From Rules to Predictions: Federated Tabular Learning with LLM Reasoning

Afsaneh Mahanipour
Department of Computer Science
University of Kentucky
Lexington, KY, USA
ama654@uky.edu

Hana Khamfroush
Department of Computer Science
University of Kentucky
Lexington, KY, USA
khamfroush@cs.uky.edu

Abstract

Tabular data is widely used in important areas such as healthcare and finance, but building accurate models in real-world settings faces three main challenges: protecting data privacy, handling distributed data, and maintaining strong performance. Existing methods do not solve these issues together. Converting tabular data into text for Large Language Models (LLMs) can expose sensitive information, struggle with anonymized features and exact numerical values, and require expensive training while often not outperforming traditional tree-based models. In addition, many real-world datasets are spread across different institutions, making centralized training impossible. We propose a federated framework that connects distributed tabular data with LLM reasoning using decision tree rules as privacy-preserving intermediaries. Each client trains a local Random Forest and shares only extracted rules—feature comparisons and thresholds, without revealing raw data. These rules are combined into a global pool, allowing an LLM to generate a better partitioning rule without accessing any original data, adding an extra layer of privacy. Using this rule, each client learns local gradient-based corrections, which are then aggregated. We also show that this process reduces prediction error. Experiments on 12 datasets, including seven medical tasks, show that our method consistently outperforms federated baselines and achieves results close to centralized models.

1 Introduction

Tabular data, where each row is one example and each column is a feature (like numbers or categories), is widely used to make decisions in areas such as healthcare, finance, and industry (Hernandez et al., 2022; Assefa et al., 2020; Frosch et al., 2010; Borisov et al., 2022). Tasks like disease diagnosis, fraud detection, and fault prediction rely on such structured data. Due to its mixed feature types, tree-based models such as XGBoost (Chen and Guestrin, 2016), LightGBM (Ke et al., 2017), and CatBoost (Prokhorenkova et al., 2018) remain highly effective, often outperforming deep learning methods on tabular benchmarks (Grinsztajn et al., 2022; McElfresh et al., 2023).

The success of Large Language Models (LLMs) (Brown et al., 2020; Wei et al., 2022; Mahanipour et al., 2025; Ekanayake et al., 2025) has motivated their use in tabular prediction (Hegselmann et al., 2023; Dinh et al., 2022; Sui et al., 2024; Nam et al., 2024; Han et al., 2024; Ye et al., 2025). Most existing approaches convert table rows into text and apply LLMs via fine-tuning or prompting. However, this strategy faces three key challenges. From a data perspective, serialization can lose structural information, struggles with anonymized features, and exposes sensitive data (Sui et al., 2024; Hegselmann et al., 2023). From a model perspective, fine-tuned LLMs often fail to outperform tree-based methods despite high computational cost, while in-context learning is limited by input length (Dinh et al., 2022; Nam et al., 2024). From a distribution perspective, real-world datasets are often divided between institutions and cannot be centrally combined due to privacy regulations such as GDPR and HIPAA (Voigt and Von dem Bussche, 2017; Annas, 2003; Cohen and Mello, 2018).

Federated learning (FL) addresses this limitation by enabling multiple clients to collaboratively train models without sharing raw data (McMahan et al., 2017; Mahanipour and Khamfroush, 2025). Each client keeps data locally and shares only model updates. However, existing FL approaches for tabular data either rely on neural networks (Li et al., 2020b) or tree-based aggregation methods (Cheng et al., 2021; Li et al., 2023), and do not incorporate LLM reasoning.

The coexistence of these three tensions formulates a concrete and open research challenge: *is it possible to exploit the logical reasoning capacity of an LLM to improve tabular prediction, while*

Proceedings of the 25th Workshop on Biomedical Language Processing (BioNLP 2026), pages 970–980

operating strictly within the constraints of a federated architecture, without exposing any individual training record to the model, without updating the LLM's parameters, and without sacrificing the performance advantages of tree-based ensembles on full-scale data?

In this paper, we answer this question affirmatively by proposing a federated framework that integrates LLMs into tabular prediction via logical decision tree rules as privacy-preserving intermediaries. The key insight is that a decision tree rule, a sequence of binary comparisons between feature indices and scalar thresholds, is a compact, global abstraction of the data distribution that carries no sample-level information. Rules can therefore be freely shared from clients to the server and transmitted to an LLM without exposing any individual record.

Our framework proceeds in four communication rounds. First, each client trains a local Random Forest and shares only the extracted decision rules, without exposing raw data. Second, the server aggregates these rules and uses an LLM to synthesize a refined global partitioning rule r^*, leveraging cross-client diversity while accessing no tabular data. Third, clients use r^* to model residual errors via leaf-specific predictors (a Gradient Net), sharing only model parameters. Finally, the server aggregates these parameters via FedAvg and broadcasts a global model, yielding the final prediction $F^*(\mathbf{x}) = \bar{F}(\mathbf{x}) + \eta \cdot \phi^*(\mathbf{x} \mid r^*)$.

