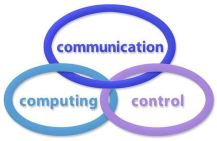
INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 21, Issue: 3, Month: June, Year: 2026
Article Number: 7263, https://doi.org/10.15837/ijccc.2026.3.7263



CCC Publications

Federated Split Learning with Large Language Models Integration: A Study on Potential Container Source Identification in Sea-Rail Intermodal Transport

W. MA, L. Huang*, Q. Zhang, Y. Wang, X. Zhang, R. Song

Weiguang Ma
Department of Information Management, School of Economics and Management
Beijing Jiaotong University, Beijing 100044, China
21113054@bjtu.edu.cn

Lei Huang*
Department of Information Management, School of Economics and Management
Beijing Jiaotong University, Beijing 100044, China
*Corresponding author: lhuang@bjtu.edu.cn

Qianyao Zhang
Department of Information Management, School of Economics and Management
Beijing Jiaotong University, Beijing 100044, China
21113052@bjtu.edu.cn

Ying Wang
Department of Information Management, School of Economics and Management
Beijing Jiaotong University, Beijing 100044, China
ywang1@bjtu.edu.cn

Xiong Zhang
Department of Information Management, School of Economics and Management
Beijing Jiaotong University, Beijing 100044, China
xiongzhang@bjtu.edu.cn

Rongjia Song
Experimental Center of Data Science and Intelligent Decision Making, Department of Information Management, School of Management
Hangzhou Dianzi University, Hangzhou 310018, China
rongjia.song@hdu.edu.cn

Abstract

To address the challenge of potential container source identification in sea-rail intermodal transport scenarios, which arises from data silos and privacy barriers among multiple stakeholders, this paper proposes FSL-Qwen, a Federated Split Learning framework integrated with Large Language Models. Innovative to this framework is the vertical partitioning of the Qwen model at the embedding layer: clients (e.g., ports, railways, customs) deploy only lightweight embedding layers for local feature extraction, while the server retains the Transformer backbone for centralized reasoning. This architecture decouples local computation from inference, theoretically reducing client-side complexity to $\mathcal{O}(1)$ and drastically minimizing communication overhead compared to standard Federated Learning. To resolve cross-domain semantic heterogeneity, a ChatML-based semantic alignment mechanism is introduced, enabling collaborative inference without sharing raw records. Privacy analysis demonstrates that the framework achieves inherent structural isolation, converting data reconstruction attacks into blind inverse problems. Experiments on a dataset of 48,800 SRIT container data confirm that FSL-Qwen achieves a predictive accuracy of 94.0% and an F1-score of 94.1%, effectively matching the centralized upper bound while limiting client-side

memory usage to merely 0.26 GB. These results validate FSL-Qwen as a robust, efficient, and privacy-preserving paradigm for intelligent logistics decision-making.

Keywords: federated learning; split learning; large language model; sea-rail intermodal transport; privacy computing

1 Introduction

Sea-rail intermodal transport (SRIT) has emerged as a critical component of sustainable global logistics, offering significant advantages in carbon emission reduction and transport efficiency. However, the scalability of SRIT is severely constrained by the data silo phenomenon distributed across stakeholders such as ports, railways, and customs [1, 2]. Although effective cargo source identification relies on the integration of logistics data, these entities are often legally or commercially restricted from sharing raw data [3]. Traditional centralized machine learning approaches fail to address these privacy constraints [4], while standard Federated Learning (FL) methods face significant challenges in handling the semantic heterogeneity of multi-source business data and the computational resource limitations of clients.Existing container source identification methods mostly rely on statistical forecasting or shallow machine learning models that work with data from a single source [5]. These methods are limited in capturing the complex, non-linear relationships between different logistics stakeholders. Moreover, the direct application of Large Language Models (LLMs) in distributed environments is limited by both privacy concerns and high computational demands, which exceed the processing power available in most industrial edge computing devices.

The essential approach to resolving this dilemma is the establishment of a privacy-preserving, data-driven cargo source mining system. Precisely identifying the dispersed potential SRIT demand across different regions and cargo types through big data analytics, while ensuring the security of sensitive data from various stakeholders, enables the integration of multi-source logistics data without violating privacy regulations. This not only enhances the accuracy of demand prediction but also ensures that data from ports, railways, and customs can be effectively aggregated for optimal resource allocation, fostering efficient and secure collaborative decision-making. This is the essential path to support normalized block train operations and promote the sustainable development of SRIT [6]. Addressing this problem enables the enhancement of transport efficiency, promotes the optimization of the transport structure, and contributes to the achievement of the "Dual Carbon" goals.

Based on the practical business challenges in the context of SRIT, this study addresses privacy-constrained distributed intelligence by proposing a federated split learning framework tailored for large language models. Specifically, we design a novel vertically partitioned LLM architecture that performs model splitting at the embedding layer, which theoretically guarantees that raw textual data never leave local clients, thereby providing fundamental data sovereignty and privacy protection, while significantly reducing client-side computational complexity via embedding lookup. To cope with the pronounced feature heterogeneity across ports, railways, and customs systems, we further introduce a role-oriented ChatML-based semantic formatting and multi-source semantic alignment mechanism, which maps heterogeneous structured inputs into a unified high-dimensional semantic space and enables the server-side LLM to conduct consistent inference over distributed and non-independent and identically distributed (non-IID) data sources. Building upon this architecture, we propose a "lightweight client–heavyweight server" deployment paradigm, and both theoretical analysis and empirical results demonstrate that, compared with standard gradient-sharing federated learning approaches, the proposed framework substantially reduces communication overhead and achieves a more favorable communication–computation trade-off, thereby making the deployment of billion-parameter models feasible in bandwidth-constrained industrial networks, aligning with recent research trends that emphasize optimization approaches for advanced services in resource-constrained edge and IoT environments [7].

The remainder of this paper is organized as follows: Section 2 reviews the relevant literature and highlights the limitations of existing research; Section 3 covers the scenario problem and the model framework structure; Section 4 presents the experimental results and model evaluation; and Section 5 discusses future research directions and concludes the paper.

2 Literature Review

2.1 Distributed Learning Architectures and Privacy Preservation

In the context of SRIT, data resides in the local databases of different stakeholders (port, railway and customs), requiring a privacy-preserving distributed learning paradigm to address the issue of data sharing. Traditional Federated Learning (FL) approaches, represented by algorithms such as FedAvg [8] and FedProx [9], enable collaborative training by aggregating local model updates while keeping the raw data stored locally. However, these horizontal Federated Learning methods assume that all clients share the same feature space,

which is not valid in the SRIT scenario, where the data fields held by the three parties (ports, railways, and customs) differ significantly.

To address feature heterogeneity, Vertical Federated Learning (VFL) [10] and Split Learning (SL) [11] have been proposed. VFL allows different parties to hold different features for the same sample ID, aligning well with the multi-stakeholder nature of logistics. However, traditional VFL frameworks typically rely on exchanging intermediate gradients or encrypted values for simple models (e.g., Logistic Regression or XGBoost), which becomes computationally prohibitive and communication-intensive when applied to the massive parameter space of Large Language Models (LLMs). Furthermore, standard Split Learning, which partitions a deep network layer-wise between client and server, often suffers from high communication latency due to the frequent exchange of heavy activation maps during forward and backward propagation.

While formal privacy-preserving techniques such as Differential Privacy [12] and Homomorphic Encryption [13] offer theoretical security guarantees, their integration into LLM training imposes significant computational overhead and precision loss, rendering them impractical for resource-constrained industrial edge devices. Consequently, the proposed FSL-Qwen framework adopts a hybrid Federated Split Learning architecture optimized for LLMs. By partitioning the model strictly at the embedding layer, it offers a distinct advantage over standard VFL and SL. Clients are only required to perform $O(1)$ lightweight embedding lookups, avoiding heavy matrix multiplications. Moreover, the transmission of low-dimensional embedding vectors significantly reduces communication bandwidth compared to transmitting full model gradients or deep-layer activations.

