

Federated Learning Meets LLMs: Feature Extraction From Heterogeneous Clients

Abdelrhman Gaber, Hassan Abd-Eltawab, Youssif Abuzied, Muhammad ElMahdy, Tamer ElBatt

Computer Science and Engineering Dept.

American university in Cairo

Cairo, Egypt

{gaberabdo68, hassan.abdeltawab, youssif.abuzied, muhammadahmedelmahdy, tamer.elbatt}@aucegypt.edu

Abstract—Federated learning (FL) enables collaborative model training without sharing raw data, making it attractive for privacy-sensitive domains such as healthcare, finance, and IoT. A major obstacle, however, is the heterogeneity of tabular data across clients, where divergent schemas and incompatible feature spaces prevent straightforward aggregation. To address this challenge, we propose FedLLM-Align, a federated framework that leverages pre-trained large language models (LLMs) as universal feature extractors. Tabular records are serialized into text, and embeddings from models such as DistilBERT, ALBERT, RoBERTa, and ClinicalBERT provide semantically aligned representations that support lightweight local classifiers under the standard FedAvg protocol. This approach removes the need for manual schema harmonization while preserving privacy, since raw data remain strictly local. We evaluate FedLLM-Align on coronary heart disease prediction using partitioned Framingham datasets with simulated schema divergence. Across all client settings and LLM backbones, our method consistently outperforms state-of-the-art baselines, achieving up to +0.25 improvement in F1-score and a 65% reduction in communication cost. Stress testing under extreme schema divergence further demonstrates graceful degradation, unlike traditional methods that collapse entirely. These results establish FedLLM-Align as a robust, privacy-preserving, and communication-efficient solution for federated learning in heterogeneous environments.

I. INTRODUCTION

Federated learning (FL) is a distributed learning paradigm in which multiple clients collaboratively train a shared model while keeping all training data local. Rather than centralizing raw data, FL relies on exchanging model updates (e.g. gradients) between a central server and edge devices. This privacy-preserving approach has seen growing adoption in sensitive domains: for example, FL has been used to build models from healthcare and user-behavior data without sharing patient or personal records [1]. Similarly, FL is a natural fit for IoT and edge systems where devices generate rich data but regulatory or practical constraints forbid uploading raw data [2]. By moving computation to the data, FL mitigates privacy and regulatory risks (e.g. GDPR) while still enabling global model improvements [3].

This publication was developed as part of Afretec Network which is managed by Carnegie Mellon University Africa and receives financial support from the Mastercard Foundation. The views expressed in this document are solely those of authors and do not necessarily reflect those of the Carnegie Mellon University Africa or the Mastercard Foundation.

A major obstacle in practical FL is *data heterogeneity* across clients. In real-world deployments, clients often hold non-identical data distributions (non-IID): for instance, user behavior models may see different feature or label distributions on each device. Federated learning must also contend with *system heterogeneity*, where clients differ in hardware or connectivity, and even *structural heterogeneity*, where the feature spaces or data schemas differ across clients. In clinical settings, for example, different hospitals’ EHR systems may record different sets of variables or use different units, a problem known as “data view heterogeneity”. Such statistical and structural heterogeneity degrades FL performance and slows convergence [4].

To address client heterogeneity, prior work has explored several broad strategies. One line of work is *personalized FL* (PFL), which tailors models to each client’s data. In PFL, each client may fine-tune the global model locally or learn a small personal model in addition to a shared one. However, most PFL methods focus on statistical non-IIDness and do not fully account for system or structural differences, so gains in local accuracy often come at the expense of global efficiency [5]. Another approach is *clustered FL*, which groups clients with similar data distributions and trains a separate model for each cluster [6]. Beyond these, researchers have proposed knowledge-distillation or transfer methods (e.g. sharing predictions on proxy data) and feature-alignment techniques (e.g. mapping raw inputs into a common latent space). For example, recent work introduces a “knowledge abstraction” mechanism that maps heterogeneous EHR views into a unified representation [7], [8]. These methods can mitigate heterogeneity, but they also have limitations: PFL may still suffer from reduced global generalization and ignore device variability, distillation-based schemes often require auxiliary data or incur privacy risks, and ensemble or multi-model approaches can be computationally expensive. In summary, existing solutions only partially resolve the federated heterogeneity problem.