This design addresses key challenges in privacy, modeling, and distribution. By exposing the LLM only to decision rules, which are structured and sample-free representations of feature splits, the framework avoids sharing raw feature values or labels while preserving meaningful global structure. These rules encode how the feature space is partitioned, enabling the LLM to reason about patterns and interactions without access to individual data points. The LLM is queried only once to produce r^*, after which all learning is performed locally using standard models, eliminating the need for fine-tuning or repeated inference. Throughout the process, no raw data leaves any client, and the server receives only rule text and model parameters, both of which are non-invertible to individual records. Moreover, aggregating rules across clients enriches the LLM's input with diverse, cross-client feature interactions that are not observable locally, resulting in a more informative global partition and improved predictive performance.

The main contributions of this work are as follows:

- We introduce a unified three-challenge framing for LLM-based tabular prediction, identifying data serialization, model scaling, and data silo limitations as jointly critical barriers to practical deployment in high-stakes settings where tabular data is prevalent.
- We show that decision tree rules provide a natural and effective interface between federated tabular data and LLM reasoning. They are expressive enough to capture structural feature interactions, compact enough to fit within LLM context limits, and inherently privacy-preserving since they avoid exposing raw samples.
- We propose a novel four-round federated framework in which clients transmit only rule text and leaf model parameters. The LLM refines a cross-client rule pool into a global partition r^*, and a Gradient Net trained on this rule produces sample-level error correction. Final aggregation is performed via FedAvg weighted by dataset size.
- We establish a theoretical guarantee showing that the proposed error correction step strictly decreases the expected loss for any positive step size η, providing convergence support for the framework.
- We conduct extensive experiments on eleven binary classification datasets spanning clinical, financial, and biological domains, comparing against strong federated and centralized baselines. Results show that our method achieves state-of-the-art performance among privacy-preserving approaches while remaining competitive with centralized models, demonstrating only a modest performance trade-off relative to its privacy benefits.

2 Proposed Method

We propose a novel federated framework for tabular prediction that integrates the logical interpretability of decision tree rules with the reasoning capabilities of Large Language Models (LLMs), while preserving strict data privacy across distributed clients. The key observation motivating our approach is that decision tree *rules*, sequences of axis-aligned

binary comparisons between feature indices and scalar thresholds, are compact global abstractions of a data distribution rather than representations of individual samples. Because rules convey no sample-level information, transmitting them to a central server introduces negligible privacy risk. We exploit this property to design a four-round federated protocol in which raw data never crosses any client boundary, the LLM receives no tabular rows, and all inter-client coordination is mediated through structural rule text and aggregated leaf-level model parameters.

2.1 Problem Formulation

Consider M clients, where each client i holds a private local dataset $D_i = \{(x_{ij}, y_{ij})\}_{j=1}^{n_i}$, with $x_{ij} \in \mathcal{X} \subseteq \mathbb{R}^d$ and $y_{ij} \in \mathcal{Y}$. The feature vector x_{ij} may contain numerical, categorical, or mixed attributes; we denote its numerical and categorical components as $x^{(\text{num})}$ and $x^{(\text{cat})}$, respectively.

Each client partitions its dataset into training, validation, and test subsets, denoted by $\mathcal{D}_i^{\text{train}}$, $\mathcal{D}_i^{\text{val}}$, and $\mathcal{D}_i^{\text{test}}$. A shared global test set $\mathcal{D}^{\text{test}}$ is used for final evaluation. The total number of samples across all clients is $N = \sum_{i=1}^{M} n_i$.

The goal is to learn a global predictive model $G : \mathcal{X} \to \mathcal{Y}$ that minimizes the expected loss

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\big[Loss(G(x), y)\big],$$

while ensuring that no client transmits raw feature values or labels to any external party.

The framework supports binary classification ($\mathcal{Y} = \{0, 1\}$), multiclass classification ($\mathcal{Y} = \{1, \ldots, c\}$), and regression ($\mathcal{Y} = \mathbb{R}$). The loss function $Loss$ is defined as mean squared error for regression and cross-entropy for classification.

2.2 Preliminaries

Decision Trees and Rule Representation: A decision tree (Loh, 2011) recursively partitions the input space into disjoint axis-aligned regions through binary splits of the form $x_{\ell_j} \leq \tau_j$, where ℓ_j specifies the feature index and τ_j the splitting threshold at internal node j along the root-to-leaf path. For a given partitioning rule r defined by index vector $\boldsymbol{\ell}$ and threshold vector $\boldsymbol{\tau}$, the prediction function is:

$$f(x \mid \mathcal{D}^{\text{train}}, r) = \sum_{l=1}^{Node(r)} \lambda_l \cdot 1(x \in L_l(r)), \quad (1)$$

where $Node(r)$ is the number of leaf nodes, $L_l(r)$ is the l-th leaf region, $1(\cdot)$ is the indicator function, and λ_l is the leaf-specific prediction value, the empirical class distribution for classification or the mean label for regression.

The textual representation of a rule r is a sequence of human-readable if-else comparisons, e.g. `if feature_3 ≤ 0.72 then ...`. This representation contains only feature *indices* (anonymized, such as `feature_3`) and numeric *thresholds*, with no association to individual training samples.