2.2 Large Language Models in Collaborative Intelligence

While the distributed architectures discussed above offer pathways for privacy preservation, the specific integration of Large Language Models (LLMs) into such collaborative frameworks introduces distinct challenges regarding computational overhead and semantic alignment [14]. Large Language Models (LLMs) have demonstrated exceptional capabilities in knowledge extraction, reasoning, and generalization across diverse domains. As their adoption expands, evaluating the inherent characteristics and reliability of these generative models has become a critical research focus [15]. However, deploying these billion-parameter models in decentralized industrial scenarios (e.g., connecting port and railway systems) presents significant bottlenecks.

First, the substantial memory footprint and computational requirements of LLMs often exceed the capabilities of edge devices deployed at logistics nodes. While model compression techniques such as quantization and pruning [16] can reduce resource demands, they typically necessitate access to the full model or dataset for calibration, which contradicts the privacy constraints of our multi-party scenario. Consequently, a collaborative inference paradigm that offloads heavy computation to a server while keeping sensitive feature extraction local is required, yet standard split learning approaches often suffer from excessive communication overhead when applied to deep Transformer architectures.

Second, integrating multi-source data from autonomous entities involves addressing severe semantic heterogeneity [17]. Data from ports, railways, and customs possess distinct feature spaces and terminologies. Direct concatenation of such heterogeneous text often leads to hallucinations or context confusion in LLMs due to the lack of explicit role definition and semantic boundaries. Therefore, developing a mechanism to align these disparate feature embeddings into a coherent semantic space, while ensuring that raw private data is not shared, remains a critical challenge in distributed collaborative intelligence.

2.3 Data Heterogeneity and Cross-Domain Correlation in SRIT

While the aforementioned semantic alignment challenges are universal to distributed collaborative learning, the SRIT domain introduces complex, real-world constraints that further exacerbate these difficulties. Unlike general distributed scenarios, the highly heterogeneous data characteristics inherent to SRIT, ranging from unstructured port logs to structured railway manifests, impose distinct requirements on the model architecture. Although the integration of distributed collaborative paradigms and large language models establishes a theoretical foundation for privacy-preserving intelligence, the highly heterogeneous data characteristics inherent to SRIT impose additional, domain-specific constraints on model architecture. Research on freight demand forecasting has evolved from traditional econometric models to deep learning approaches such as LSTMs and Transformers [18, 19]. However, these existing methods are predominantly confined to single-source data, typically relying solely on historical port throughput or railway waybill sequences to predict aggregate trends.

In the context of SRIT, traditional single-source data processing paradigms fail to effectively capture the complex, non-linear correlations across disparate domains. The identification of potential container sources is inherently a dynamic process, driven by the coupled interactions among port vessel scheduling, spatiotemporal changes in yard storage, railway capacity availability, and customs clearance efficiency. These factors do not exist in isolation but are highly entangled. Furthermore, data from different stakeholders exhibits significant semantic heterogeneity: ports primarily record vessel arrival and container logistics, railways manage train schedules and waybills, while customs track cargo ownership and flow direction. Such intrinsic disparities make

it difficult for traditional centralized models or shallow federated networks to effectively integrate heterogeneous data from diverse sources. Consequently, there is a lack of frameworks capable of simultaneously preserving privacy while explicitly modeling, aligning, and fusing these multi-source heterogeneous features to support fine-grained cargo source identification. Our proposed FSL-Qwen framework is designed specifically to address these practical business pain points, overcoming the dual challenges of cross-domain data integration and privacy preservation.

3 Methodology

3.1 Problem Formulation

We formulate the potential container source identification in SRIT as a distributed inference problem over vertically partitioned data. Consider a system with $K = 3$ distinct participants: Port (P), Railway (R), and Customs (C). Each participant $k \in \{P, R, C\}$ holds a private local dataset $\mathcal{D}_k = \{x_i^{(k)}\}_{i=1}^{N}$, where N is the number of samples (containers), and $x_i^{(k)} \in \mathbb{R}^{d_k}$ represents the local feature vector for the i-th sample with dimension d_k.

In this scenario, the full feature set for a specific container $x_i = \{x_i^{(P)}, x_i^{(R)}, x_i^{(C)}\}$ captures the complete lifecycle information, including port container information, railway scheduling data, and customs declarations. However, due to strict data sovereignty and privacy barriers [20], the joint feature vector x_i cannot be aggregated centrally. Each party k must keep $x_i^{(k)}$ local. The target variable y_i (representing the optimal transport mode and destination) is inferred based on the collaborative information.

The objective of this study is to learn a mapping function $\mathcal{F} : \{x^{(P)}, x^{(R)}, x^{(C)}\} \to y$ that maximizes the conditional probability $P(y|x^{(P)}, x^{(R)}, x^{(C)})$ without physically sharing the raw features $x^{(k)}$. We frame this as a multi-source sequence-to-sequence generation task, where the heterogeneous inputs from three domains are encoded into latent embeddings and fused to generate a structured decision sequence containing the transport mode, destination, and rationale.

3.2 Federated Split Learning with Qwen Framework

Drawing upon the hybrid architecture concepts proposed by Zheng et al. [21], our FSL-Qwen framework integrates the advantages of Federated Learning (FL) and Split Learning (SL) through a novel pruning and reconstruction strategy for the Qwen2.5 large language model [22]. As illustrated in Figure 1, to resolve the conflict between the colossal computational demands of LLMs and the resource constraints of edge clients, this framework implements a vertical partitioning strategy strictly at the embedding layer.

Let $\mathcal{M}$ denote the global Qwen2.5 model, which consists of an embedding layer E, a stack of H Transformer layers $\mathcal{T}_{1\ldots H}$, and a language modeling head $\mathcal{H}$. In standard deployments, the entire $\mathcal{M}$ resides on a single device. In our FSL-Qwen framework, we decompose the model function $y = \mathcal{M}(x)$ into two distinct physical stages:

$$y = \underbrace{[\mathcal{H} \circ \mathcal{T}_H \circ \ldots \circ \mathcal{T}_1]}_{\text{Server-side}} \Big(\underbrace{E(x)}_{\text{Client-side}} \Big) \tag{1}$$

where the operator $\circ$ denotes functional composition, representing the sequential processing through the stacked Transformer layers. Specifically, the lightweight embedding function $E(\cdot)$ is deployed on each client k (e.g., ports, railways, customs). It functions as a local feature extractor, transforming the preprocessed expert text sequences $x^{(k)}$ into high-dimensional semantic vectors $\mathbf{Z}^{(k)} = E(x^{(k)}) \in \mathbb{R}^{L \times d}$. This ensures that raw business data is converted into a unified latent representation without leaving the local domain.

The collaborative training process mainly includes three core steps. First, in forward propagation, clients independently compute the embedding vectors $\mathbf{Z}^{(k)}$ for their respective data and upload them to the server. The server then fuses these embeddings and executes the deep Transformer computations to generate predictions. Second, in backward propagation, the server calculates the loss and gradients. The gradients with respect to the embeddings, $\nabla \mathbf{Z}^{(k)}$, are transmitted back to the clients, which then update their local embedding parameters. Third, in federated synchronization, the server periodically synchronizes its Token Embedding weights with all clients. This synchronization ensures consistency in feature representations across all participants, preventing semantic misalignment during distributed training.

3.2.1 Multi-source Semantic Alignment

To address the challenge of integrating heterogeneous data from ports, railways, and customs into a unified semantic space without sharing raw data, we propose a two-stage alignment strategy: localized embedding computation and ChatML-based federated input construction.