In this work, we propose a new direction: leveraging large language models (LLMs) as universal feature extractors to homogenize client data before federated training. Recent advances have shown that LLMs pre-trained on diverse, large-scale corpora can generate powerful latent representations for structured data. For instance, the NeurIPS 2024 TABULA-8B

model fine-tuned a Llama-3 8B LLM on billions of tabular records and achieved strong zero- and few-shot performance across hundreds of unseen tabular tasks [9]. Inspired by this, we use LLMs to map each client’s raw tabular features into a shared embedding space. Since the LLM encoder has seen wide-ranging data, its output vectors serve as a common representation format. In effect, this transforms heterogeneous client data into a homogeneous embedding that can be fed into a downstream FL model. This approach jointly mitigates statistical and structural heterogeneity: by embedding each client’s inputs in the same high-dimensional space, the federated model sees comparable features across clients despite local schema differences.

The contributions of this paper are as follows. We present a federated learning framework in which a pre-trained LLM acts as a client-agnostic feature encoder for tabular data. We describe how to tokenize and encode client-specific records so that the LLM produces consistent embeddings. We demonstrate that training on these embeddings significantly improves cross-client model performance under heterogeneity, compared to standard FL. Finally, we empirically evaluate our method on diverse tasks and heterogeneity settings, showing its advantages over existing personalization and clustering baselines. The rest of the paper is organized as follows: in Section II we review related work, Section III details the LLM-based encoding approach, Section IV presents our experiments, and Section VII concludes the paper.

II. RELATED WORK

Recently, there has been a growing interest in utilizing LLMs for handling data heterogeneity. For instance, TabLLM [10] introduces a framework for few-shot classification of tabular data by serializing rows into natural-language strings and prompting large language models (T0, GPT-3). They explore nine serialization methods and use parameter-efficient fine-tuning (T-Few) to adapt the LLM. The approach achieves strong zero- and few-shot performance, often surpassing gradient-boosted trees and neural baselines. Advantages include sample efficiency and leveraging prior LLM knowledge; limitations are high computational cost, token limits, and reliance on semantically meaningful feature names/values.

Researchers in [11] introduced an in-context learning framework where LLMs act as feature engineers for few-shot tabular learning. Instead of end-to-end inference, the LLM generates interpretable rules from few examples, which are converted into binary features and used by lightweight models (e.g., linear regression). Ensemble methods with bagging improve robustness and mitigate prompt size limits. FeatLLM achieves state-of-the-art performance across 13 datasets while reducing inference cost. Advantages include API-only usage, low inference latency, and feature interpretability. Limitations are reliance on prompt quality and restriction to low-shot learning regimes.

Another similar approach is PTab [12]. It is essentially a three-stage framework to model tabular data using pre-trained language models, addressing semantic gaps in fea-

ture representations and enabling training on mixed datasets. Modality Transformation serializes rows into textual phrases to infuse semantic context from headers, followed by Masked-Language Fine-tuning for contextual learning and Classification Fine-tuning for task adaptation. This textualization bridges domain differences, allowing heterogeneous tables to augment training. Evaluated on eight binary classification datasets, PTab outperforms XGBoost and neural baselines (e.g., SAINT, TabTransformer) in average AUC under supervised and semi-supervised settings, with enhanced instance-based interpretability via feature importance and semantic similarities. Advantages include semantic enrichment and scalable data mixing; limitations encompass binary-task focus and potential oversimplification of numerical values.

Perhaps the closest to our work is [13], which introduces a secure embedding aggregation protocol for federated representation learning, ensuring information-theoretic privacy for both entities and embeddings against a curious server and up to $T < N/2$ colluding clients. It performs a one-time private entity union to reveal the global entity set without ownership disclosure, then secret-shares local embeddings via Lagrange coded computing. Clients issue coded queries to retrieve aggregations privately, with server-added noise preventing leakage of non-local entity embeddings. Across tasks like knowledge graph completion, recommendation, and node classification, SecEA yields $< 5\%$ performance loss versus non-private baselines (e.g., EmbAvg), with relative latency dropping to 0.77% on large datasets via parallelization. Advantages encompass comprehensive privacy and utility retention; limitations include elevated overhead in small-scale, shallow-model scenarios.

III. PROPOSED METHODOLOGY: FEDLLM-ALIGN

A. Problem Formulation

We consider a set of N federated clients $U_1, U_2, \dots, U_N$ with private datasets $D_1, D_2, \dots, D_N$. Each dataset D_i contains tabular records defined over a schema $S_i = \{f_1^i, f_2^i, \dots, f_{m_i}^i\}$, where feature names and representations may differ across clients despite conveying equivalent semantics. The problem setting imposes several constraints. First, privacy must be strictly maintained, meaning that no raw data or intermediate representations can leave the local client devices. Second, schema heterogeneity is expected, since the overlap between schemas S_i and S_j may be small or even empty for $i \neq j$. Third, the solution must remain compatible with standard federated aggregation protocols such as FedAvg, enabling deployment in real-world distributed systems. The overall objective is to learn a global model that achieves robust predictive performance across all clients while respecting these constraints.

