single-copy greedy baseline. Thus, it is useful to contextualize gains from learning or replication.
- PRoPHET [56]. This contribution is based on classic probability, performing contact-history routing. It becomes relevant to assess with the FL-based approach.
- MaxProp [57]. This strategy is based on queue management and path calculation. Considering its outstanding overhead performance, it is relevant to address in the scenario.
- Spray-and-wait [58]. This algorithm provides multi-copy routing, following a clean delivery/overhead trade-off.

Considering these comparative strategies, the next subsection details the considered evaluation metrics.

4.2.2. Evaluation Metrics

The metrics obtained in the assessment relate to two main elements of the proposal: routing performance and FL model accuracy.

For routing, Quality of Service (QoS) metrics are used to evaluate the impact of the routing strategy on communications. Specifically, four key statistics are considered:

- Delivery probability. This metric relates the number of received messages to the number of sent messages. As such, it strongly reflects communication success and serves as a key performance indicator for the scenario.
- Average latency. This metric measures the average time required to transmit a message from the sender to the destination.
- Average overhead. This metric compares the number of message copies sent across the network to the number of received messages. It serves as a useful indicator of channel utilization.
- Average hop count. This metric calculates the average number of nodes involved in forwarding a message from the sender to the destination.

For the aggregated FL model, a regression calculation is performed. To evaluate its performance, four key metrics are considered:

- MSE. As mentioned in Equation (5), this variable measures the average of the squares of the errors, that is, the average squared difference between predicted and actual values. It penalizes larger errors more heavily and is a standard metric for regression tasks.
- Root Mean Squared Error (RMSE). The square root of MSE, providing an error metric in the same units as the predicted variable. It offers more interpretable insights into error magnitude in real-world units.
- Mean Absolute Error (MAE). Calculates the average of the absolute differences between predicted and actual values. Unlike MSE, MAE treats all errors equally and is more robust to outliers.

- R-squared (R^2). Indicates the proportion of variance in the dependent variable that is predictable from the independent variables. A higher R^2 signifies that the model explains a greater portion of the variance in the data.

These metrics provide a balanced and comprehensive evaluation of the FL model's ability to accurately predict encounter scores, helping to quantify how well the model generalizes across diverse network conditions and nodes. Considering these variables, the next section presents the obtained results.

4.3. Results

The results are organized into two evaluated scenarios—low-density and high-density contexts. For each setup, we report routing QoS metrics and FL model performance.

4.3.1. Low Density Scenario

This scenario serves as a baseline to test the proposal's performance. It is a challenging setup with fewer nodes and a lower probability of contact. We next analyze its impact on routing QoS and FL model performance.

Routing QoS Evaluation

Figure 5 presents the QoS metrics obtained from the low-density simulated scenario.

Figure 5a displays the delivery probability regarding the message generation interval. With sparser contacts, FL underperforms comparative strategies. PRoPHET and MaxProp deliver the highest ratios, followed by Spray-and-Wait. Epidemic improves with a lighter load but remains below the history-aware schemes. The FL-based approach trails in this regime, reflecting fewer labeled encounters and slower model maturation. First Contact becomes the lowest. All curves rise with the message interval, consistent with reduced contention.

Figure 5b represents the latency regarding the message generation interval. Delays are overall high but decrease as the interval grows. Flooding is fastest here because it exploits the first available paths. PRoPHET/MaxProp are close, while FL shows a larger penalty under sparsity due to selective forwarding plus slower learning. Finally, First Contact is consistently the slowest.

Figure 5c represents overhead regarding the message generation interval. All methods achieve a moderate consumption. Epidemic becomes the highest, while FL-based routing is less efficient than alternative multi-copy schemes, being outperformed by PRoPHET, MaxProp, and Spray-and-Wait. The increase for FL-based routing relative to these baselines aligns with its weaker scoring under limited feedback. First Contact stays near the minimum by construction.

Figure 5d displays hop count over the message generation interval. The value rises as the interval increases. FL-based routing remains in an intermediary band. First Contact is similar but a touch lower at small intervals, consistent with single-copy forwarding that sometimes stalls rather than exploring longer chains.

In low density, contact scarcity slows FL-based labeling and aggregation, so PRoPHET/MaxProp, which rely on encounter predictability and aging, achieve higher delivery and lower overhead. At the same time, Epidemic minimizes delay at the cost of airtime. These results are consistent with the design of the architecture: the FL-based approach lacks the required dataset of encounters, degrading as signals are sparse. Potential mitigation includes adaptive forwarding thresholds during cold start, limited exploration budgets, or warm-starting models from prior contexts. These proposals will be involved in future works.