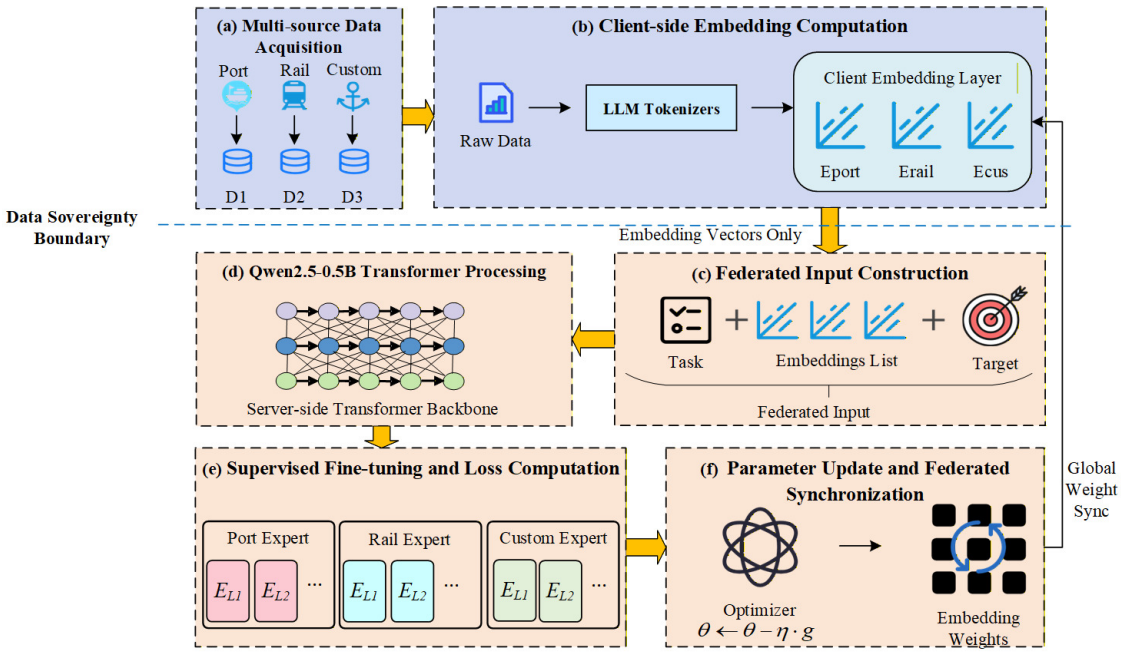


Figure 1: Architectural Overview of the FSL-Qwen Framework

The first phase is to embed computing on the client side. The embedding layer of the pre-trained LLM Qwen2.5 is pruned and deployed on local clients. This design ensures that raw operational data is transformed into latent feature representations locally. Let $x^{(c)} = \{t_1, t_2, \ldots, t_n\}$ denote the tokenized sequence of expert business data for client $c \in \{\text{port}, \text{railway}, \text{customs}\}$. The local embedding layer maps these tokens into a dense vector space:

$$E_c = \text{Emb}_c(x^{(c)}) \in \mathbb{R}^{n \times d} \tag{2}$$

where Emb_c denotes the local embedding function, and d represents the embedding dimension. Crucially, the embedding vectors E_c encapsulate the core semantic information while obfuscating the precise numerical or textual content, ensuring privacy preservation before transmission.

The second phase is the construction of federal input. Upon receiving the embedding vectors $\{E_{\text{port}}, E_{\text{railway}}, E_{\text{customs}}\}$, the server integrates them into a single sequence using the ChatML conversational format [23]. To resolve semantic ambiguity across different entities, we introduce role-specific identifiers:

- **Object Reference Tokens:** `<|object_ref_start|>` and `<|object_ref_end|>` delimit the information blocks.
- **Role Identifiers:** `<|Port Expert|>`, `<|Railway Expert|>`, and `<|Customs Expert|>` explicitly define the semantic source.

This structured design allows the server-side LLM to recognize the boundaries and roles of different data sources, thereby achieving precise cross-domain feature fusion. The detailed procedure is outlined in Algorithm 1.

3.2.2 Federated Optimization and Synchronization

The training process involves collaborative forward and backward propagation between clients and the server. To ensure the convergence of the distributed model, we implement a Federated Synchronization Mechanism.

During backpropagation, the gradients with respect to the embeddings, ∇E_c, are computed by the server and transmitted back to specific clients. Clients update their local parameters θ_c using standard gradient descent. However, since the client-side embedding layers are physically separated from the server, their parameters may diverge over time. To address this issue, the server distributes its updated global Token Embedding weights $W_{\text{emb}}^{\text{server}}$ to all clients after each training round. This synchronization guarantees that all parties perform feature representation within a unified semantic space, which is essential for the stability of the FSL-Qwen framework. The complete training procedure is summarized in Algorithm 2.

Algorithm 1 Federated Input Generation Process

Input: Client local data $\{\mathcal{D}_c\}_{c\in C}$, task instruction I
Output: Federated input E_{fed}

// Stage 1: Client-side embedding computation
for each client $c \in C$ **do**
  $x^{(c)} \leftarrow \text{Tokenize}(\mathcal{D}_c)$;
  $E_c \leftarrow \text{Emb}_c(x^{(c)})$; ▷ Local privacy-preserving projection
end for
// Stage 2: Server-side federated alignment
$E_{fed} \leftarrow \text{Emb}(I)$;
$E_{fed} \leftarrow E_{fed} + [\texttt{<obj\_start>}, \texttt{<Port Expert>}, E^{(\text{port})}, \texttt{<obj\_end>}]$;
$E_{fed} \leftarrow E_{fed} + [\texttt{<obj\_start>}, \texttt{<Railway Expert>}, E^{(\text{railway})}, \texttt{<obj\_end>}]$;
$E_{fed} \leftarrow E_{fed} + [\texttt{<obj\_start>}, \texttt{<Customs Expert>}, E^{(\text{customs})}, \texttt{<obj\_end>}]$;
return E_{fed};

Algorithm 2 Federated Split Learning Optimization

Input: Training data $\mathcal{D}$, set of clients C
Output: Global model parameters θ

Initialize server-side model parameters θ;
Synchronize embedding weights W_{emb} to all clients;
for $t = 1, 2, ..., T$ **do**
  // Forward Propagation
  for each client $c \in C$ **do**
    $E_c \leftarrow \text{ClientEmbedding}_c(\mathcal{D}_c)$;
    Upload E_c to server;
  end for
  $E_{fed} \leftarrow \text{ConstructFederatedInput}(\{E_c\}_{c\in C})$;
  $\mathcal{L} \leftarrow \text{ComputeLoss}(E_{fed}, y)$; ▷ See Appendix B for Loss details
  // Backward Propagation
  $\theta \leftarrow \theta - \eta\nabla_\theta\mathcal{L}$; ▷ Server update
  for each client $c \in C$ **do**
    Receive ∇E_c from server;
    Update local parameters θ_c;
  end for
  // Consistency Synchronization
  $W_{emb}^c \leftarrow W_{emb}^{\text{server}}$ for all $c \in C$;
end for
return θ;

3.3 Theoretical Analysis of Efficiency

To demonstrate the computational advantages of FSL-Qwen, we perform a comparative analysis against the vertical federated framework VFed-PU proposed by Huang et al. [24]. We analyze the complexity from two perspectives: Data Alignment and Model Training.

3.3.1 Alignment Complexity Analysis

In multi-party logistics, aligning samples (e.g., matching a container at the port with a railway waybill) is the prerequisite for training. VFed-PU employs Private Set Intersection (PSI) based on Diffie-Hellman key exchange [25]. Let N be the dataset size, and T_{exp} be the time cost of one modular exponentiation operation, which is known to be computationally expensive.

1. VFed-PU (Cryptographic Alignment): The complexity is dominated by the encryption operations required for intersection. According to the protocol in VFed-PU, each party must perform exponentiation for

every sample twice. The total alignment time complexity is:

$$T_{align}^{VFed} \approx \mathcal{O}(N \log N \cdot T_{exp}) + \mathcal{O}(m + k) \tag{3}$$

where the second term represents Bloom filter operations (m is filter size, k is hash functions). As N grows to millions of logistics records, the T_{exp} term creates a significant pre-processing bottleneck.