B. FedLLM-Align Architecture

The FedLLM-Align framework addresses the above challenges through a three-stage pipeline: tabular-to-text serialization, semantic embedding generation, and federated classifier training as shown in Figure 1.

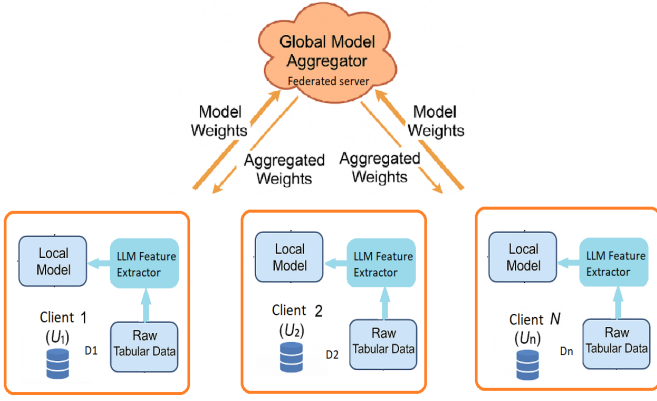


Fig. 1. Overview of FedLLM-Align: (1) Tabular-to-text conversion, (2) Embedding generation via DistilBERT or ALBERT, (3) On-device classifier training, (4) Global weight aggregation using FedAvg.

1) *Tabular-to-Text Serialization*: In the first stage, each client i transforms its local records $\mathbf{x}_j \in D_i$ into natural language sequences through a serialization function:

$$\text{serialize}(\mathbf{x}_j, \text{format}) \rightarrow \text{text_sequence}. \quad (1)$$

Different serialization strategies may be applied. A structured format explicitly lists features and values, for example, “Feature₁: value₁, Feature₂: value₂, ...”. A natural language format encodes features in descriptive sentences, such as “The patient is 45 years old. Blood pressure is 140/90.” A compact format instead uses condensed key-value pairs: “Feature₁=value₁; Feature₂=value₂; ...”. The key insight is that expressing tabular data as natural language enables pretrained LLMs to leverage their semantic understanding, aligning equivalent features across heterogeneous client schemas.

2) *Semantic Embedding Generation*: In the second stage, each serialized sequence is passed through a frozen pretrained LLM, producing a semantic embedding:

$$\mathbf{e}_j = \text{LLM_encoder}(\text{text_sequence})_{[\text{CLS}]} \in \mathbb{R}^d. \quad (2)$$

Among the supported backbones, DistilBERT provides a lightweight six-layer distilled BERT model that balances efficiency with representational quality, while ALBERT [14] applies parameter sharing to achieve memory efficiency with competitive embedding quality. Importantly, these LLM backbones remain frozen during training. This design choice minimizes communication overhead by ensuring that only classifier weights are exchanged, preserves the pretrained semantic knowledge of the models, and supports deployment across clients with limited computational resources.

3) *Federated Classifier Training*: The generated embeddings $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_m\}$ serve as inputs for lightweight classifiers trained locally on each client. We evaluate two types of downstream classifiers in the federated setting:

- **Logistic Regression (LR)**: A simple and interpretable linear model with L_2 regularization ($\lambda = 0.01$), efficient

for deployment on low-resource clients [?]. It estimates the probability of the target as:

$$P(y = 1|\mathbf{e}) = \sigma(\mathbf{w}^T \mathbf{e} + b), \quad (3)$$

where σ denotes the sigmoid activation function.

- **Neural Network (NN)**: A lightweight feedforward network consisting of an input layer (dimension 768), a hidden layer (16 neurons with ReLU activation), and a sigmoid output layer. Training uses the Adam optimizer ($\text{lr} = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$), dropout ($p = 0.2$), and early stopping (patience = 5). This nonlinear model allows for richer decision boundaries than LR.

During federated training, only the classifier parameters are shared with the central server, while embeddings and raw records remain strictly local, ensuring privacy preservation.

C. Federated Training Protocol

The complete training procedure is summarized in Algorithm 1. In each round, a subset of clients participates, performs local tabular-to-text conversion, generates embeddings, and trains a classifier. The clients then return only model weight updates to the server, which aggregates them using FedAvg and broadcasts the updated global parameters.