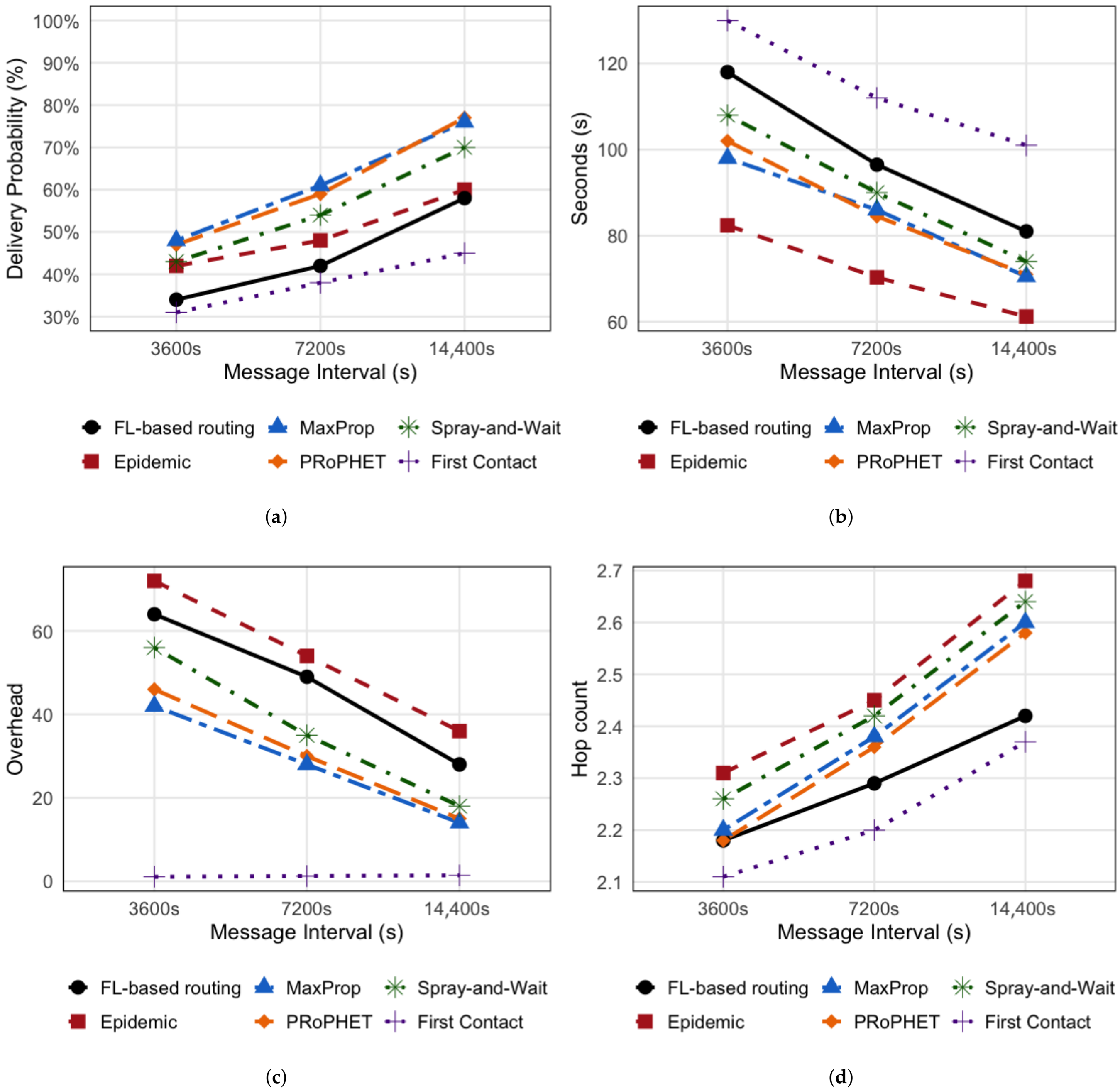


Figure 5. QoS outcomes comparison in low density sceanario: (**a**) Delivery probability versus message generation interval. (**b**) Average latency versus message generation interval. (**c**) Average overhead ratio versus message generation interval. (**d**) Average hop count versus message generation interval.

FL Model Assessment

With only 50 destinations and 100 intermediates, labeled encounters are scarce and noisy, which slows convergence and leads to underfitting. As a result, errors increase and explanatory power drops compared to the dense case. Table 4 summarizes the aggregate training+validation results for the low-density scenario:

The RMSE and MAE indicate wide dispersion and large absolute errors, while R^2 shows limited variance explanation, consistent with reduced label density. A high class imbalance due to more failed than successful relays and fewer repeated contacts per node drives the model to a low performance. This weaker fit aligns with the routing outcomes:

under sparse contacts, the FL scoring is less discriminative, which raises latency and overhead relative to PRoPHET/MaxProp and lowers delivery probability.

