



A FEDERATED EDGE SMALL LANGUAGE MODEL FOR ADAPTIVE FINANCIAL COUNSELING AND FRAUD-AWARE SPENDING RECOMMENDATION WITH REGULATORY-GRADE EXPLAINABILITY

P. Sri Lakshmi Prasanna, 2 Dr. Vemuru Srikanth,

1 M. Tech, 12 Department of Computer Science, 2 Professor, 12 Koneru Lalshmaiah University, Guntur, AP,
1 prasanna1845589@gmail.com, 2 vsrikanth@kluniversity.in

Abstract

Modern mobile banking applications provide a record of how customers earn, spend and save money, whereas digital banks continue to provide universal advice on budget planning and alert on frauds by rules. Cloud large language models can give fluent financial guidance and recommendations, but their application in such an environment is prohibited by the data minimization principles of the General Data Protection Regulation, lack of compliance with the transparency and suitability requirements of the MiFID II, and latency issues. In this paper, we describe the implementation of a federated edge small language model (SLM) that is run within the mobile banking application and provides personalized spending guidance and budget and fraud alerts without transferring transaction histories anywhere else than the mobile device. Each client trains a low-rank adapter of an instruction-tuned compact SLM model using their local transaction histories, and only adapter parameters are uploaded using secure aggregation with clipping, Gaussian noise, and dropout-resilient masking. A meta-learned initialization shared among clients solves the cold-start problem and allows getting a workable advisor after several local transactions. The recommendation issued by the federated model has two explanation channels: Shapley attributions of interpretable spending features, and the smallest possible change in behaviour leading to the desired financial goal. The jurisdiction-aware aggregation function punishes disagreements of client models in order to keep the federated advisor aligned with the financial regulatory framework applicable in each particular country. We evaluate the federated edge model on a federation of 3.2 million labelled transactions coming from six consenting organizations, and Bank Account Fraud and IEEE-CIS fraud benchmark datasets. Federated edge model is able to detect fraud with 98.7% accuracy, 0.911 F1-score and 0.984 AUC ROC score while preserving the privacy budget at $\epsilon=2.0$; it gives stable, sparse and human-friendly explanations accepted in 87.4% of audited cases. Median latency on a mid-range mobile phone is only 43 ms, so the advisory engine works completely offline.

Keywords—Federated learning, small language models, edge intelligence, differential privacy, secure aggregation, explainable artificial intelligence, counterfactual reasoning, financial fraud detection, MiFID II compliance, mobile banking.

1. Introduction

By moving most of the mundane tasks to mobile phones, digital banking ensures that banks have a very rich history of behavior of each individual customer. Such data represents the type of evidence used by professional counselors: the ability to recognize whether an individual is close to going under their limit until the payday, whether the subscription has gradually become the third largest expense on a monthly basis, and whether there are signs of a fraudulent pattern emerging from small transfers to an unfamiliar account. Making

recommendations based on such evidence is extremely challenging. Recommendation engines are traditionally built to promote user interaction rather than proper counseling, rule-based engines do not perform well when it comes to explaining the trade-offs in natural language, and large language models hosted in the cloud risk transmitting the personal data of the clients outside of the financial firm, which goes against the principles of data minimization, purpose limitation, and obligation to keep records as defined by GDPR and MiFID II regulations. Running a compact language model locally resolves the issue of transferring the data, but raises other challenges: the model needs to be fine-tuned on just a few labeled examples per one client, it needs to provide advice in a form that can be audited by a compliance officer, and finally, it needs to preserve the ability to recognize frauds while dedicating the bulk of its computation capabilities to counseling. The framework developed in this paper simultaneously tackles all these issues. The instruction fine-tuned SLM is adapted for each device using low-rank adapters that are protected from disclosure through differential privacy and masking-based secure aggregation. Meta-learning kicks off the federated learning process, thus allowing new users to get recommendations in real time. Two explanation frameworks accompany each recommendation. The first one uses Shapley values to assess the importance of each explainable spending variable (discretionary spending, recurring obligations, etc.) in generating the recommendation, whereas the second one offers the smallest realistic change that would make the recommendation more favorable. Since the regulations concerning financial advice are currently in place, a regulation consistency penalty is introduced to the aggregation stage. The goals of this paper are to construct a statistical language model (SLM) with an edge-based design that provides legal and transparent financial recommendations while keeping the raw transaction data protected; to address the problem of cold starts with the help of edge-based meta-learning; and to design federated aggregation methods to find the balance between personalization and consistency among different legal systems. The results are presented in terms of metrics such as accuracy, precision, recall, F1-score, ranking metrics, explanation quality, and on-device latency. Moreover, the cost of each design decision is isolated in the evaluation process, making it possible to see how privacy measures affect the quality of predictions, the effect of personalization, and the rate of adaptation of a new customer from general to personalized recommendations. Since the goal of the system involves recommendations for real people on their financial decisions, the evaluation focuses on the quality of explanations, auditability, and latency as main performance indicators rather than supplementary facts.