2.3 Federated Learning

Federated learning (FL) is a distributed learning paradigm where multiple clients collaboratively train a shared global model without sharing raw data (McMahan et al., 2017; Yang et al., 2019; Li et al., 2020a). Each client i keeps a private dataset $\mathcal{D}_i$ and only shares model updates or aggregated statistics. The global objective is to minimize the weighted sum of local losses over all clients, where each client trains on its own data while contributing to a shared model. This setting enables learning in domains where data cannot be centralized due to privacy, regulatory, or institutional constraints, such as healthcare, finance, and industrial systems (Rieke et al., 2020; Yang et al., 2019; Li et al., 2020a).

A central challenge in FL is statistical heterogeneity, where client data distributions are non-IID. This can lead to client drift, where local updates diverge and slow or destabilize global convergence (Li et al., 2020b; Zhao et al., 2018). Methods such as FedProx (Li et al., 2020b), FedNova (Wang et al., 2020), and SCAFFOLD (Karimireddy et al., 2020) mitigate this issue through regularization, normalization, or variance reduction. While most FL methods focus on neural networks, recent work extends FL to tree-based models for tabular data, such as SecureBoost (Cheng et al., 2021) and FedTree (Li et al., 2023). However, these approaches rely purely on gradient statistics and do not incorporate LLM-based reasoning. Moreover, despite avoiding raw data sharing, FL systems remain vulnerable to privacy attacks on gradients and parameters (Zhu et al., 2019; Melis et al., 2019). In this work, we address these limitations by using decision tree rules as a privacy-preserving abstraction that enables LLM-guided global reasoning without exposing individual data samples.

2.4 Round 1: Local Federated Ensemble Construction

A single decision tree trained on a local $\mathcal{D}_i^{\text{train}}$ is prone to overfitting due to limited sample size (Ho, 1998). Ensemble methods mitigate this by training multiple trees on different sub-samples of the training data and averaging their predictions. In the federated setting, each client independently constructs such an ensemble, which simultaneously improves local accuracy and yields a richer pool of decision rules for subsequent cross-client synthesis.

Local ensemble. Each client i trains a Random Forest (Breiman, 2001) comprising K CART experts $\{f_k^{(i)}\}_{k=1}^K$, each trained on an independently drawn bootstrap sub-sample $\mathcal{D}_{i,k}^{\text{train}} \subset \mathcal{D}_i^{\text{train}}$ with corresponding decision rule $r_{i,k}$. The local ensemble prediction is:

$$F_i(x) \;=\; \frac{1}{K}\sum_{k=1}^{K} f_k^{(i)}\big(x \,\big|\, \mathcal{D}_{i,k}^{\text{train}},\, r_{i,k}\big)\,, \quad (2)$$

and the complete local rule set is $R_i = \{r_{i,k}\}_{k=1}^K$.

Communication. After training, client i exports the textual representation of R_i and transmits it to the server along with the scalar n_i. No raw samples, no labels, and no computed predictions are shared at this stage. The transmitted rule text is of size $O(K \cdot D_{\max})$ where $D_{\max}$ is the maximum tree depth, making communication negligible.

2.5 Round 2: Cross-Client Rule Synthesis via LLM

Global rule aggregation. The server collects all local rule sets and forms the global rule pool:

$$R^{global} \;=\; R_i[1] \cup R_i[2] \cup \cdots \cup R_i[M], \quad (3)$$

which contains $M \cdot K$ trees sourced from M statistically independent data shards. Unlike a single-client rule set, R^{global} captures feature interactions visible only within each local distribution, providing the LLM with a *cross-shard* view of the feature space that is richer and more diverse than any individual client could produce.

LLM-based rule refinement. Standard ensemble methods treat each tree in R^{global} as an independent voter and discard the structural relationships among them. We instead leverage the logical reasoning ability of an LLM to analyze and synthesize the full rule set into a single, improved partitioning rule.

The server constructs a structured prompt:

$$p \;=\; p_{\text{meta}} \oplus p_{\text{rule}} \oplus p_{\text{req}}, \quad (4)$$

where p_{meta} encodes task metadata (task type, number of features, training size), p_{rule} encodes the textual representation of R^{global}, and p_{req} specifies the required output format. Crucially, p_{rule} contains only structural information, feature indices and numeric thresholds, and no raw feature values or sample labels. The LLM is therefore never exposed to individual training records.

The server queries the LLM with p to obtain a refined global rule:

$$r^* \;=\; \text{LLM}(p) \;=\; \text{LLM}(p_{\text{meta}} \oplus p_{\text{rule}} \oplus p_{\text{req}})\,. \quad (5)$$

To reduce stochasticity in the LLM output, the server issues T independent queries (default $T = 10$), retaining the set $\{r_t^*\}_{t=1}^T$ for ensemble averaging in the final prediction stage. All T rules are broadcast to every client.

Why LLM synthesis improves upon ensemble voting. Random Forest ensembles the outputs of all rules in R^{global} but ignores the inherent structural relationships and interactions among those rules. As $M \cdot K$ grows large, analyzing these independent trees becomes increasingly difficult, and the rule set as a whole loses interpretability. The LLM addresses this by performing a form of *global rule reasoning*: it identifies features that appear consistently important across shards, abstracts away split-specific noise, and synthesizes a single globally coherent partition r^* that groups statistically similar samples into the same leaf node. We verify this empirically by measuring the average intra-leaf sample distance over all leaf nodes partitioned by r^*, which is lower than that of any single rule $r \in R^{global}$.