Table 4. FL model performance in low density scenario.

Variable	Value
MSE	0.168
RMSE	0.410
MAE	0.336
R-squared	0.12

4.3.2. High Density Scenario

The high-density scenario provides a realistic, insightful context to test the real purpose of the proposed architecture. This setup represents the ideal situation to deploy the architecture. Aiming to fully assess the performance of the proposal, the next subsections analyze the QoS impact and the FL model outcomes.

Routing QoS Evaluation

Figure 6 presents the QoS metrics obtained from the high-density simulated scenario.

Figure 6a shows the delivery probability obtained for the routing strategies regarding the message generation interval. Across all intervals, the FL-based scheme attains the highest delivery. History-aware multi-copy baselines, MaxProp and PRoPHET track closely, especially at low traffic (14,400 s), where PRoPHET nearly matches FL. Spray-and-Wait is lower, while Epidemic improves with a lighter load but remains below these methods. First Contact is the weakest. In all cases, delivery increases as the message interval grows, reflecting reduced contention.

Figure 6b shows the average latency in relation to the message generation interval. In this case, latency falls as load decreases for every algorithm. At higher load (3600–7200 s), Epidemic is slightly faster than FL, whereas at 14,400 s FL becomes competitive and PRoPHET is lowest. MaxProp is close to FL throughout. Spray-and-Wait is slower, and First Contact is consistently the slowest, consistent with single-copy waiting.

Figure 6c displays the average overhead ratio with respect to the message generation interval. In this case, FL yields the smallest overhead among multi-copy schemes, dropping from 63.93 to 8.70 as the interval increases. PRoPHET and MaxProp are close, while Spray-and-Wait is higher. Epidemic incurs the largest overhead by design. As expected, First Contact's overhead is near the minimum but at the cost of delivery and delay.

Figure 6d shows the average hop count relative to the message generation interval. In this case, the outcomes are similar across methods and rise modestly with the interval. FL remains low to moderate ($2.00 \rightarrow 2.39$), comparable to PRoPHET/MaxProp and below Epidemic at higher intervals.

In this high-density scenario, the FL-based relay scoring achieves the best delivery with the lowest multi-copy overhead, trading only a small latency penalty under heavy load and matching leading heuristics at lighter load. History-aware baselines, such as PRoPHET and MaxProp, degrade gracefully and approach FL as traffic eases, while Epidemic offers lower delay early but at prohibitive overhead. First Contact minimizes overhead by construction but suffers in both delivery and latency. Overall, the results support FL-guided, edge-coordinated DTN routing as an effective offloading strategy for local, ephemeral exchanges in crowded areas.