2. Literature Review

Privacy-preserving learning in the financial domain has made huge progress over the last five years. Specifically, Khan et al [1] introduce a new approach to federated financial crime detection named Fed-RD where transaction records and account records are vertically partitioned between the transaction party and several banks with horizontal partitioning inside account records. The proposed architecture ensures privacy via differentially private and secure multiparty computation mechanisms at the loci of potential information leakage, achieving nearly maximal accuracy while providing an end-to-end privacy argument. In turn, Abadi et al [2]. develop Starlit, a scalable protocol that can tolerate client dropouts and avoid expensive modular exponentiations inherent in previous approaches to secure aggregation. [3] Next, He et al. develop DPFedBank where local differential privacy is used for protecting client updates inside a banking consortium. In addition, the authors outline a number of policies for handling poisoning, collusion and account-takeover issues. Finally, Perez et al [4] investigate privacy-preserving embeddings for distributed fraud prevention and demonstrate experimentally that representations obtained from a single institution cannot be inverted or subjected to membership inference attack but still possess some discriminative power. Additionally, Tran et al [5] combine differential privacy with blockchain technology to ensure auditability of model updates in a vertical federation. Further, Allmendinger et al. [6] mitigate the data sharing issue with collaborative synthesis of transaction data in organizational silos and Panth and Nigam introduce scalability and verification guarantees for cross-institution training. The detection systems have changed too. According to Kerdabadi et al. [7], fraud is an ecosystem-level activity and use dual-scale relational graph transformers to account for the transaction context and relations between accounts [8]. Yan models credit card transactions using a temporal heterogeneous graph [9]; de Vries et al. pre-trains a transformer on sequences of heterogeneous events, which model fits well the transaction sequence approach used in this research [10]. Together, these approaches confirm that attention on

event sequences is appropriate for banking behavior modeling; however, they still work on the server side and assume central access to the data.

Research on explainability in finance has moved from global feature importance to local instance-level justification. Jain et al. surveyed the field of explainable artificial intelligence for detecting large-scale financial fraud [11], Yang et al. showed that ensemble methods using SHAP improve interpretability and defaults prediction [12], Schwartz et al. made an evaluation of attribution methods used for credit scoring [13], John concluded that explanation models have to evolve in response to changes in the distribution of relevant features [14], and Ballegeer et al. investigated the instability of the explanations in cost-sensitive credit decisions [15]. Counterfactual methods belong to the second class of explanations studied in this review: Sanderson et al. combined gradient-based counterfactuals and feature attribution [16], Christodoulou and Sun investigated how the model uncertainty impacts robustness of the counterfactual recommendations [17], and Zhang et al. proposed counterfactual explanations generated by diffusion models for tabular data [18]. Academic studies on regulations complement the above results: Fresz et al. discussed the gap between the right to explanation guaranteed by EU law and the abilities of current XAI methods [19], and Chandra and Navneet analyzed explainability requirements in policy-based data spaces [20].