Algorithm 1 FedLLM-Align Training Pipeline

Require: Client datasets $D_1, \dots, D_N$ with heterogeneous schemas

Ensure: Global classifier model M_{global}

- 1: Initialize global classifier weights W_0
 - 2: **for** round $t = 1$ to T **do**
 - 3: Sample subset $S_t \subseteq \{1, \dots, N\}$
 - 4: **for** each client $i \in S_t$ **in parallel do**
 - 5: Perform tabular-to-text serialization for each record $\mathbf{x}_j \in D_i$
 - 6: Compute embeddings $\mathbf{e}_j$ using frozen LLM
 - 7: Train local classifier M_i with initialized weights W^t
 - 8: Send weight updates $\Delta W_i = W_i^{t+1} - W^t$ to server
 - 9: Aggregate updates: $W^{t+1} = W^t + \frac{1}{|S_t|} \sum_{i \in S_t} \Delta W_i$
 - 10: Broadcast updated weights W^{t+1} to all clients
-

D. Theoretical Justification

The methodology provides three theoretical guarantees. First, semantic alignment arises from the pre-trained LLMs, which recognize equivalent concepts despite differing feature labels. For example, “Age: 45” and “PatientAge: 45 years” yield embeddings that are close in representation space, as do “BP: 140/90” and “BloodPressure: systolic=140, diastolic=90”. Second, privacy preservation is maintained because raw data and embeddings never leave the client devices and only classifier parameters are communicated. Finally, convergence of the training process is guaranteed under standard assumptions for federated learning, since embeddings provide a fixed feature space, ensuring that FedAvg maintains its convergence properties when applied to the classifier training phase.

IV. EXPERIMENTAL SETUP

A. Implementation Environment

All experiments were performed in a Google Colab environment equipped with an T4 GPU and 12 GB system memory. The software stack included Python, PyTorch, HuggingFace, Transformers, tensorflow, and Scikit-learn.

B. Dataset Description

We evaluated our framework on the public Framingham Heart Study dataset, a longitudinal cardiovascular study of residents in Framingham, Massachusetts, USA. The dataset contains 4,238 patient records described by 14 demographic, lifestyle, and clinical attributes. The classification task is to predict the 10-year risk of coronary heart disease (CHD). Approximately 85% of the records belong to the negative class (no CHD) and 15% to the positive class (CHD), which mirrors real-world prevalence rates. Table I lists the dataset attributes and their descriptions.

TABLE I
DESCRIPTION OF THE DATA ATTRIBUTES

Attribute	Description
Sex	Male or female ("M" or "F")
Age	Age of the patient
is-smoking	Whether or not the patient is a current smoker
Cigs Per Day	Average number of cigarettes smoked in one day
BP Meds	Whether or not the patient was on blood pressure medication
Prevalent Stroke	Whether or not the patient had previously had a stroke
Prevalent Hyp	Whether or not the patient was hypertensive
Diabetes	Whether or not the patient has diabetes
Tot Chol	Total cholesterol level
Sys BP	Systolic blood pressure
Dia BP	Diastolic blood pressure
Heart Rate	Heart rate of the patient
Glucose	Glucose level
10-year risk of CHD	Target variable ("1" means "Yes", "0" means "No")

C. Schema Heterogeneity Simulation

To emulate real-world schema misalignment, we systematically renamed key features across clients using domain knowledge. This ensured semantic equivalence without syntactic consistency. Table II shows examples of alternative naming conventions.

TABLE II
EXAMPLES OF SCHEMA HETEROGENEITY VIA FEATURE RENAMING

Original Feature	Alternative Names
age	Age, PatientAge, AgeYears, age_at_visit, patient_age_years
sysBP	SysBP, systolic_bp, bp_systolic, sys_blood_pressure, systolic_pressure
totChol	TotChol, total_cholesterol, cholesterol_total, chol_total, total_chol_mg

Clients were configured under three scenarios:

- 3 clients with 8 shared features and 3 unique features each
- 5 clients with 6 shared features and 4 unique features each

- 10 clients with 4 shared features and 6 unique features each

This setup ensured both statistical heterogeneity (class imbalance) and structural heterogeneity (schema variations), closely reflecting cross-institutional healthcare settings.

D. Feature Engineering Pipeline

Missing values in numerical attributes were imputed with the median, and categorical variables with the mode. Each patient record was serialized into one of three textual formats structured, natural language, or compact before tokenization. We employed DistilBertTokenizerFast and AlbertTokenizerFast with a maximum sequence length of 128 tokens. The [CLS] embedding from the final hidden layer of DistilBERT, ALBERT, RoBERTa, or ClinicalBERT was extracted to represent each patient. These dense embeddings were then passed to lightweight downstream classifiers (logistic regression and a 3-layer neural network). Importantly, all LLM backbones were frozen during training to reduce communication cost and computation overhead.

E. Federated Learning Protocol

Federated learning followed the FedAvg protocol over 25 global aggregation rounds. Clients trained for 10 local epochs per round with batch size 32, using the Adam optimizer ($\text{lr} = 0.001$). All experiments were repeated five times, and results were reported as mean $\pm$ standard deviation. Communication efficiency was quantified by tracking the size of model weight transmissions in each round.