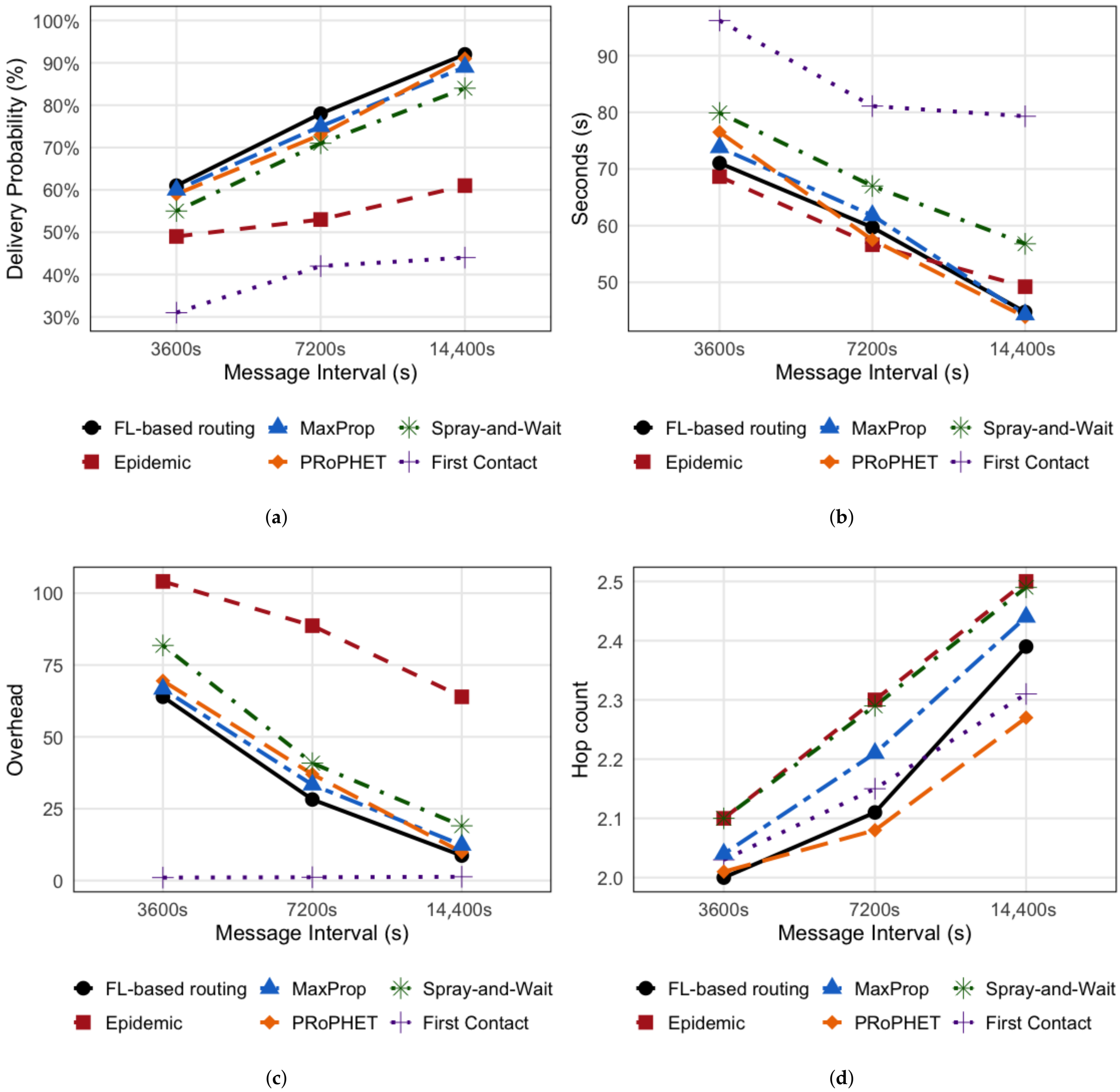


Figure 6. QoS outcomes comparison in high-density scenario: (**a**) Delivery probability versus message generation interval. (**b**) Average latency versus message generation interval. (**c**) Average overhead ratio versus message generation interval. (**d**) Average hop count versus message generation interval.

FL Model Assessment

Table 5 presents the results obtained for each performance variable of the model, after the training and evaluation in the high-density scenario. The MSE and RMSE represent the average squared and root mean squared deviations from the true values, respectively. Given that the predicted scores range between 0 and 1, such as contact success probability or encounter quality, an RMSE of approximately 0.31 indicates moderate dispersion around the actual values. The MAE shows that, on average, predictions deviate by about 24%, which may be significant depending on the application. An R-squared value of 0.41 indicates that the model explains only 41% of the variability in the outcome variable. While this performance is better than random guessing, it suggests that the model struggles to capture the full complexity of the relationships within the dataset.

Table 5. FL model performance in high-density scenario.

Variable	Value
MSE	0.095
RMSE	0.308
MAE	0.246
R-squared	0.41

The regression fit of the encounter-score model is moderate, but routing performance depends primarily on ranking and selective pruning rather than precise probability calibration. The forwarding rule applies a threshold on the score, so preserving a coarse ordering between weak and plausible relays suffices to suppress low-value transmissions. Moreover, labels are assigned from realized delivery paths and thus contain path-dependence and counterfactual noise: encounters that could have led to delivery under slightly different queue/timing conditions are labeled as failures. Such noise penalizes squared-error metrics even when the ranking used for decisions is adequate.

4.4. Discussion

In the high-density setting, the FL-based router attains consistently high delivery and clear channel-use savings. Delivery probability improves over the alternative strategies at all intervals and approaches the best heuristic baselines at long intervals and heavier-traffic cases. Latency shows a modest trade-off at short intervals but becomes competitive as traffic eases. Taken together, the dense-scenario outcomes indicate that learning encounter scores and forwarding selectively can sustain delivery while curbing redundant copies, which directly contributes to offloading. The obtained results demonstrate strong performance for the FL-based routing model, achieving a high delivery probability while outperforming the comparative testbeds in terms of overhead and hop count.

In the case of the low-density setting, all methods face sparser contacts and longer paths, and the relative ranking shifts. Heuristic multi-copy schemes such as PRoPHET and MaxProp lead delivery and latency, whereas the FL-based router trails in delivery and shows higher latency. Overhead decreases system-wide under sparsity; however, FL is no longer the lowest among multi-copy strategies. These results align with the learning signal available in this regime: fewer successful paths and fewer repeated encounters reduce label density and diversity, weakening score discrimination.