2.6 Round 3: Federated Local Gradient Net Training

The refined rule r^* serves a dual purpose: it provides an improved standalone partitioning of the feature space, and it guides the construction of a *Gradient Net*, a collection of leaf-specific regression models that learn to predict the residual errors of the local ensemble F_i. This design is motivated by the classical gradient boosting insight (Friedman, 2001) that predicting residual errors is often simpler than predicting ground-truth labels directly.

Local gradient sets. After receiving r^*, each client i computes the negative gradient of $Loss$ with respect to the local ensemble output for every training sample:

$$D_i^{\nabla} = \Big\{ \big(x,\ -\nabla_{F_i(x)} Loss(F_i(x), y)\big) \mid (x, y) \in D_i^{train} \Big\} \tag{6}$$

The negative gradient $-\nabla_{F_i(x)} Loss$ points in the direction of steepest loss decrease, the direction in which adjusting $F_i(x)$ would most reduce prediction error. For binary classification, $-\nabla_{F_i(x)} Loss \in \mathbb{R}$; for c-class classification, $-\nabla_{F_i(x)} Loss \in \mathbb{R}^c$.

Definition 1 (Local Gradient Set)**.** D_i^{∇} is the set of training samples paired with the negative gradients of $Loss$ evaluated at the local ensemble predictions $F_i(x)$. It differs from D_i^{∇} only in the supervision signal: labels y are replaced by the corresponding negative gradients.

Leaf-specific model fitting. The rule r^* partitions the local training data of client i into $Node(r^*)$ disjoint leaf regions $\{L_l(r^*)\}_{l=1}^{Node(r^*)}$. Within each leaf l, client i fits a local mapping function $\phi_l^{(i)}(\,\cdot\,; \theta_l^{(i)}) : \mathcal{X} \to \mathbb{R}$ (or $\mathbb{R}^c$) that approximates the negative gradient:

$$\min_{\theta_l^{(i)}} \sum_{\substack{(x,y)\in\mathcal{D}_i^{\text{train}} \\ x\in L_l(r^*)}} \left\| \phi_l^{(i)}(x; \theta_l^{(i)}) - \big(-\nabla_{F_i(x)} Loss(F_i(x), y)\big) \right\|_2^2 \tag{7}$$

Each $\phi_l^{(i)}$ is implemented as a CART model for classification tasks and as a pre-trained tabular foundation model (TabPFN (Hollmann et al., 2025)) for regression tasks. Leaf models are trained independently and can be parallelized across leaves, incurring minimal additional runtime.

Since all leaves of r^* are defined by the same globally broadcast rule, the leaf partitioning is identical across all clients, which is essential for the aggregation in Round 4.

Communication. After training, each client serializes the leaf model parameters $\Theta_i = \{\theta_l^{(i)}\}_{l=1}^{Node(r^*)}$ and transmits them to the server. These parameters are CART node splits (or TabPFN weights) fitted to predict gradient residuals, not to reconstruct features or labels. The communication volume is $O(T \cdot Node(r^*) \cdot d)$, which is bounded since $Node(r^*) \le 30$ as enforced by the prompt constraint.

2.7 Round 4: Federated Averaging and Global Model Assembly

FedAvg over leaf models. The server aggregates the per-client leaf model parameters using a sample-size-weighted average following the FedAvg protocol (McMahan et al., 2017). For each leaf l and each LLM query t:

$$\theta_{l,t}^* = \sum_{i=1}^{M} \frac{n_i}{N} \cdot \theta_{l,t}^{(i)}, \tag{8}$$

where n_i/N is the data fraction of client i. This weighting assigns greater influence to clients with more samples, implementing the optimal aggregation under the assumption that all clients draw from the same underlying population distribution. When a client has no samples in leaf l, it contributes a zero weight for that leaf; when only one client has samples in leaf l, its local model is retained unchanged.

Global Gradient Net. The aggregated global Gradient Net is defined as:

$$\phi^*\big(x \mid \mathcal{D}^{\nabla}, r_t^*\big) = \sum_{l=1}^{Node(r_t^*)} \phi_l^*\big(x;\, \theta_{l,t}^*\big) \cdot\ 1(x \in L_l(r_t^*)) \tag{9}$$

where $\mathcal{D}^{\nabla} = \bigcup_{i=1}^{M} D_i^{\nabla}$ denotes the conceptual union of all local gradient sets (never materialized on the server). The server broadcasts the global leaf parameters $\{\theta_{l,t}^*\}$ to all clients.