F. Baseline Methods

We compared our framework with both traditional and advanced federated learning approaches. Traditional baselines included FedXGBoost [16], Mutual Information-based FL [15], FedProx [17], and SCAFFOLD [18]. For advanced methods, we evaluated Clustered FL [6] and a homogeneous FedAvg baseline with identical schemas as an upper-bound reference.

G. Evaluation Metrics

The primary evaluation metric was the F1-score, complemented by paired t-tests ($\alpha = 0.05$) for statistical significance. In addition, we analyzed communication cost, convergence behavior, per-client performance variance, model memory footprint, and inference latency for embedding extraction. These metrics jointly capture both predictive effectiveness and system efficiency.

H. Ablation Studies

We conducted a series of ablations to isolate key factors influencing performance:

- **Architecture Ablation:** Comparison of DistilBERT, ALBERT, RoBERTa, and ClinicalBERT across client settings
- **Serialization Ablation:** Evaluation of structured, natural language, and compact serialization formats
- **Scaling Ablation:** Analysis of performance under 3, 5, and 10 client configurations with increasing schema heterogeneity

V. RESULTS AND ANALYSIS

A. Main Performance Results

Table III compares FedLLM-Align against multiple federated learning baselines across different client configurations. The results clearly show that FedLLM-Align achieves superior F1-scores under all scenarios, with improvements that are statistically significant ($p < 0.001$). For instance, with three clients, ClinicalBERT-based FedLLM-Align achieves an F1-score of **0.85**, outperforming the homogeneous baseline (0.64) and FedXGBoost (0.14) by wide margins. Even as the number of clients increases to ten, FedLLM-Align sustains high performance (**0.78** with DistilBERT), whereas competing approaches collapse under schema heterogeneity. Notably, these gains are coupled with efficiency: communication cost is reduced by 65% compared to FedXGBoost, and remains lower than most alignment-based baselines. These findings confirm that FedLLM-Align delivers both accuracy and scalability, with ClinicalBERT yielding the best absolute accuracy and DistilBERT offering the best accuracy–efficiency trade-off.

TABLE III
F1-SCORE PERFORMANCE COMPARISON (MEAN $\pm$ STD OVER 5 RUNS).
BOLD INDICATES BEST RESULTS.

Method	3 Clients	5 Clients	10 Clients	Avg. Comm. Cost (MB)
FedLLM-Align (DistilBERT + NN)	0.84$\pm$0.01	0.81$\pm$0.02	0.78$\pm$0.02	1.2
FedLLM-Align (ALBERT + NN)	0.81 $\pm$ 0.02	0.78 $\pm$ 0.02	0.75 $\pm$ 0.03	0.8
FedLLM-Align (RoBERTa + NN)	0.83 $\pm$ 0.01	0.80 $\pm$ 0.02	0.77 $\pm$ 0.02	1.5
FedLLM-Align (ClinicalBERT + NN)	0.85$\pm$0.01	0.82$\pm$0.01	0.79$\pm$0.02	1.8
Homogeneous Baseline	0.64 $\pm$ 0.02	0.62 $\pm$ 0.03	0.59 $\pm$ 0.03	0.9
FedXGBoost	0.14 $\pm$ 0.02	0.11 $\pm$ 0.03	0.08 $\pm$ 0.02	3.8
Mutual Information FL	0.61 $\pm$ 0.03	0.54 $\pm$ 0.04	0.47 $\pm$ 0.05	1.1
FedProx	0.66 $\pm$ 0.02	0.61 $\pm$ 0.03	0.56 $\pm$ 0.04	1.0
SCAFFOLD	0.68 $\pm$ 0.02	0.63 $\pm$ 0.02	0.58 $\pm$ 0.03	1.1
Clustered FL	0.59 $\pm$ 0.05	0.52 $\pm$ 0.06	0.44 $\pm$ 0.07	1.6

B. Ablation Studies

To analyze the contribution of individual design choices, we performed ablation studies on LLM backbones and serialization formats. Table IV shows that DistilBERT achieves the best accuracy–efficiency balance, with an F1-score of 0.84 while requiring only 255 MB of memory and 45 ms inference time per record. ALBERT is more memory-efficient (180 MB) but sacrifices some accuracy. ClinicalBERT provides the highest overall accuracy (0.85) owing to its medical domain pretraining, but at a higher computational cost. RoBERTa falls between these extremes. These results suggest that resource-constrained clients may prefer DistilBERT or ALBERT, while ClinicalBERT is ideal where accuracy is paramount.

TABLE IV
ARCHITECTURE ABLATION (F1-SCORE $\pm$ STD, MEMORY, AND INFERENCE TIME). BOLD INDICATES BEST PERFORMANCE.