2. FSL-Qwen (Semantic Alignment): Our framework avoids the use of cryptographic intersection. In contrast, we leverage the LLM's robust contextual understanding capabilities to perform semantic alignment through the ChatML formatting method. The complexity of this process is linear with respect to the total sequence length L:

$$T_{align}^{FSL} \approx \mathcal{O}(N \cdot L) \tag{4}$$

Since modular exponentiation T_{exp} involves complex large-integer arithmetic, $T_{exp} \gg 1$. Therefore, $T_{align}^{FSL} \ll T_{align}^{VFed}$. Our approach transforms the $\mathcal{O}(N \log N)$ cryptographic overhead into a simple $\mathcal{O}(N)$ string formatting process, significantly reducing system initialization latency.

3.3.2 Client-side Computational Load

We explicitly compare the burden on edge clients (e.g., Port/Railway servers). VFed-PU employs a guest model on the client side to extract representations [26], whereas FSL-Qwen employs a pure Embedding Layer.

(1) VFed-PU (Deep Guest Model): Let the Guest Model consist of K hidden layers, and let d_{proj} denote the characteristic projection dimension. The computation for a single token involves matrix multiplications:

$$C_{client}^{VFed} \approx \sum_{k=1}^{K} \mathcal{O}(d_{in}^{(k)} \cdot d_{out}^{(k)}) \approx \mathcal{O}(K \cdot d_{proj}^2) \tag{5}$$

This quadratic complexity $\mathcal{O}(d_{proj}^2)$ accumulates with network depth.

(2) FSL-Qwen (Embedding Lookup): Our client model is strictly an embedding look-up table. For a sequence of length L, the operation is a direct memory access (DMA) to fetch rows from the weight matrix W_{emb}. There are no floating-point multiplications (FLOPs) involved in the projection:

$$C_{client}^{FSL} \approx \mathcal{O}(L \cdot 1) \tag{6}$$

We define the Client Efficiency Ratio η as the ratio of computational complexity between the baseline and our method:

$$\eta = \frac{C_{client}^{VFed}}{C_{client}^{FSL}} \approx \frac{K \cdot d_{proj}^2}{L \cdot 1} \tag{7}$$

In practical Vertical Federated Learning deployments, the client-side projection dimension d_{proj} is often smaller than the server-side model dimension. However, even under a conservative estimate where d_{proj} is moderate (e.g., $d_{proj} = 128$), the quadratic complexity of matrix multiplication $\mathcal{O}(d_{proj}^2)$ remains computationally significantly heavier than the constant-time memory access $\mathcal{O}(1)$ of the embedding lookup in FSL-Qwen. Consequently, η remains consistently greater than 1, ensuring that our framework shifts the bottleneck from computation to lightweight memory access, which is theoretically optimal for resource-constrained edge nodes.

Table 1: Comparison of Computational and Communication Complexity

Framework	Alignment Mechanism	Client Computation	Comm. Overhead	Scalability
Standard FedAvg	N/A (Assumes Aligned)	High ($\mathcal{O}(M_{param})$)	High (Full Gradients)	Low
VFed-PU	Cryptographic PSI ($\mathcal{O}(N \log N)$)	Medium ($\mathcal{O}(K \cdot d_{proj}^2)$)	Medium (Hidden Reps)	Medium
FSL-Qwen (Ours)	**Semantic ChatML ($\mathcal{O}(N \cdot L)$)**	**Lowest ($\mathcal{O}(1)$, Lookup)**	**Low ($\mathcal{O}(L \cdot d)$)**	**High**

3.4 Privacy and Security Analysis

To evaluate the security boundaries of FSL-Qwen, we formulate the data reconstruction attack as an optimization problem and assess privacy leakage through information-theoretic metrics. This analysis explicitly contrasts our inherent structural isolation with established vertical federated frameworks such as VFed-PU [24] and vulnerabilities identified in recent Split Learning studies [27].

3.4.1 Formal Threat Model and Reconstruction Hardness

To rigorously evaluate the security boundaries, we define a formal threat model based on the honest-but-curious assumption. We consider a semi-honest server adversary $\mathcal{A}$ that operates under the honest-but-curious assumption. This adversary faithfully executes the FSL-Qwen protocol and returns correct gradients and outputs, while being bounded by polynomial-time computational resources, consistent with standard cryptographic assumptions. The adversary's primary objective is to infer the private client input x solely based on the received intermediate results, which include the forward-pass embeddings Z and backward-pass gradients ∇Z. regarding the knowledge scope, we adopt the standard white-box assumption for the server, granting the adversary full knowledge of the global model architecture, the training procedure, and the server-side parameters. In addition, we do not assume the server has access to the client-side embedding projection matrix W_{emb}, effectively treating it as a strictly local private key. Furthermore, the adversary is restricted from accessing the raw input data x, any intermediate activations resident within the client's local scope, or task-specific labels, except for information that is implicitly observable from the embedding semantics.

Based on this threat model, the adversary's objective is to solve the inverse problem defined as:

$$\hat{x} = \arg\min_{x' \in \mathcal{X}} \left(\mathcal{L}_{rec}(f(x'; \hat{W}), Z) + \lambda \mathcal{R}(x') \right) \tag{8}$$

where $\hat{W}$ is the adversary's estimate of the client's projection matrix, and $\mathcal{R}(\cdot)$ is a regularization term, such as the language fluency prior utilized to constrain the reconstruction search space [28].

Theorem 1 (Hardness under Unknown Projection): In FSL-Qwen, the projection matrix W_{emb} is kept strictly local and never shared. Unlike standard Split Learning where the first-layer parameters are often initialized commonly or inferred, our W_{emb} acts as a private key. The adversary faces a *Blind Inverse Problem*. Even if the adversary assumes a standard tokenizer, the mapping from discrete tokens to continuous vectors depends on the specific, potentially fine-tuned or permuted local W_{emb}. The search space for joint estimation of x and W is:

$$\mathcal{S} \cong \mathcal{V}^L \times \mathbb{R}^{|\mathcal{V}| \times d} \tag{9}$$

Recent studies on Feature-Oriented Reconstruction Attacks [27] demonstrate that reconstruction is feasible only when the adversary has "white-box" access to the client model parameters or "black-box" query access to the exact function. FSL-Qwen eliminates both conditions: parameters are hidden, and the client does not provide an oracle for arbitrary queries (only training gradients are returned).

3.4.2 Analysis of Gradient Leakage Resistance Mechanisms

Beyond direct reconstruction, another critical privacy metric is the resistance to gradient-based leakage, as exemplified by the Deep Leakage from Gradients (DLG) attack [29]. We analyze the gradient received by the client k during backpropagation:

$$\nabla Z^{(k)} = \frac{\partial \mathcal{L}}{\partial Z^{(k)}} = \sum_{j \in \mathcal{B}} \alpha_j \cdot \frac{\partial \mathcal{L}_{task}}{\partial h_{fused}} \tag{10}$$

The gradient $\nabla Z^{(k)}$ is derived from the server's loss $\mathcal{L}$, which is computed over the aggregated representation $h_{fused} = \text{Concat}(Z^{(P)}, Z^{(R)}, Z^{(C)})$. This operation inherently results in the superposition of gradients from multiple heterogeneous sources, thus the Signal-to-Noise Ratio (SNR) for reconstructing client k's data is:

$$\text{SNR}_k \propto \frac{||\nabla Z^{(k)}||^2}{\sum_{i \neq k} ||\nabla Z^{(i)}||^2 + \sigma_{sys}^2} \tag{11}$$

In the multi-modal SRIT scenario, the heterogeneity of Port, Railway, and Customs data ensures that the cross-terms in the gradient summation are non-trivial. Crucially, this mechanism serves as an effective alternative to traditional Differential Privacy strategies [30]. Whereas formal Differential Privacy guarantees protection by injecting artificial noise that inevitably degrades the semantic precision required for logistics reasoning, FSL-Qwen leverages the intrinsic heterogeneity of multi-source gradients as a natural obfuscator. This approach maintains high predictive utility while effectively lowering the Signal-to-Noise Ratio for potential reconstruction attacks.