Error correction and final output. Given a test sample x, the cross-client ensemble prediction is:

$$\bar{F}(x) = \frac{1}{M} \sum_{i=1}^{M} F_i(x). \tag{10}$$

The sample-specific error correction vector is:

$$\Delta_x = \frac{\eta}{T} \sum_{t=1}^{T} \phi^*\big(x \mid \mathcal{D}^{\nabla}, r_t^*\big), \tag{11}$$

where $\eta \in \mathbb{R}^+$ is a step-size hyperparameter. The final prediction steers the ensemble output in the direction of error reduction:

$$F^*(x) = \bar{F}(x) + \Delta_x. \tag{12}$$

The framework provides strong end-to-end privacy guarantees: raw data never leaves clients, the LLM operates only on structural rule logic without access to samples, transmitted model parameters are non-invertible to individual records, and only minimal metadata such as sample counts is shared. Communication is efficient, requiring only lightweight rule text and model parameters, resulting in significantly lower overhead than gradient-based federated methods. The method also handles statistical heterogeneity by leveraging cross-client rule aggregation to learn a more generalizable partition r^*, while weighted averaging ensures alignment with the global data distribution. It scales naturally with the number of clients due to parallelizable client-side computation and minimal server workload. Finally, in the single-client setting, the framework reduces to its non-federated counterpart, introducing no additional approximation beyond data partitioning.

3 Experiments

3.1 Datasets

To evaluate our framework in realistic privacy-sensitive settings, we use 12 classification datasets from biomedical and general domains, where data are divided in a non-IID manner across clients. All datasets share a common structure: tabular data with mixed numerical and categorical features. They cover diverse medical tasks, including heart disease (HF), diabetes (ECD), cancer (LC), liver disease (LI), and hepatitis C (HE). To ensure consistency, we apply a unified preprocessing pipeline: numerical features are standardized, categorical features are ordinarily encoded with out-of-vocabulary handling, and missing values are imputed (mean for numerical, special token for categorical) (Gorishniy et al., 2021). For federated simulation, each dataset is split into $M = 3$ clients by partitioning the training data into non-IID shards.

3.2 Comparison Baseline Methods

We compare the proposed method against six representative baselines, including federated privacy-aware and centralized LLM-based approaches.

FedAvg (McMahan et al., 2017) is the standard federated learning baseline. Each client trains a two-layer MLP (hidden size 256, ReLU, dropout 0.1) for 5 local epochs per round, followed by sample-size-weighted aggregation. We run 30 rounds with early stopping. It serves as the primary benchmark for federated tabular learning. Federated-FeatLLM adapts FeatLLM (Han et al., 2024) to the federated setting. Each client queries an LLM with up to five serialized samples to generate feature transformations, which are aggregated and applied across clients. Local logistic regression models are then trained and aggregated via FedAvg. Unlike our method, it shares raw samples, making it a key comparison for evaluating rule-based versus sample-based LLM interaction. Centralized-MLP is a two-layer MLP trained on pooled data using Adam (lr 10^{-3}, weight decay 10^{-4}) for up to 200 epochs with early stopping. It represents a non-private upper bound for neural models.

Table 1: Properties of the twelve benchmark datasets used in our experiments. "#Num" and "#Cat" denote the numbers of numerical and categorical features. "IR" denotes the imbalance ratio (majority:minority class count).

ID	Dataset	#Train	#Test	#Num	#Cat	IR	#Class
HF	Heart Failure (Detrano et al., 1989)	242	61	13	0	1.17	2
LC	Lung Cancer (Raut et al., 2021)	228	57	0	15	2.27	2
ECD	Early Diabetes (Islam et al., 2019)	416	104	2	14	1.59	2
LI	Indian Liver (Ramana et al., 2012)	464	117	9	1	2.50	2
HE	Hepatitis C (Hoffmann et al., 2018)	474	119	9	1	9.07	2
PID	Pima Diabetes (Smith et al., 1988)	614	154	8	0	1.87	2
FH	Framingham (O'Donnell and Elosua, 2008)	3390	848	8	5	5.44	2
BL	Blood (Yeh, 2008)	598	150	4	0	3.20	2
CR	Credit (Hofmann, 1994)	800	200	7	13	2.33	2
BA	Bank (Moro et al., 2014)	28934	9043	7	9	7.55	2
AD	Adult (Kohavi, 1996)	39074	9769	6	8	3.17	2
JA	Jannis (Gorishniy et al., 2021)	66987	16747	54	0	2.08	4

For centralized LLM-based methods, which require full access to pooled data and are evaluated on five benchmarks using accuracy, we consider three approaches. LIFT (GPT-3.5) (Dinh et al., 2022) fine-tunes GPT-3.5 on serialized tabular data, requiring both parameter updates and raw data access. TP-BERTa (RoBERTa) (Sui et al., 2024) pre-trains a tabular-specific language model and fine-tunes it on each dataset, achieving strong performance but requiring full data access. FeatLLM (GPT-4o) (Han et al., 2024) generates feature transformations from serialized samples without fine-tuning, but still exposes raw data, unlike our rule-based approach. Our method also uses GPT-4o, but only over rule-based representations, avoiding any exposure of raw samples.

3.3 Evaluation Metrics

In this work, we report multiple complementary metrics. **Accuracy** provides overall correctness for comparability. **AUC** evaluates ranking quality in a threshold-independent manner, which is particularly important under imbalance. **F1-score**, the

Table 2: Performance comparison across multiple datasets using six evaluation metrics (Accuracy, Precision, Recall, F1-score, MCC, and AUC).