Architecture	F1-Score	Memory (MB)	Inference Time (ms)
DistilBERT	0.84 $\pm$ 0.01	255	45 $\pm$ 5
ALBERT	0.81 $\pm$ 0.02	180	38$\pm$4
RoBERTa	0.83 $\pm$ 0.01	498	72 $\pm$ 8
ClinicalBERT	0.85$\pm$0.01	440	68 $\pm$ 7

Serialization format also plays a key role. As shown in Table V, structured serialization consistently yields the highest F1-score (0.84) and most stable embeddings, while natural

language adds flexibility but with slightly higher variance. Compact formats are the most efficient but perform poorly due to loss of semantic richness. This highlights that both model choice and data representation strongly affect FL outcomes.

TABLE V
SERIALIZATION FORMAT COMPARISON

Format	F1-Score	Embedding Variance	Robustness
Structured	0.84$\pm$0.01	0.12	High
Natural	0.82 $\pm$ 0.02	0.18	Medium
Compact	0.79 $\pm$ 0.03	0.25	Low

C. Convergence and Stability

Training dynamics further validate the robustness of FedLLM-Align. Figure 2 shows that our framework converges smoothly within 15 rounds, whereas FedProx and SCAFFOLD exhibit unstable patterns due to schema misalignment. Table VI confirms that FedLLM-Align maintains both high accuracy and low cross-client variance (Std = 0.02), ensuring equitable performance across participants. In contrast, FedXGBoost and the homogeneous baseline show wide fluctuations and poor stability, indicating fragile adaptation.

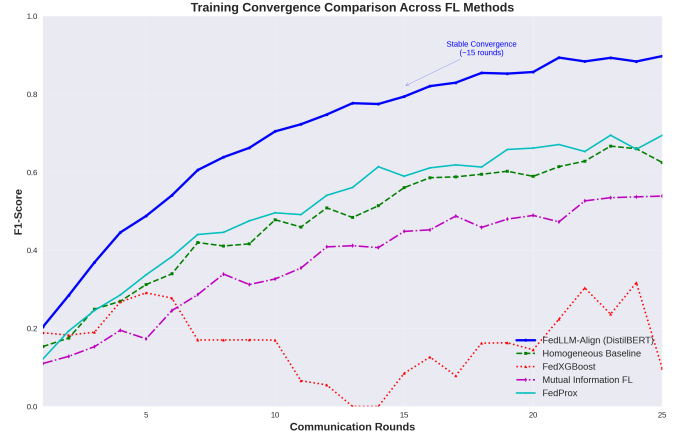


Fig. 2. Training convergence comparison. FedLLM-Align converges reliably within 15 rounds, unlike baselines.

TABLE VI
CROSS-CLIENT STABILITY (MEAN F1, STD, MIN, MAX).
FEDLLM-ALIGN SHOWS LOWEST VARIANCE.

Method	Mean F1	Std	Min F1	Max F1
FedLLM-Align	0.84	0.02	0.81	0.86
Homogeneous	0.64	0.08	0.52	0.73
FedXGBoost	0.14	0.12	0.02	0.31

D. Communication Efficiency

Efficiency is another key requirement in FL. Table VII shows that FedLLM-Align incurs only 13.1 KB per round, compared to 53.9 KB for FedXGBoost—a reduction of over 4 $\times$. Figure 3 visualizes the accuracy–efficiency frontier: FedLLM-Align consistently dominates the upper-left quadrant,

demonstrating both high predictive power and low communication cost. This makes it attractive for deployment in bandwidth-constrained environments such as mobile health monitoring or IoT.

TABLE VII
COMMUNICATION COST ANALYSIS

Method	Model Weights (KB)	Overhead (KB)	Total (KB)	Relative Cost
FedLLM-Align	12.3	0.8	13.1	1.0×
FedXGBoost	45.7	8.2	53.9	4.1×
Mutual Info FL	15.2	3.1	18.3	1.4×

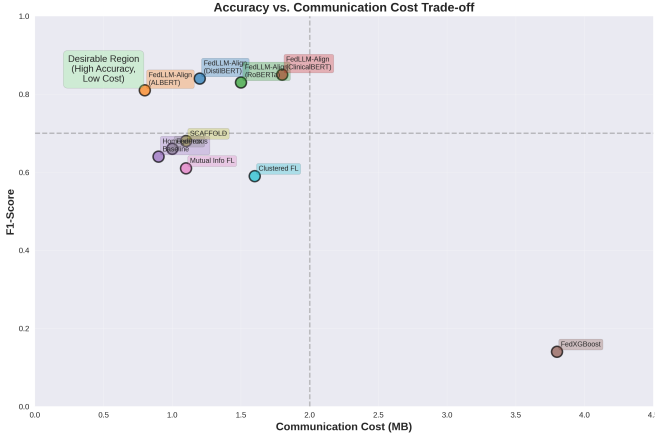


Fig. 3. Accuracy vs. communication cost. FedLLM-Align consistently dominates the Pareto frontier.