3.4.3 Resistance to Model Inversion Attacks

We explicitly contrast our architectural defense against vulnerabilities found in Vertical Federated frameworks like VFed-PU. VFed-PU requires transmitting intermediate activation representations h_{deep} from a deep Guest Model (DNN). Recent benchmarks on Model Inversion Attacks (MIA) [31] indicate that deep intermediate representations act as semantic compressors, projecting high-dimensional inputs onto a low-dimensional manifold where data density is high.

To quantify this risk, we define the Inversion Uncertainty $\mathcal{U}(Z)$ as the entropy of the posterior distribution of the input x given the observed vector Z:

$$\mathcal{U}(Z) = -\int p(x|Z, \theta_{adv}) \log p(x|Z, \theta_{adv})\, dx \tag{12}$$

where θ_{adv} represents the adversary's prior knowledge.

(1) VFed-PU Case (Low Uncertainty): Deep networks are designed to be invariant to noise and sensitive to semantic features. As the layer depth l increases, the representation $h^{(l)}$ eliminates nuisance variations, effectively reducing the conditional entropy $\mathcal{U}(h^{(l)})$. Consequently, the adversary can leverage public datasets to train a shadow model $\mathcal{M}_{shadow} \approx \mathcal{M}_{client}$, leading to a converging reconstruction error:

$$\lim_{t\to\infty} ||\hat{x}_t - x_{true}||^2 \leq \epsilon_{manifold} \tag{13}$$

This implies that deep activations retain sufficient structural constraints to guide the inversion to a precise semantic vicinity.

(2) FSL-Qwen Case (High Uncertainty): In contrast, FSL-Qwen performs the split strictly at the embedding layer ($l = 0$). As established in Theorem 1, the mapping $Z = x \cdot W_{emb}$ constitutes an underdetermined linear system for an adversary lacking W_{emb}. Without the non-linear semantic filtering provided by deep layers, the embedding vector Z does not inherently lie on a predictable semantic manifold. The inversion task degenerates into a blind rotation problem with an unbounded error lower bound:

$$\mathbb{E}[||\hat{x}_{adv} - x_{true}||^2] \geq \sigma_W^2 \cdot \text{Trace}((Z^T Z)^{-1}) \tag{14}$$

where σ_W^2 represents the variance of the unknown projection weights. This inequality demonstrates that without the private key W_{emb}, the reconstruction error is mathematically bounded away from zero, effectively neutralizing the gradient-based optimization used in standard MIA.

4 Framework evaluation using practical data

4.1 Data Description and Preprocessing

The data utilized in this study is derived from multi-source, multi-dimensional business records and external supplementary information, covering the core segments of the SRIT key business process. First, the Port Operation Data originates from the Guangzhou Port Container Production Business System, recording the container's entire process from vessel arrival, loading/unloading, stacking, to outbound movement. Second, the Railway Transport Data comes from the China Railway 95306 Freight Waybill System, including elements such as the departure/arrival stations of container block trains passing through Nansha Port, cargo categories, freight rates, and transport transit times. Furthermore, the Customs Statistical Data encompasses declaration records, clearance progress, inspection results, and regulatory requirements, which reflect the efficiency and compliance dynamics of the cross-border segment. To further supplement and expand the data samples, external data was also introduced: this includes road transport transit time and distance information from the port to inland stations, collected via the Baidu Maps API, as well as records of price subsidies and block train commencement policies released during the study period. The overall data time span is from June 2022 to June 2024. The multi-stakeholder dataset exhibits significant feature heterogeneity and strict data isolation. Unlike standard public datasets, this real-world distribution provides a rigorous testbed for validating the effectiveness of Vertical Federated Learning in handling non-IID data. Specific field descriptions are presented in Table 2.

Table 2: Description of Data Sources

Data Source	Key Fields	Data Volume	Application Purpose
Guangzhou Port Container Production Business System	Container ID, Container Type, Container Size, Cargo Weight, Vessel Berthing Time, Unloading Operation Time, etc.	1048575	Describe the container circulation process within the port segment and support feature extraction for source generation.
China Railway 95306 Freight Waybill and Manifest System	Container ID, Departure/Arrival Station Code, Cargo Category, Freight Rate, Transport Transit Time, Consignor/Consignee, etc.	108672	Provide rail capacity and cargo flow information, supporting the modeling of potential container sources and freight rate constraints.
Guangzhou Customs Statistical Data	Declaration Form ID, Statistical Date, Cargo Type Code, Clearance Progress, Regulatory Requirements, etc.	62940	Reflect cross-border supervision and clearance efficiency, supplementing features related to cargo flow and compliance.
Baidu Maps API	Origin/Destination Coordinates, Transport Distance, Transit Time, Route Selection Strategy, etc.	442	Serve as supplementary variables for comparing transit time differences between SRIT and direct road transport.
Policy Documents and Announcements	Policy Type, Policy Execution Period, Policy Coverage Area, Subsidy Price, etc.	550	Reflecting the influence of incentive measures on container source selection.

To transform raw, heterogeneous business records into a format suitable for LLM collaborative training, we implemented a Federated Sample Generation pipeline. Within this framework, we utilize an Export Text Conversion mechanism to convert raw structured records into natural language narratives specific to each domain (e.g., "Port Expert", "Railway Expert" and "Customs Expert"). These narratives are then aligned using the Container Number as a unique identifier. The aligned samples are formatted according to the Qwen ChatML specification, using role-specific tokens (e.g., <|Port Expert|>) to delineate semantic boundaries. To ensure that the model learns the true SRIT logistics chain, we applied a multi-stage alignment process to the initial raw records. First, we applied multi-party intersection to retain only container IDs present simultaneously across Port, Railway, and Customs databases. Second, we enforced strict spatio-temporal continuity, filtering out records where the time gap between port arrival and rail departure exceeded valid transfer windows. Finally, we performed a quality integrity check to exclude samples with incomplete critical attributes. This rigorous process yielded 48,800 valid samples, which were then formatted into structured input-output pairs for model training and partitioned into training and testing sets at an 8:2 ratio.

4.2 Experimental Setup

The FSL-Qwen framework was deployed and evaluated using PyTorch, with Qwen2.5-0.5B-Instruct serving as the backbone. To rigorously simulate the computational constraints typical of edge logistics nodes, we adopted specific memory-efficient configurations during training. To evaluate the effectiveness of FSL-Qwen, we compare it against 4 representative paradigms: (1) Local Training: Uses only Port data with the same Qwen2.5-0.5B model and training configuration. (2) Centralized Training: Merges all data centrally and trains the full model. (3) Standard FedAvg [32]: Although designed

for horizontal partitioning, we adapted it for vertical scenarios by treating each stakeholder's feature subset as a separate client. FedAvg uses the same embedding dimension and Transformer backbone as FSL-Qwen. (4) **VFed-PU**, the federated framework for logistics employing a SplitNN architecture.The comprehensive hardware specifications, software environment, and hyperparameter settings, including learning rate, batch size, and epochs, are detailed in Table 3, with all methods using identical settings.

Table 3: Implementation Environment and Hyperparameter Settings for FSL-Qwen

Parameter	**Value / Description**
Hardware & Software Environment	
GPU	NVIDIA vGPU (32GB VRAM)
CPU / RAM	Intel Xeon / 256GB System Memory
Framework	Python 3.10, PyTorch 2.0, Transformers
Training Hyperparameters (FSL-Qwen)	
Base Model	Qwen2.5-0.5B-Instruct
Optimizer	AdamW (Weight Decay: 0.01)
Learning Rate	2×10^{-6}
Physical Batch Size	8
Gradient Accumulation	2 steps (Effective Batch Size: 16)
Max Seq. Length	1024 Tokens
Gradient Clipping	Max Norm 1.0
Epochs	1

Performance is assessed across two dimensions: (1) Predictive Utility, measured by Accuracy and Macro-F1 Score on the transport mode decision task; and (2) System Efficiency, quantified by Communication Cost (MB/epoch) and Client-side Peak Memory (GB), to empirically validate the theoretical advantage of our $\mathcal{O}(1)$ client-side complexity and minimal transmission overhead.