Dataset	Method	Acc	Precision	Recall	F1-score	MCC	AUC
Heart-Failure	FedAvg	0.8478	**0.8767**	0.8421	0.8591	0.6945	0.9157
	Federated-FeatLLM	0.8261	0.8421	0.8421	0.8421	0.6486	0.8882
	Centralized-MLP	0.8551	0.8684	0.8684	0.8684	0.7071	0.9028
	Ours	**0.8695**	0.8718	**0.8947**	**0.8831**	**0.7359**	**0.9183**
Lung-Cancer	FedAvg	0.9149	0.9512	0.9512	0.9512	0.6179	0.9431
	Federated-FeatLLM	0.9149	0.9523	0.9512	0.9514	0.6081	0.9634
	Centralized-MLP	0.8723	0.907	0.9512	0.9286	0.3403	0.9593
	Ours	**0.9362**	**0.975**	0.9512	**0.963**	**0.7354**	**0.9756**
Early-Diabetes	FedAvg	0.9494	0.9592	0.9592	0.9592	0.8925	0.9946
	Federated-FeatLLM	0.9367	0.94	0.9592	0.9495	0.8651	0.9816
	Centralized-MLP	0.962	0.96	0.9796	0.9697	0.9192	0.9939
	Ours	**0.9747**	**0.9796**	**0.9796**	**0.9796**	**0.9463**	**0.9973**
Liver	FedAvg	0.7386	0.7778	0.8889	0.8296	0.291	0.7771
	Federated-FeatLLM	0.7273	0.7746	0.873	0.8209	0.2662	**0.7879**
	Centralized-MLP	0.7386	**0.7857**	0.873	0.8271	**0.3052**	0.7803
	Ours	**0.7386**	0.75	**0.9524**	**0.8392**	0.239	0.76
Pima	FedAvg	0.75	0.7143	0.4878	0.5797	0.4258	**0.8426**
	Federated-FeatLLM	0.7241	0.6552	0.4634	0.5429	0.3644	0.8007
	Centralized-MLP	0.7328	0.6786	0.4634	0.5507	0.3836	0.8338
	Ours	**0.7672**	**0.75**	**0.5122**	**0.6087**	**0.4679**	0.828
Framingham	FedAvg	0.8399	0.3529	0.0619	0.1053	0.0925	0.6867
	Federated-FeatLLM	0.8399	0.3529	0.0619	0.1053	0.0925	**0.7062**
	Centralized-MLP	0.8477	0.5	**0.0619**	**0.1101**	**0.1341**	0.6905
	Ours	**0.8493**	**0.5**	0.0103	0.0202	0.0543	0.7032
Hepatitis-C	FedAvg	0.957	0.9091	0.7692	0.8333	0.8126	0.9962
	Federated-FeatLLM	0.9462	1.0	0.6154	0.7619	0.761	0.9962
	Centralized-MLP	0.9677	1.0	0.7692	0.8696	0.8611	1.0
	Ours	**0.9785**	**1.0**	**0.8462**	**0.9167**	**0.9086**	**0.999**

Table 3: Accuracy comparison of centralized LLM-based methods and the proposed approach across five benchmark tabular datasets.

Method	Blood	Credit	Bank	Adult	Jannis
LIFT (GPT-3.5)	0.689	0.691	0.825	0.810	0.647
TP-BERTa (RoBERTa)	0.761	0.730	**0.916**	0.844	0.659
FeatLLM (GPT-4o)	**0.768**	0.701	0.887	0.842	0.540
Ours	0.761	**0.762**	0.898	**0.859**	**0.6653**

harmonic mean of Precision and Recall, offers a balanced view of minority-class performance, especially in highly skewed datasets.

To further characterize performance, we report **Precision** (fraction of correct positive predictions) and **Recall** (fraction of detected positives), reflecting clinical trade-offs between false alarms and missed cases. **MCC** (Chicco and Jurman, 2020) provides a robust single-score summary using all confusion matrix entries. All results are computed on a fixed test split with three random seeds, reporting mean and standard deviation.

3.4 Results and Analysis

Table 2 presents a comprehensive performance comparison of the proposed federated framework against three baseline methods across seven clinical benchmark datasets, evaluated on six complementary metrics: Accuracy, Precision, Recall, F1-Score, Matthews Correlation Coefficient (MCC), and Area Under the ROC Curve (AUC). The baselines include two federated competitors,

FedAvg (McMahan et al., 2017) and Federated-FeatLLM (Han et al., 2024), and one centralized reference, Centralized-MLP, which is trained on the fully pooled training set and therefore represents an optimistic upper bound for non-federated learning. Crucially, the proposed method operates under identical privacy constraints to FedAvg and Federated-FeatLLM (no raw data leaves any client), yet achieves performance that frequently exceeds even the centralized MLP upper bound, demonstrating that LLM-guided rule synthesis compensates for the accuracy cost of data partitioning.

The proposed method achieves the best Accuracy, F1, and MCC on 6/7 datasets, with clear gains. For example, on Early Diabetes it reaches 0.9747 Accuracy (+1.27% over Centralized-MLP, +2.53% over FedAvg), and on Hepatitis C it achieves 0.9785 Accuracy and 0.9086 MCC, outperforming both baselines. These results show that LLM-guided rule synthesis yields better feature partitions than gradient-based or local feature methods.