The FL model metrics corroborate this picture. In the dense case, RMSE indicates underfitting due to scarce, imbalanced labels and limited repetition across nodes. Even so, the FL router maintains reasonable hop counts and bounded overhead and remains competitive with the alternative strategies at longer intervals, while PRoPHET/MaxProp becomes preferable when crowds thin.

Overall, the study suggests a clear operating region for the proposal: persistent crowds with abundant local interactions, where edge-assisted aggregation plus local scoring yield both offloading and delivery gains. In sparser contexts, performance needs to be improved. In future works, density-aware operations, such as relaxing the forwarding threshold, lengthening aggregation windows, and using a heuristic fallback when local validation degrades, will be explored.

Considering the performance of the proof of concept, the results strongly support further research and development of the prototype. Despite the addressed challenges in both testbed scenarios, the current proof of concept demonstrates the technical viability of the architecture, providing a solid foundation for subsequent development.

5. Conclusions

This paper presents an FL-driven architecture for opportunistic communication in crowded local environments, supported by edge AI, B5G poles, and DTN principles. By leveraging edge-assisted aggregation and contextual relay scoring, the system enables adaptive routing without requiring centralized data collection. The proposed model operates effectively in dynamic, large-scale scenarios such as urban tourism, where mobility patterns and connectivity are highly variable. The architecture also demonstrates how edge computing can coordinate mobile nodes without compromising their autonomy, while ensuring privacy-preserving learning and sensitivity to context during message propagation.

The proof-of-concept simulation based on Macau's downtown layout confirms the feasibility of the design: in dense scenarios, a delivery probability of 92% is achieved, with an average value of 77%; latency reaches an average of 58.5 s, while maintaining reduced overhead compared to state-of-the-art routing. In low-density scenarios, the proposal faces challenges to generalize, achieving a performance lower than comparative benchmarks. Considering these results, several research directions remain open to address practical limitations and scalability concerns. These include providing density-aware operation, supporting online and continual learning for real-time adaptation, refining node segmentation to improve routing decisions, and introducing crowd sensing mechanisms to increase environmental awareness and response accuracy. Together, these improvements will advance the architecture towards a resilient and scalable communication system for next-generation intelligent urban networks.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/fi17090392/s1.

Author Contributions: Conceptualization, M.J.-A., M.Z. and V.N.G.J.S.; methodology, M.J.-A., M.Z. and V.N.G.J.S.; software, M.J.-A., M.Z. and V.N.G.J.S.; validation, M.J.-A., M.Z. and V.N.G.J.S.; funding acquisition, V.N.G.J.S. All authors have read and agreed to the published version of the manuscript.

Funding: V.N.G.J.S. acknowledges that work is funded by FCT/MECI through national funds and, when applicable, co-funded EU funds under UID/50008: Instituto de Telecomunicações.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is available upon request due to the internal policies of the affiliated institution and in Supplementary Materials.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Theopilus, Y.; Al Mahmud, A.; Davis, H.; Octavia, J.R. Persuasive strategies in digital interventions to combat internet addiction: A systematic review. *Int. J. Med. Inform.* **2025**, *195*, 105725. [CrossRef]
2. Spaho, E.; Jaupi, O. A Framework for Integrating 5G and Vehicular Delay Tolerant Networks in Smart Cities. In *Advanced Information Networking and Applications, Proceedings of the 39th International Conference on Advanced Information Networking and Applications (AINA-2025)*; Barolli, L., Ed.; Springer: Cham, Switzerland, 2025; pp. 86–95.
3. Shah, I.A.; Ahmed, M. An Optimized Load Balancing Probabilistic Protocol for Delay Tolerant Networks. *SN Comput. Sci.* **2025**, *6*, 140. [CrossRef]
4. Jesús-Azabal, M.; García-Alonso, J.; Galán-Jiménez, J. Beyond 5G Satellite Transmission and Indoor UAVs in Dams Monitoring: An Opportunistic Data Collection Strategy for Isolated Rural Areas. *IT Prof.* **2024**, *26*, 36–41. [CrossRef]
5. Jesús-Azabal, M.; Zhang, Z.; Gao, B.; Yang, J.; Soares, V.N.G.J. Connection-Aware Digital Twin for Mobile Adhoc Networks in the 5G Era. *Future Internet* **2024**, *16*, 399. [CrossRef]