4.3 Results and Discussion

4.3.1 Comparative Performance Analysis

The performance evaluation on the test set (20% of the 48,800 aligned samples) confirms that FSL-Qwen achieves predictive utility comparable to centralized training while satisfying strict privacy constraints. Table 4 presents the comprehensive metrics across decision accuracy, precision, recall, and system efficiency.

In terms of predictive utility, the framework attains an accuracy of 94.0% and a Macro-F1 score of 94.1%, closely approximating the ideal upper bound of centralized training (94.9%). Furthermore, FSL-Qwen demonstrates a balanced performance profile with a Precision of 93.5% and a Recall of 94.8%. This high Recall is particularly critical for the SRIT scenario, indicating that the model successfully identifies nearly 95% of potential convertible cargo, thereby minimizing the instances where convertible cargo is incorrectly classified as non-convertible. Simultaneously, the high Precision ensures that the identified cargo sources are highly likely to accept rail transport, reducing operational waste caused by invalid solicitations. When compared to the vertical federated baseline, VFed-PU, our approach yields a significant improvement of approximately 7.0% in F1-score. This gain is attributed to the semantic alignment mechanism and the pre-trained knowledge inherent in the LLM backbone, which captures complex, non-linear logistics correlations more effectively than the shallow neural networks used in prior studies.

Beyond predictive accuracy, the experimental results validate the theoretical efficiency advantages derived from the proposed $\mathcal{O}(1)$ client-side complexity. As illustrated in Table 4, the client-side peak video memory (VRAM) usage is reduced to merely 0.26 GB, as clients are only required to maintain the lightweight embedding layer. In contrast, standard baselines like FedAvg or Local Training require loading the full 0.5B parameter model and optimizer states, consuming approximately 3.5 GB of

Table 4: Performance and Efficiency Comparison

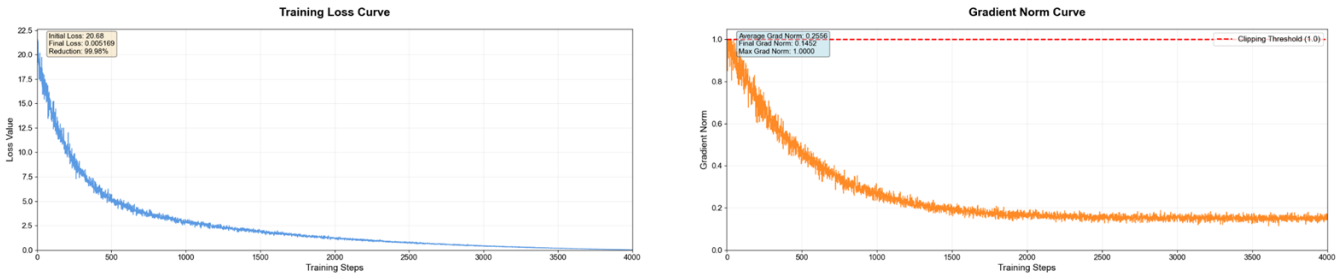
Method	Predictive Utility				System Efficiency			Privacy
	Acc	Prec.	Rec.	F1	Comm.[a]	VRAM[b]	Time[c]	Level
Lower/Upper Bounds								
Local Training (Port)	0.684	0.652	0.621	0.636	**0**	3.5 GB	1.2 h	High
Centralized (Ideal)	0.949	0.946	0.951	0.948	N/A	3.5 GB	1.5 h	None
Distributed Baselines								
Standard FedAvg	0.821	0.815	0.802	0.808	950 MB	3.5 GB	12.8 h	Medium
VFed-PU	0.875	0.862	0.881	0.871	45.2 MB	1.8 GB	4.5 h	Medium
FSL-Qwen (Ours)	**0.940**	**0.935**	**0.948**	**0.941**	**1.2 MB**	**0.26 GB**	**1.8 h**	**High**

[a] Communication cost per training step (MB/Step). FedAvg transmits full model weights.
[b] Peak Video RAM usage on the client side.
[c] Average total training time to reach convergence (hours).

VRAM, which often exceeds the capacity of edge devices at railway stations or customs checkpoints. Furthermore, the communication overhead is minimized to 1.2 MB per step, representing a reduction of orders of magnitude compared to the transmission of full model weights or deep intermediate activations required by other federated paradigms. This confirms that FSL-Qwen is a viable solution for deploying large-scale intelligence in resource-constrained industrial network environments.

4.3.2 Convergence and Stability Analysis

To assess the optimization stability of the vertically partitioned architecture, we analyze the training dynamics through the loss landscape and gradient norm evolution over 4,000 training steps, as visualized in Figure 2.



(a) Training Loss Curve (b) Gradient Norm Curve

Figure 2: Training Process Analysis

Figure 2a depicts the training loss trajectory. It can be observed that the model exhibits a smooth and monotonic decay, achieving rapid convergence within the initial 500 steps and steadily minimizing the loss over the full 4,000 steps. In contrast to standard Split Learning paradigms, which often suffer from oscillation due to the deep separation between client-side feature extraction and server-side supervision, our framework maintains a consistent optimization path. This indicates that the gradients are effectively backpropagated across the vertical cut layer without experiencing significant vanishing or instability.

Figure 2b further corroborates the system's numerical stability by tracking the gradient norm. Initially, the norm touches the clipping threshold (1.0) due to the cold start but rapidly decreases and stabilizes between 0.15 and 0.20. This low-variance behavior validates the effectiveness of the embedding alignment strategy. By mapping heterogeneous data (Port, Rail, Customs) into a unified semantic space, the framework prevents the feature divergence that typically disrupts training in non-IID distributed environments, thereby ensuring reliable weight updates throughout the process.

4.3.3 Ablation Studies

To verify the contribution of the proposed algorithmic components, we conducted ablation studies focusing on two critical dimensions. The first dimension examines the Semantic Alignment Strategy to evaluate the efficacy of the ChatML architecture, while the second dimension assesses the Multi-source Data Fusion to determine the necessity of vertical federation. Table 5 summarizes the quantitative results.

Table 5: Ablation Analysis of Algorithmic Components and Data Modalities

Configuration	**Acc**	**F1-Score**	**Δ F1 (Impact)**
FSL-Qwen (Full Framework)	**0.940**	**0.941**	**-**
A. Impact of Algorithmic Components (Embedding & ChatML)			
w/o ChatML Role Tokens	0.910	0.890	-5.1%
w/o Expert Text Conversion	0.880	0.855	-8.6%
B. Impact of Vertical Data Federation			
w/o Port Domain	0.890	0.852	-8.9%
w/o Railway Domain	0.920	0.884	-5.7%
w/o Customs Domain	0.910	0.875	-6.6%

The results in Group A explicitly address the efficacy of our proposed alignment mechanisms, responding to the need for evaluating embedding strategies and segmentation validity. Specifically, removing the role-specific tokens (e.g., `<|Port Expert|>`) and using raw vector concatenation leads to a 5.1% drop in F1-score. This degradation confirms that simply aggregating embeddings is insufficient for heterogeneous data. The ChatML tokens function as semantic anchors in the high-dimensional space, enabling the server-side Transformer to correctly attend to and distinguish between the feature distributions of different clients, thus validating the specific advantage of our ChatML-based segmentation over standard concatenation methods. Furthermore, the most severe algorithmic performance drop (-8.6%) occurs when the Expert Text Conversion module is removed, i.e., feeding raw database fields directly into the tokenizer. This finding validates our embedding layer configuration hypothesis: raw, unstructured database records lack the semantic density required for LLMs. By converting structured records into natural language narratives before embedding projection, we effectively bridge the gap between numerical business data and the LLM's pre-trained linguistic knowledge.