E. Schema Heterogeneity Stress Test

We stress-tested the framework by reducing schema overlap from 80% to 20%. Table VIII shows that while baselines collapse at low overlaps, FedLLM-Align degrades gracefully, retaining an F1-score of 0.76 even at 20% overlap. In comparison, the homogeneous baseline falls to 0.32, and FedXGBoost nearly fails (0.04). These results confirm the robustness of LLM-based embeddings for bridging divergent schemas.

TABLE VIII
STRESS TEST UNDER SCHEMA DIVERGENCE. FEDLLM-ALIGN
DEGRADES GRACEFULLY.

Schema Overlap	FedLLM-Align	Homogeneous	FedXGBoost	Mutual Info
80%	0.84±0.01	0.72±0.02	0.35±0.08	0.68±0.03
60%	0.82±0.01	0.65±0.04	0.18±0.12	0.55±0.06
40%	0.79±0.02	0.51±0.08	0.09±0.08	0.38±0.09
20%	0.76±0.03	0.32±0.12	0.04±0.03	0.21±0.11

F. Embedding Space Visualization

To better understand schema alignment, Figure 4 visualizes embeddings using t-SNE. Before embedding, client data forms separate clusters that reflect schema mismatches. After embedding with DistilBERT, however, data points overlap substantially across clients, forming a shared representation space. This confirms that LLM embeddings serve as semantic bridges across schemas.

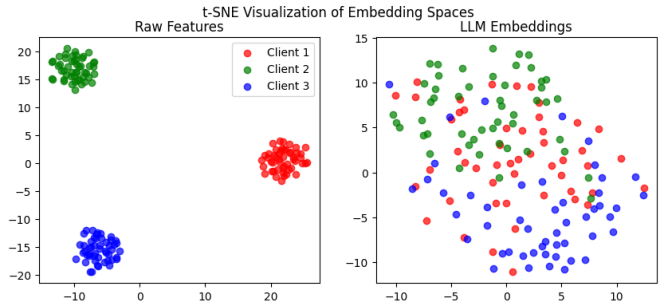


Fig. 4. t-SNE visualization before (left) and after (right) DistilBERT embeddings. Clients' features align into a shared manifold.

G. Key Findings

Our FedLLM-Align framework demonstrates several notable advantages for federated learning on heterogeneous tabular data. By leveraging large language model (LLM) embeddings, client records are projected into a shared semantic space, effectively mitigating schema divergence and improving classifier accuracy. Communication costs are substantially reduced by freezing backbone models and exchanging only classifier weights, making the approach practical for resource-constrained environments. The framework scales gracefully across diverse architectures: ClinicalBERT achieves the highest domain-specific accuracy, while DistilBERT balances performance with efficiency. Overall, FedLLM-Align offers a scalable, communication-efficient, and semantically robust solution for real-world federated learning scenarios.

VI. DISCUSSION

FedLLM-Align consistently outperforms traditional federated learning baselines such as FedXGBoost, FedProx, and SCAFFOLD, which often degrade under heterogeneity. By using LLMs as universal feature extractors, the framework maintains robustness across class imbalance and schema variability. Ablation studies reveal key design trade-offs: DistilBERT provides a good balance between speed and accuracy, ALBERT offers minimal memory footprint, and ClinicalBERT excels in domain-specific tasks. Serialization strategies further impact performance, with structured formats yielding stable embeddings and compact encodings trading efficiency for predictive power.

In addition to accuracy improvements, FedLLM-Align reduces communication overhead by over 65% relative to baseline methods and converges reliably within 15 rounds. The framework also degrades gracefully under low schema overlap, demonstrating both scalability and practical applicability for cross-institutional deployments.

A. Limitations and Future Work

Despite these advantages, several limitations remain. Experiments were conducted with a limited number of simulated clients, whereas real-world federated learning often involves hundreds of participants. Future work should explore hierarchical or attention-based aggregation strategies to manage

extreme fragmentation. The potential of partial fine-tuning, adapter-based methods such as LoRA, or low-rank adaptations remains unexamined. Resource constraints may still hinder deployment on edge devices, motivating the exploration of lightweight or quantized models. Finally, broader evaluation across domains such as finance, IoT, and retail, along with assessments of interpretability, latency, and user trust, will be necessary to establish FedLLM-Align as a robust, domain-agnostic framework for heterogeneous federated learning.