6. Jesús-Azabal, M.; García-Alonso, J.; Galán-Jiménez, J. Evaluating the quality of service of Opportunistic Mobile Ad Hoc Network routing algorithms on real devices: A software-driven approach. *Ad Hoc Netw.* **2024**, *163*, 103591. [CrossRef]
7. D'Argenio, P.R.; Fraire, J.; Hartmanns, A.; Raverta, F. Comparing Statistical, Analytical, and Learning-Based Routing Approaches for Delay-Tolerant Networks. *ACM Trans. Model. Comput. Simul.* **2025**, *35*, 1–26. [CrossRef]
8. Varga, P.; Jászberényi, A.I.; Pásztor, D.; Nagy, B.; Nasar, M.; Raisz, D. How Beyond-5G and 6G Makes IIoT and the Smart Grid Green—A Survey. *Sensors* **2025**, *25*, 4222. [CrossRef] [PubMed]
9. Kumari, R.; Tyagi, D.K.; Battula, R.B. NEIL: An EfficIent cLuster-based device discovery for D2D communication in B5G. *Peer-to-Peer Netw. Appl.* **2024**, *18*, 32. [CrossRef]
10. Zhan, S.; Huang, L.; Luo, G.; Zheng, S.; Gao, Z.; Chao, H.C. A Review on Federated Learning Architectures for Privacy-Preserving AI: Lightweight and Secure Cloud–Edge–End Collaboration. *Electronics* **2025**, *14*, 2512. [CrossRef]
11. Lin, N.; Wang, Y.; Zhang, E.; Yu, K.; Zhao, L.; Guizani, M. Feedback Delay-Tolerant Proactive Caching Scheme Based on Federated Learning at the Wireless Edge. *IEEE Netw. Lett.* **2023**, *5*, 26–30. [CrossRef]
12. Yu, Z.; Hu, J.; Min, G.; Lu, H.; Zhao, Z.; Wang, H.; Georgalas, N. Federated Learning Based Proactive Content Caching in Edge Computing. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, 9–13 December 2018; pp. 1–6. [CrossRef]
13. Bhardwaj, M.; Singh, J.; Gupta, N.; Jadon, K.S.; Dhurandher, S.K. A novel federated learning approach for routing optimisation in opportunistic IoT networks. *Int. J. Sens. Netw.* **2024**, *46*, 24–38. [CrossRef]
14. Fasihi, M.R.; Mark, B.L. Device-to-Device Communication in 5G/6G: Architectural Foundations and Convergence with Enabling Technologies. *arXiv* **2025**, arXiv:2507.06946.
15. NIST. *Study of 5G New Radio (NR) Support for Direct Mode Communications*; Technical report; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2021.
16. Kamei, S.; Taghipour, S. A comparison study of centralized and decentralized federated learning approaches utilizing the transformer architecture for estimating remaining useful life. *Reliab. Eng. Syst. Saf.* **2023**, *233*, 109130. [CrossRef]
17. Gandhi, J.; Narmawala, Z. A comprehensive survey on Machine Learning techniques in opportunistic networks: Advances, challenges and future directions. *Pervasive Mob. Comput.* **2024**, *100*, 101917. [CrossRef]
18. McMahan, H.B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), Ft. Lauderdale, FL, USA, 20–22 April 2017; Volume 54, pp. 1273–1282.
19. Bonawitz, K.; Eichner, H.; Grieskamp, W.; Huba, D.; Ingerman, A.; Ivanov, V.; Kiddon, C.; Konečný, J.; Mazzocchi, S.; McMahan, H.B.; et al. Towards Federated Learning at Scale: System Design. In Proceedings of the MLSys, Stanford, CA, USA, 31 March–2 April 2019.
20. Kairouz, P.; McMahan, H.B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A.N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. Advances and Open Problems in Federated Learning. *J. Mach. Learn. Res.* **2021**, *22*. [CrossRef]
21. Yang, Q.; Liu, Y.; Chen, T.; Tong, Y. Federated Learning: Concepts, Applications, and Challenges. *IEEE Access* **2019**, *7*,hle38137. [CrossRef]
22. Vepakomma, P.; Gupta, O.; Swedish, T.C.; Raskar, R. Split Learning for Health: Distributed Deep Learning without Sharing Raw Patient Data. *arXiv* **2018**, arXiv:1812.00564. [CrossRef]
23. Erunkulu, O.O.; Zungeru, A.M.; Lebekwe, C.K.; Mosalaosi, M.; Chuma, J.M. 5G Mobile Communication Applications: A Survey and Comparison of Use Cases. *IEEE Access* **2021**, *9*, 97251–97295. [CrossRef]
24. Bora, B.; Chauhan, R.; Giri, S.; Rawat, R. Peer to peer confidentiality in networking. In *Challenges in Information, Communication and Computing Technology*; CRC Press: Boca Raton, FL, USA, 2024; pp. 720–725. [CrossRef]
25. Zheng, H.; Tran, T.; Shadmon, R.; Arden, O. Decentagram: Highly-Available Decentralized Publish/Subscribe Systems. In Proceedings of the 2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Brisbane, Australia, 24–27 June 2024; pp. 274–287. [CrossRef]
26. Arruda, C.E.; Moraes, P.F.; Agoulmine, N.; Martins, J.S.B. Enhanced Pub/Sub Communications for Massive IoT Traffic with SARSA Reinforcement Learning. In *Machine Learning for Networking, Proceedings of the Third International Conference, Paris, France, 24–26 November 2020*; Renault, É., Boumerdassi, S., Mühlethaler, P., Eds.; Springer: Cham, Switzerland, 2021; pp. 204–225.
27. Kang, Z.; Canady, R.; Dubey, A.; Gokhale, A.; Shekhar, S.; Sedlacek, M. A Study of Publish/Subscribe Middleware Under Different IoT Traffic Conditions. In Proceedings of the International Workshop on Middleware and Applications for the Internet of Things, Delft, The Netherlands, 7–11 December 2020; pp. 7–12. [CrossRef]
28. Tariq, A.H.; Amin, U. Peer-to-peer multi-energy trading in a decentralized network: A review. *Renew. Sustain. Energy Rev.* **2025**, *208*, 114969. [CrossRef]
29. Miyoshi, T.; Yamashita, S.; Yamazaki, T. Enhancing Road Safety: A Pedestrian-to-Vehicle Collision Warning System Integrating Location-Based P2P and LiDAR Sensing. In Proceedings of the 2025 1st International Conference on Consumer Technology (ICCT-Pacific), Matsue, Japan, 29–31 March 2025; pp. 1–4. [CrossRef]