Group B verifies the hypothesis that SRIT cargo identification requires a holistic view of the logistics chain. Removing Port data causes the largest performance collapse (-8.9%), confirming that physical container attributes such as weight and berthing time are the primary determinants for transport mode feasibility. Meanwhile, the exclusion of Railway or Customs data results in moderate drops of 5.7% and 6.6%, respectively. While less dominant than Port data, their absence prevents the model from reasoning about capacity availability and compliance costs, which leads to increased False Positives in the Rail-Convertible prediction. This confirms that the vertical federated architecture provides information gains that localized models cannot achieve.

4.4 Case Study: Validation of Decision Logic

To evaluate the model's ability to internalize the underlying economic principles of Sea-Rail Intermodal Transport without relying on the memorization of training labels, we conducted batch inference on a dataset of 5,000 randomly sampled road freight records. This section analyzes the correlation between the predicted railway conversion probability and transport distance, serving as a proxy for verifying the consistency between the model's decision boundary and established logistics theories.

As illustrated in Figure 3, the inferred substitution probability exhibits a non-linear trend consistent with transport economics. For cargo transport distances below 500km, the model predicts a minimal substitution rate of approximately 2.3%. This suppression aligns with the industrial consensus that the high fixed handling costs of rail transport render short-haul routes less competitive compared to the flexibility of road transport. Conversely, as the distance exceeds the 800km threshold,

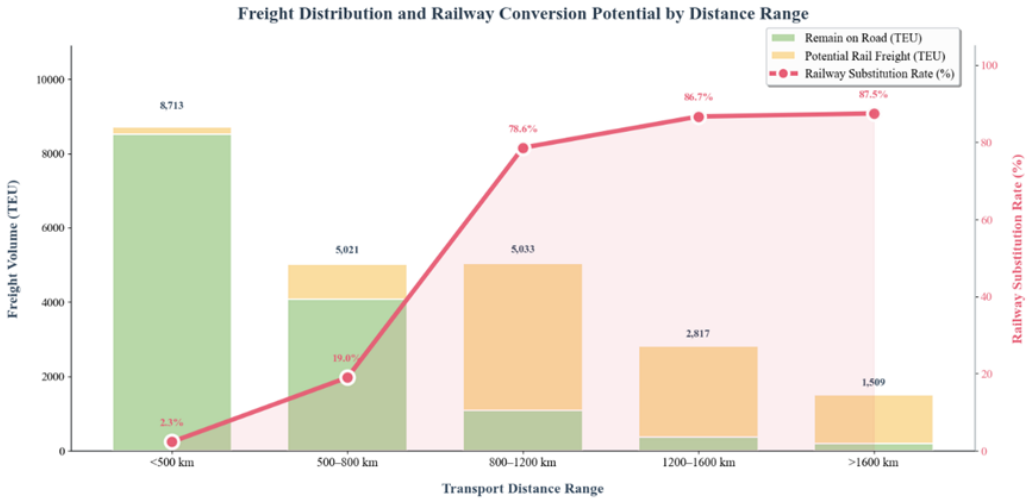


Figure 3: Correlation between transport distance and predicted railway substitution rate.

the substitution rate increases significantly to 82.5%. This distributional shift indicates that the model correctly identifies the critical breakeven point where the marginal cost advantage of rail transport outweighs the associated drayage and intermediate handling costs [33].

Significantly, the architecture operates without manually pre-defined distance constraints or heuristic rules. The observed decision logic demonstrates that FSL-Qwen, through supervised fine-tuning on the aligned semantic space, has autonomously captured the complex relationship between distance, cost, and mode selection. This alignment with economic theory confirms that the framework bases its predictions on valid logistics reasoning rather than spurious correlations, thereby ensuring reliability in practical deployment.

5 Conclusions

In this paper, we addressed the critical challenge of data silos in cross-border logistics by proposing FSL-Qwen, a vertical federated split learning framework powered by large language models. By implementing a novel semantic alignment strategy based on the ChatML protocol and expert text conversion, the framework successfully bridges the semantic gap among heterogeneous data sources from ports, railways, and customs without sharing raw privacy data. Extensive experiments on a large-scale real-world dataset demonstrated that the model achieves a predictive accuracy of 94.0% and an F1-score of 94.1%, effectively matching the performance upper bound of centralized training while surpassing state-of-the-art vertical federated baselines. Beyond predictive utility, the system exhibits superior computational efficiency by reducing communication overhead to 1.2 MB per step and limiting client-side memory usage to 0.26 GB, which validates the theoretical analysis of constant client-side complexity and confirms the feasibility for deployment on resource-constrained edge devices. Furthermore, the case study verified that the model has autonomously internalized complex transport economic principles regarding distance and cost constraints, ensuring that the decision logic is consistent with industrial consensus rather than relying on spurious correlations. Future work will focus on expanding the framework to support multi-modal inputs such as shipping document images and exploring advanced differential privacy mechanisms to further enhance security guarantees against potential gradient inversion attacks.

Acknowledgements

This research was funded by the National Natural Science Foundation of China (Grant No. 52172311) and the Fundamental Research Funds for the Central Universities (Grant No. 2024JBZX042).

Appendix A Server-side Transformer Backbone

The server-side model employs a standard 24-layer Transformer architecture. Each layer consists of a multi-head self-attention module and a feed-forward network (FFN). The computation for the l-th layer is defined as:

$$H^{(l)} = \text{LayerNorm}(H^{(l-1)} + \text{MultiHeadAttn}(H^{(l-1)})) \tag{15}$$

$$H^{(l)} = \text{LayerNorm}(H^{(l)} + \text{FFN}(H^{(l)})) \tag{16}$$

Detailed forward propagation steps are provided in Algorithm 3.

Algorithm 3 Server-side Transformer Forward (Standard)

Input: Federated input E_{fed}
Output: Logits
$H^{(0)} \leftarrow E_{fed}$;
for $l = 1$ **to** 24 **do**
    $H^{(l)} \leftarrow \text{TransformerBlock}(H^{(l-1)})$;
end for
$logits \leftarrow H^{(24)} W_{lm}$;
return $logits$;

Appendix B Supervised Fine-tuning Objective

We utilize a Supervised Fine-tuning (SFT) approach with a specific masking strategy. The loss is computed only on the decision tokens (output), masking the instruction and input tokens. The objective function is the standard cross-entropy loss:

$$\mathcal{L}_{\text{SFT}} = -\sum_{i=\text{start}}^{m} \log P(y_i \mid \mathbf{X}_{\text{cond}}, y_{<i}) \tag{17}$$

This ensures the model focuses on learning the logistics decision logic rather than memorizing the input prompts.

Appendix C Inference Strategy

During inference, we employ a beam search decoding strategy to enhance the stability of the generated decision sequences. The server iteratively predicts the next token based on the federated embeddings uploaded by clients. The standard inference procedure is outlined in Algorithm 4.

Algorithm 4 Inference Process

Input: New data $\mathcal{D}_{new}$
Output: Decision $\hat{y}$
Collect embeddings E_c from all clients;
$E_{fed} \leftarrow \text{ConstructFederatedInput}(\{E_c\})$;
$H \leftarrow \text{Transformer}(E_{fed})$;
$\hat{y} \leftarrow \text{BeamSearch}(H)$;
return $\hat{y}$;

References

[1] Luo, Y. (2024). Paradigm shift and theoretical implications for the era of global disorder, *Journal of International Business Studies*, 55(2), 127–135, 2024.

[2] McKibbin, W.; Fernando, R. (2023). The global economic impacts of the COVID-19 pandemic, *Economic Modelling*, 129, 106551, 2023.