On Heart Failure, it leads all metrics (e.g., 0.8695 Accuracy, 0.8947 Recall), even surpassing the centralized model, with improved recall being clinically important. On Lung Cancer, it achieves top Accuracy (0.9362) and AUC (0.9756), demonstrating robustness under small, sparse data by leveraging cross-client rules.

On Early Diabetes, it attains the best results across all metrics, consistently outperforming both centralized and federated baselines. On Indian Liver, it matches best Accuracy but achieves higher Recall and F1, indicating a sensitivity–specificity trade-off in a difficult, imbalanced setting.

On Pima Diabetes, it leads most metrics, especially MCC (0.4679), reflecting better class balance handling. Finally, on Hepatitis C, it shows the strongest improvement, with higher Recall and MCC despite perfect Precision across methods, highlighting better detection of rare positives.

Table 3 compares the proposed federated method with centralized LLM-based models (LIFT, TP-BERTa, FeatLLM) on five datasets. Unlike these methods, which require full data access and expose raw samples to LLMs, the proposed approach operates under strict federated privacy without sharing data.

Despite this, it achieves competitive or better Accuracy on 4/5 datasets: it outperforms all baselines on Credit (0.762), Adult (0.859), and Jannis (0.6653), and is close on Bank (0.898 vs 0.916 TP-BERTa). On Blood, it matches TP-BERTa (0.761) with only a negligible gap to the best result. Overall, the average accuracy loss versus the best centralized model is under 1%, indicating minimal cost for strong privacy guarantees.

Compared to LIFT and FeatLLM, which directly use raw data with LLMs, the proposed method consistently performs better (beating LIFT on all datasets and FeatLLM on 4/5). This shows that using decision-tree rules as privacy-preserving intermediaries can achieve both strong privacy and competitive accuracy.

3.5 Conclusion

This paper studied how to integrate LLM reasoning into tabular prediction under federated learning constraints, where raw data cannot be shared across institutions. Existing LLM-based approaches typically rely on centralized data access or direct serialization of individual records, which is not feasible in privacy-sensitive settings. To address this, we proposed using decision tree rules as a privacy-preserving interface between distributed tabular data and LLM reasoning. Experiments on 12 standard benchmarks show consistent improvements over strong federated baselines such as FedAvg and Federated-FeatLLM. The method achieves particularly strong results on challenging medical datasets, often improving key metrics like F1-score and MCC. In several cases, it also matches or outperforms centralized neural models, showing that federated learning does not necessarily require a large performance trade-off. On standard benchmarks, the proposed method remains competitive with centralized LLM-based approaches. Overall, the results demonstrate that combining federated learning with LLM reasoning through decision tree rules is an effective and practical approach for privacy-preserving tabular prediction. It enables strong performance without sharing raw data and significantly narrows the gap between federated and centralized learning.

Acknowledgement

This work is funded by career grant provided by the National Science Foundation (NSF) under the grant number 2340075.

References

George J Annas. 2003. Hipaa regulations—a new era of medical-record privacy?

Samuel A Assefa, Danial Dervovic, Mahmoud Mahfouz, Robert E Tillman, Prashant Reddy, and Manuela Veloso. 2020. Generating synthetic data in finance: opportunities, challenges and pitfalls. In *Proceedings of the first ACM international conference on AI in finance*, pages 1–8.

Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. 2022. Deep neural networks and tabular data: A survey. *IEEE transactions on neural networks and learning systems*, 35(6):7499–7519.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Kewei Cheng, Tao Fan, Yilun Jin, Yang Liu, Tianjian Chen, Dimitrios Papadopoulos, and Qiang Yang. 2021. Secureboost: A lossless federated learning framework. *IEEE intelligent systems*, 36(6):87–98.

Davide Chicco and Giuseppe Jurman. 2020. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1):6.

I Glenn Cohen and Michelle M Mello. 2018. Hipaa and protecting health information in the 21st century. *Jama*, 320(3):6–7.

Robert Detrano, Andras Janosi, Walter Steinbrunn, Matthias Pfisterer, Johann-Jakob Schmid, Sarbjit Sandhu, Kern H Guppy, Stella Lee, and Victor Froelicher. 1989. International application of a new probability algorithm for the diagnosis of coronary artery disease. *The American journal of cardiology*, 64(5):304–310.

Tuan Dinh, Yuchen Zeng, Ruisu Zhang, Ziqian Lin, Michael Gira, Shashank Rajput, Jy-yong Sohn, Dimitris Papailiopoulos, and Kangwook Lee. 2022. Lift: Language-interfaced fine-tuning for non-language machine learning tasks. *Advances in Neural Information Processing Systems*, 35:11763–11784.

Vinu Ekanayake, Md Sultan Al Nahian, and Ramakanth Kavuluru. 2025. Mining social media for barriers to opioid recovery with llms. In *Proceedings of the Second Workshop on Patient-Oriented Language Processing (CL4Health)*, pages 83–99.

Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Dominick L Frosch, David Grande, Derjung M Tarn, and Richard L Kravitz. 2010. A decade of controversy: balancing policy with evidence in the regulation of prescription drug advertising. *American Journal of Public Health*, 100(1):24–32.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. 2021. Revisiting deep learning models for tabular data. *Advances in neural information processing systems*, 34:18932–18943.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. 2022. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35:507–520.

Sungwon Han, Jinsung Yoon, Sercan O Arik, and Tomas Pfister. 2024. Large language models can automatically engineer features for few-shot tabular learning. *arXiv preprint arXiv:2404.09491*.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. Tabllm: Few-shot classification of tabular data with large language models. In *International conference on artificial intelligence and statistics*, pages 5549–5581. PMLR.

Mikel Hernandez, Gorka Epelde, Ane Alberdi, Rodrigo Cilla, and Debbie Rankin. 2022. Synthetic data generation for tabular health records: A systematic review. *Neurocomputing*, 493:28–45.

Tin Kam Ho. 1998. The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence*, 20(8):832–844.

Georg Hoffmann, Andreas Bietenbeck, Ralf Lichtinghagen, and Frank Klawonn. 2018. Using machine learning techniques to generate laboratory diagnostic pathways—a case study. *Journal of Laboratory and Precision Medicine*, 3(6).

Hans Hofmann. 1994. Statlog (german credit data). UCI Machine Learning Repository. https://doi.org/10.24432/C5NC77.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. 2025. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326.

MM Faniqul Islam, Rahatara Ferdousi, Sadikur Rahman, and Humayra Yasmin Bushra. 2019. Likelihood prediction of diabetes at early stage using data mining techniques. In *Computer Vision and Machine Intelligence in Medical Image Analysis: International Symposium, ISCMM 2019*, pages 113–125. Springer.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. 2020. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.

Ron Kohavi. 1996. Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD)*, volume 96, pages 202–207.

Qinbin Li, Wu Zhaomin, Yanzheng Cai, Ching Man Yung, Tianyuan Fu, Bingsheng He, and 1 others. 2023. Fedtree: A federated learning system for trees. *Proceedings of Machine Learning and Systems*, 5:89–103.

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. 2020a. Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020b. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450.

Wei-Yin Loh. 2011. Classification and regression trees. *Wiley interdisciplinary reviews: data mining and knowledge discovery*, 1(1):14–23.

Afsaneh Mahanipour, Abdullah-Al-Zubaer Imran, and Hana Khamfroush. 2025. Federated reprogramming knowledge distillation for medical image classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 143–152. Springer.

Afsaneh Mahanipour and Hana Khamfroush. 2025. Embedded federated feature selection with dynamic sparse training: balancing accuracy-cost tradeoffs. In *2025 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Duncan McElfresh, Sujay Khandagale, Jonathan Valverde, Vishak Prasad C, Ganesh Ramakrishnan, Micah Goldblum, and Colin White. 2023. When do neural nets outperform boosted trees on tabular data? *Advances in Neural Information Processing Systems*, 36:76336–76369.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. Pmlr.

Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE symposium on security and privacy (SP)*, pages 691–706. IEEE.

Sérgio Moro, Paulo Cortez, and Paulo Rita. 2014. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31.

Jaehyun Nam, Kyuyoung Kim, Seunghyuk Oh, Jihoon Tack, Jaehyung Kim, and Jinwoo Shin. 2024. Optimized feature generation for tabular data via llms with decision tree reasoning. *Advances in neural information processing systems*, 37:92352–92380.

Christopher J O'Donnell and Roberto Elosua. 2008. Cardiovascular risk factors. insights from framingham heart study. *Revista Española de Cardiología (English Edition)*, 61(3):299–310.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31.

Bendi Venkata Ramana, M Surendra Prasad Babu, and NB Venkateswarlu. 2012. A critical comparative study of liver patients from usa and india: an exploratory analysis. *International Journal of Computer Science Issues (IJCSI)*, 9(3):506.

Smita Raut, Shraddha Patil, and Gopichand Shelke. 2021. Lung cancer detection using machine learning approach. *International Journal of Advance Scientific Research and Engineering Trends (IJASRET)*.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, and 1 others. 2020. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119.

Jack W Smith, James E Everhart, William C Dickson, William C Knowler, and Robert Scott Johannes. 1988. Using the adap learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the annual symposium on computer application in medical care*, page 261.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024. Table meets llm: Can large language models understand structured table data? a benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 645–654.

Paul Voigt and Axel Von dem Bussche. 2017. The eu general data protection regulation (gdpr). *A practical guide, 1st ed., Cham: Springer International Publishing*, 10(3152676):10–5555.

Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. 2020. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, and 1 others. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19.

Hangting Ye, Jinmeng Li, He Zhao, Dandan Guo, and Yi Chang. 2025. Llm meeting decision trees on tabular data. *arXiv preprint arXiv:2505.17918*.

I-Cheng Yeh. 2008. Blood transfusion service center. UCI Machine Learning Repository. https://doi.org/10.24432/C5GS39.

Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. 2018. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*.

Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep leakage from gradients. *Advances in neural information processing systems*, 32.