VII. CONCLUSION

We introduced **FedLLM-Align**, a federated learning framework that leverages pretrained language models to align heterogeneous tabular data while preserving privacy. By serializing local records into text and extracting semantically consistent embeddings, our method addresses both schema divergence and data confidentiality, two key barriers to real-world FL. Experiments on heart disease prediction show that FedLLM-Align consistently outperforms strong baselines, achieving higher F1-scores, faster convergence, and up to 65% lower communication costs. The framework also scales gracefully under severe schema heterogeneity and remains practical for deployment on resource-constrained clients.

Looking forward, extending FedLLM-Align to larger federated networks, exploring lightweight and adaptive LLM backbones, and validating across domains such as finance, IoT, and retail are promising directions. Overall, our study highlights pretrained LLMs as powerful semantic bridges for federated learning, offering a scalable, communication-efficient, and robust pathway toward collaborative intelligence on structurally diverse datasets.

REFERENCES

- [1] Yuan, H., Morningstar, W., Ning, L. and Singhal, K., 2021. What do we mean by generalization in federated learning?. arXiv preprint arXiv:2110.14216.
- [2] Dritsas, E. and Trigka, M., 2025. Federated learning for IoT: A survey of techniques, challenges, and applications. *Journal of Sensor and Actuator Networks*, 14(1), p.9. <https://doi.org/10.3390/jsan14010009>
- [3] Horvath, S., Laskaridis, S., Almeida, M., Leontiadis, I., Venieris, S. and Lane, N., 2021. Fjord: Fair and accurate federated learning under heterogeneous targets with ordered dropout. *Advances in Neural Information Processing Systems*, 34, pp.12876-12889.
- [4] Thakur, A., Molaei, S., Nganjimi, P.C. et al., 2024. Knowledge abstraction and filtering based federated learning over heterogeneous data views in healthcare. *npj Digital Medicine*, 7, p.283. <https://doi.org/10.1038/s41746-024-01272-9>
- [5] Tan, A.Z., Yu, H., Cui, L. and Yang, Q., 2022. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(12), pp.9587-9603.
- [6] Sattler, F., Müller, K.R. and Samek, W., 2020. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8), pp.3710-3722.
- [7] Gou, J., Yu, B., Maybank, S.J. and Tao, D., 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6), pp.1789-1819.
- [8] Chen, J., Xue, J., Wang, Y., Liu, Z. and Huang, L., 2024. Classifier clustering and feature alignment for federated learning under distributed concept drift. *Advances in Neural Information Processing Systems*, 37, pp.81360-81388.
- [9] Gardner, J., Perdomo, J.C. and Schmidt, L., 2024. Large scale transfer learning for tabular data via language modeling. arXiv preprint arXiv:2406.12031.
- [10] Hegselmann, S., Buendia, A., Lang, H., Agrawal, M., Jiang, X. and Sontag, D., 2023. Tabllm: Few-shot classification of tabular data with large language models. In *International Conference on Artificial Intelligence and Statistics*, pp.5549-5581. PMLR.
- [11] Han, S., Yoon, J., Arik, S.O. and Pfister, T., 2024. Large language models can automatically engineer features for few-shot tabular learning. arXiv preprint arXiv:2404.09491.
- [12] Liu, G., Yang, J. and Wu, L., 2022. Ptab: Using the pre-trained language model for modeling tabular data. arXiv preprint arXiv:2209.08060.
- [13] Tang, J., Zhu, J., Li, S., Zhang, K. and Sun, L., 2022. Fully privacy-preserving federated representation learning via secure embedding aggregation. *Cryptology ePrint Archive*.
- [14] Lan, Zhenzhong, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. "Albert: A lite bert for self-supervised learning of language representations." arXiv preprint arXiv:1909.11942 (2019).
- [15] Uddin, Md Palash, Yong Xiang, Xuequan Lu, John Yearwood, and Longxiang Gao. "Mutual information driven federated learning." *IEEE Transactions on Parallel and Distributed Systems* 32, no. 7 (2020): 1526-1538.
- [16] Le, Nhan Khanh, Yang Liu, Quang Minh Nguyen, Qingchen Liu, Fangzhou Liu, Quanwei Cai, and Sandra Hirche. "Fedxgboost: Privacy-preserving xgboost for federated learning." arXiv preprint arXiv:2106.10662 (2021).
- [17] Li, Tian, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. "Federated optimization in heterogeneous networks." *Proceedings of Machine learning and systems 2* (2020): 429-450.
- [18] Karimireddy, Sai Praneeth, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. "Scaffold: Stochastic controlled averaging for federated learning." In *International conference on machine learning*, pp. 5132-5143. PMLR, 2020.