[3] Liu, W.; Wang, L.; Yan, B.; Zhu, X.; Liu, Z. (2025). Integrated optimization of dynamic deployment and scheduling for rail-mounted gantry cranes in sea-rail intermodal port with on-dock rails, *Transportation Research Part E: Logistics and Transportation Review*, 202, 104312, 2025.

[4] Shenoy, D.; Bhat, R., ; Krishna Prakasha, K. (2025). Exploring privacy mechanisms and metrics in federated learning, *Artificial Intelligence Review*, 58(8), 223, 2025.

[5] Wang, Y.; Wu, Y.; Hao, C.; Hong, C. (2025). Research on the scheduling in sea-rail intermodal trains based on full-length and full-occupied strategy, *Research in Transportation Business & Management*, 59, 101301, 2025.

[6] Feng, C.; Lei, Y. (2024). Research on interval prediction method of railway freight based on big data and TCN-BiLSTM-QR, *IET Intelligent Transport Systems*, 18(12), 2713–2724, 2024.

[7] Montoya, G.; Lozano-Garzón, C.; Paternina-Arboleda, C.; Donoso, Y. (2025). A Mathematical Optimization Approach for Prioritized Services in IoT Networks for Energy-constrained Smart Cities, *International Journal of Computers Communications & Control*, 20(1).

[8] Zhang, X.; Mavromatis, A.; Vafeas, A.; Nejabati, R.; Simeonidou, D. (2023). Federated feature selection for horizontal federated learning in IoT networks, *IEEE Internet of Things Journal*, 10(11), 10095–10112, 2023.

[9] Gulati, S.; Guleria, K.; Goyal, N.; Alzubi, A. A.; Castilla, A. K. (2024). A privacy-preserving collaborative federated learning framework for detecting retinal diseases, *IEEE Access*, 12, 170176–170203, 2024.

[10] Gong, M.; Zhang, Y.; Gao, Y.; Qin, A. K.; Wu, Y.; Wang, S.; Zhang, Y. (2023). A multi-modal vertical federated learning framework based on homomorphic encryption, *IEEE Transactions on Information Forensics and Security*, 19, 1826–1839, 2023.

[11] Vepakomma, P.; Gupta, O.; Swedish, T.; Raskar, R. (2018). Split learning for health: Distributed deep learning without sharing raw patient data, *arXiv preprint*, arXiv:1812.00564, 2018.

[12] Xue, R.; Xue, K.; Zhu, B.; Luo, X.; Zhang, T.; Sun, Q.; Lu, J. (2023). Differentially private federated learning with an adaptive noise mechanism, *IEEE Transactions on Information Forensics and Security*, 19, 74–87, 2023.

[13] Alqazzaz, A. (2025). Federated Learning with Homomorphic Encryption: A Privacy-Preserving Solution for Smart Cities, *International Journal of Computational Intelligence Systems*, 18(1), 304, 2025.

[14] Piccialli, F.; Chiaro, D.; Qi, P.; Bellandi, V.; Damiani, E. (2025). Federated and edge learning for large language models, *Information Fusion*, 117, 102840, 2025.

[15] Zhou, H.; Inkpen, D.; Kantarci, B. (2024). Evaluating and mitigating gender bias in generative large language models, *International Journal of Computers Communications & Control*, 19(6).

[16] Dantas, P. V.; Cordeiro, L. C.; Junior, W. S. (2025). A review of state-of-the-art techniques for large language model compression, *Complex & Intelligent Systems*, 11(9), 407, 2025.

[17] Putrama, I. M.; Martinek, P. (2024). Heterogeneous data integration: Challenges and opportunities, *Data in Brief*, 56, 110853, 2024.

[18] Mokhtari, Z.; Amani-Beni, M.; Asgarian, A.; Russo, A.; Qureshi, S.; Karami, A. (2023). Spatial prediction of the urban inter-annual land surface temperature variability: An integrated modeling approach in a rapidly urbanizing semi-arid region, *Sustainable Cities and Society*, 93, 104523, 2023.

[19] Peng, T.; Gan, M.; Ou, Q.; Yang, X.; Wei, L.; Rødal Ler, H.; Yu, H. (2024). Railway cold chain freight demand forecasting with graph neural networks: A novel GraphARMA-GRU model, *Expert Systems with Applications*, 255, 124693, 2024.

[20] Liang, X. (2025). Cross-border logistics risk warning system based on federated learning, *Scientific reports*, 15(1), 39131, 2025.

[21] Zheng, J.; Chen, Y.; Lai, Q. (2024). PPSFL: Privacy-Preserving Split Federated Learning for heterogeneous data in edge-based Internet of Things, *Future Generation Computer Systems*, 156, 231–241, 2024.

[22] Qin, J.; Zhang, X.; Liu, B.; Qian, J. (2023). A split-federated learning and edge-cloud based efficient and privacy-preserving large-scale item recommendation model, *Journal of Cloud Computing*, 12(1), 57, 2023.

[23] Li, Y.; Yan, Y.; Tong, Z.; Wang, Y.; Yang, Y.; Bai, M., et al. (2025). Efficient fine-tuning of small-parameter large language models for biomedical bilingual multi-task applications, *Applied Soft Computing*, 175, 113084, 2025.

[24] Huang, L.; Jiang, D.; Zhang, X.; Wang, Y.; Bai, T. (2025). VFed-PU: Identifying Containers with Potential to be Shipped by Rail from Ports with Privacy Protection, *Tehnički vjesnik*, 32(2), 485–494, 2025.

[25] He, Y.; Tan, X.; Ni, J.; Yang, L. T.; Deng, X. (2022). Differentially private set intersection for asymmetrical id alignment, *IEEE Transactions on Information Forensics and Security*, 17, 3479-3494, 2022.

[26] Romanini, D.; Hall, A. J.; Papadopoulos, P.; Titcombe, T.; Ismail, A.; Cebere, T.; Hoeh, M. A. (2021). Pyvertical: A vertical federated learning framework for multi-headed splitnn, *arXiv preprint*, arXiv:2104.00489, 2021.

[27] Xu, X.; Yang, M.; Yi, W.; Li, Z.; Wang, J.; Hu, H.; Liu, Y. (2024). A stealthy wrongdoer: Feature-oriented reconstruction attack against split learning, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12130–12139, 2024.

[28] Qiu, Y.; Liu, Y.; Yu, H.; Fang, H.; Chen, B.; Xia, S. T.; Xu, K. (2025). Revisiting the Privacy Risks of Split Inference: A GAN-Based Data Reconstruction Attack via Progressive Feature Optimization, *arXiv preprint*, arXiv:2508.20613, 2025.

[29] Zhu, L.; Liu, Z.; Han, S. (2019). Deep leakage from gradients, *Advances in neural information processing systems*, 32, 2019.

[30] Vepakomma, P.; Swedish, T.; Raskar, R.; Gupta, O.; Dubey, A. (2018). No peek: A survey of private distributed deep learning, *arXiv preprint*, arXiv:1812.03288, 2018.

[31] Yang, W.; Wang, S.; Wu, D.; Cai, T.; Zhu, Y.; Wei, S.; Li, Y. (2025). Deep learning model inversion attacks and defenses: a comprehensive survey, *Artificial Intelligence Review*, 58(8), 242, 2025.

[32] McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data, *Artificial intelligence and statistics*, PMLR 1273–1282, 2017.

[33] Krauth, M.; Ribesmeier, M.; Bešinović, N. (2025). Optimising mode choice in a bi-modal freight network considering sustainability and urban logistic stakeholder perspectives, *Transportation Research Interdisciplinary Perspectives*, 31, 101442, 2025.

Cite this paper as:

MA, W.; Huang, L.; Zhang, Q.; Wang, Y.; Zhang, X.; Song, R. (2026). Federated Split Learning with Large Language Models Integration: A Study on Potential Container Source Identification in Sea-Rail Intermodal Transport, *International Journal of Computers Communications & Control*, 21(3), 7263, 2026.

https://doi.org/10.15837/ijccc.2026.3.7263