30. Hamad, S.; Yeferny, T. Routing Approach for P2P Systems Over MANET Network. *arXiv* **2020**, arXiv:2101.03109. [CrossRef]
31. Adnan, M.H.; Ahmad Zukarnain, Z. Device-To-Device Communication in 5G Environment: Issues, Solutions, and Challenges. *Symmetry* **2020**, *12*, 1762. [CrossRef]
32. Castillo, A.; Juiz, C.; Bermejo, B. Delay and Disruption Tolerant Networking for Terrestrial and TCP/IP Applications: A Systematic Literature Review. *Network* **2024**, *4*, 237–259. [CrossRef]
33. Madamori, O.; Max-Onakpoya, E.; Grant, C.; Baker, C. Using Delay Tolerant Networks as a Backbone for Low-Cost Smart Cities. In Proceedings of the 2019 IEEE International Conference on Smart Computing (SMARTCOMP), Washington, DC, USA, 12–15 June 2019; pp. 468–471. [CrossRef]
34. Jesús-Azabal, M.; Herrera, J.L.; Laso, S.; Galán-Jiménez, J. OPPNets and Rural Areas: An Opportunistic Solution for Remote Communications. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 8883501. [CrossRef]
35. Jesús-Azabal, M.; Berrocal, J.; García-Alonso, J.; Soares, V.N.G.J.; Galán-Jiménez, J. An Opportunistic Routing Solution to Monitor Isolated Elderly People in Rural Areas. In *Gerontechnology, Proceedings of the Second International Workshop, IWoG 2019, Cáceres, Spain, 4–5 September 2019*; García-Alonso, J., Fonseca, C., Eds.; Springer: Cham, Switzerland, 2020; pp. 195–203.
36. Jesús-Azabal, M.; Soares, V.N.G.J.; Galán-Jiménez, J. ML-Enhanced Live Video Streaming in Offline Mobile Ad Hoc Networks: An Applied Approach. *Electronics* **2024**, *13*, 1569. [CrossRef]
37. Almotairi, K.H. An improved multipath video data communication in a vehicular delay-tolerant network. *PLoS ONE* **2022**, *17*, e0273751. [CrossRef]
38. Ferreira, R.; Moreira, W.; Mendes, P.; Gerla, M.; Cerqueira, E. Improving the Delivery Rate of Digital Inclusion Applications for Amazon Riverside Communities by Using an Integrated Bluetooth DTN Architecture. *arXiv* **2014**, arXiv:1405.7084. [CrossRef]
39. Turcanu, I.; Castignani, G.; Faye, S. On the Integration of Digital Twin Networks into City Digital Twins: Benefits and Challenges. In Proceedings of the 2024 IEEE 21st Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, USA, 6–9 January 2024; pp. 752–758. [CrossRef]
40. Xiong, Y.; Jiang, S. Multi-Decision Dynamic Intelligent Routing Protocol for Delay-Tolerant Networks. *Electronics* **2023**, *12*, 4528. [CrossRef]
41. Wang, X.; Xiong, G.; Cao, H.; Li, J.; Liu, Y. Decentralized Federated Learning with Model Caching on Mobile Agents. *Proc. Aaai Conf. Artif. Intell.* **2025**, *39*, 21296–21303. [CrossRef]
42. Saif, S.; Islam, M.J.; Jahangir, M.Z.B.; Biswas, P.; Rashid, A.; Nasim, M.A.A.; Gupta, K.D. A Comprehensive Review on Understanding the Decentralized and Collaborative Approach in Machine Learning. *arXiv* **2025**, arXiv:2503.09833.
43. He, Z.; Zhu, G.; Zhang, S.; Luo, E.; Zhao, Y. FedDT: A Communication-Efficient Federated Learning via Knowledge Distillation and Ternary Compression. *Electronics* **2025**, *14*, 2183. [CrossRef]
44. Caldarola, D.; Cagnasso, P.; Caputo, B.; Ciccone, M. Beyond Local Sharpness: Communication-Efficient Global Sharpness-aware Minimization for Federated Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 11–15 June 2025; pp. 25187–25197.
45. You, L.; Guo, Z.; Yuen, C.; Chen, C.Y.C.; Zhang, Y.; Poor, H.V. A framework reforming personalized Internet of Things by federated meta-learning. *Nat. Commun.* **2025**, *16*, 3739. [CrossRef]
46. Liu, Y.; Li, S.; Li, W.; Qian, H.; Xia, H. A Personalized Federated Learning Algorithm Based on Dynamic Weight Allocation. *Electronics* **2025**, *14*, 484. [CrossRef]
47. Amadeo, M.; Campolo, C.; Ruggeri, G.; Molinaro, A. Improving communication performance of Federated Learning: A networking perspective. *Comput. Netw.* **2025**, *267*, 111353. [CrossRef]
48. Almeida, L.; Teixeira, R.; Baldoni, G.; Antunes, M.; Aguiar, R.L. Federated Learning for a Dynamic Edge: A Modular and Resilient Approach. *Sensors* **2025**, *25*, 3812. [CrossRef] [PubMed]
49. Albelaihi, R. Mobility Prediction and Resource-Aware Client Selection for Federated Learning in IoT. *Future Internet* **2025**, *17*, 109. [CrossRef]
50. Singh, J.; Dhurandher, S.K.; Woungang, I. Federated Learning Empowered Routing for Opportunistic Network Environments. In Proceedings of the 2024 IEEE International Conference on Communications Workshops (ICC Workshops), Denver, CO, USA, 9–13 June 2024; pp. 1998–2004. [CrossRef]
51. Keränen, A.; Ott, J.; Kärkkäinen, T. The ONE simulator for DTN protocol evaluation. In Proceedings of the 2nd International Conference on Simulation Tools and Techniques, Rome, Italy, 2–6 March 2009. [CrossRef]
52. Liu, Y.; Qu, Z.; Wang, J. Compressed Hierarchical Federated Learning for Edge-Level Imbalanced Wireless Networks. *IEEE Trans. Comput. Soc. Syst.* **2025**,*1*, 1–12. [CrossRef]
53. Koulouras, G.; Katsoulis, S.; Zantalis, F. Evolution of Bluetooth Technology: BLE in the IoT Ecosystem. *Sensors* **2025**, *25*, 996. [CrossRef]
54. Rashidi, L.; Towsley, D.; Mohseni-Kabir, A.; Movaghar, A. On the Performance Analysis of Epidemic Routing in Non-Sparse Delay Tolerant Networks. *IEEE Trans. Mob. Comput.* **2023**, *22*, 4134–4149. [CrossRef]

55. Alajeely, M.; Doss, R.; Ahmad, A. Routing Protocols in Opportunistic Networks—A Survey. *IETE Tech. Rev.* **2018**, *35*, 369–387. [CrossRef]
56. Sok, P.; Kim, K. Distance-based PRoPHET routing protocol in Disruption Tolerant Network. In Proceedings of the 2013 International Conference on ICT Convergence (ICTC), Jeju Island, Republic of Korea, 14–16 October 2013; pp. 159–164. [CrossRef]
57. Das, M.; Sarkar, S.; Iqbal, S.M.A. TTL based MaxProp routing protocol. In Proceedings of the 2016 19th International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 18–20 December 2016; pp. 7–12. [CrossRef]
58. Spyropoulos, T.; Psounis, K.; Raghavendra, C.S. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, New York, NY, USA, 26 August 2005; pp. 252–259. [CrossRef]