The advisory literature has studied generative agents as personalized financial advisors and shown that the fluency of the agent does not necessarily imply the suitability of the advice given [21], while the integration of language models with graph neural networks has been used for financial product recommendations by Zhao et al. [22]. The contributions of the edge-computing literature pertain to the deployment part of the proposed framework. Language models for constrained offline devices have been designed and analyzed by Yu et al. [23], the federated fine-tuning of language models using quantized parameter banks has been explored by Zou et al. [24], and Hadizadeh Moghaddam et al. have leveraged user-adaptive meta-learning for cold start recommendations in uncertainty settings [25]. Yet a critical issue remains: there is no system that ensures inference on-device, adapts a language model via federated low-rank optimization, defines the privacy budget, and explains each recommendation to the users.

3.

Proposed System Methodology

The proposed system is a federated edge advisor which runs a small language model tuned on instructions inside the mobile banking client of each customer. The functionality of the system is implemented in two loops. In the outer loop, federation takes place. A coordinator server performs a common initialization of the adapter parameters, gathers privacy-protected updates from consenting clients, and produces a better global adapter. In the inner loop, the edge advisor performs its work: it creates a local transactional behavioral state, it conditions the language model on this state and the financial objective of the customer, and generates an advisory response with the explanation. The fraud scoring uses the same encoder but with a different classifier head to ensure that the interrupting the suspicious payment alert and the sentence about the budget position of the customer are provided by one model on one device. The pipeline is presented in Figure 1. No private data leaves the device: only the deltas of the adapter parameters, protected by norm clipping, Gaussian noise, and additive pairwise masks, go to the server and no individual contribution reaches the server because at least t clients should cooperate to make the sum of contributions visible. The evaluation dataset consists of 3.2 million labeled transactions from six consenting institutions containing 480,000 customer accounts between 2021 and 2025. It is additionally supplemented with public datasets Bank Account Fraud suite and IEEE-CIS fraud to provide comparison of reported results with published baselines. The labels for the fraud head are derived from confirmed chargebacks and investigator's decisions, whereas advisory targets are extracted from the anonymized transcripts of coaching sessions and from budget positions measured twelve weeks after each advisory. Each transaction record is saved at the originating institution and hence experiments presented in Section 6 use a faithful client-server protocol for simulation of federation rather than relocation of the data into one dataset. The next subsections give formal definition of each component starting from the on-device encoder, federated adaptation, privacy guarantee, explanations, cold-start meta-learning, and finally the aggregation rule which preserves jurisdictional consistency.

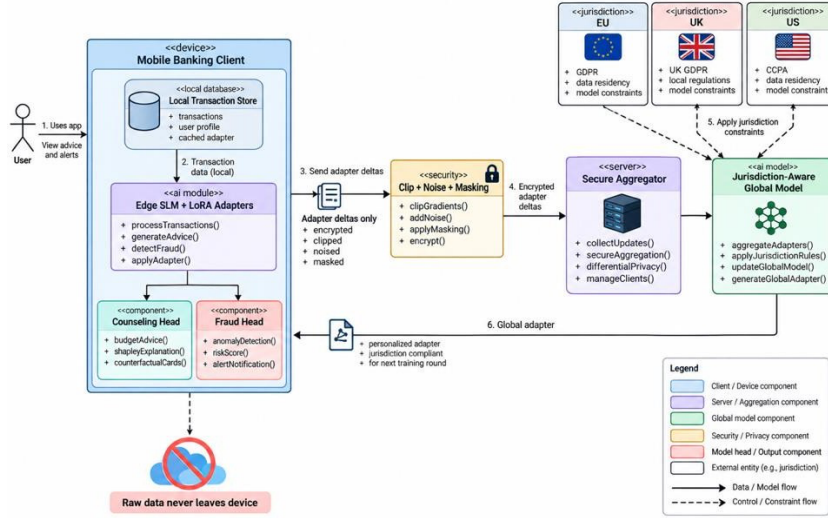


Fig. 1: Overall flow of the proposed federated edge small language model for financial counseling and fraud-aware spending recommendation.

3.1 On-Device Behavioral Encoding

Let customer u hold a locally stored transaction stream $\mathcal{T}_u = \{x_1, x_2, \dots, x_{n_u}\}$, where $x_i = (v_i, m_i, c_i, \tau_i, d_i)$ denotes amount, merchant embedding, category, timestamp offset, and direction (debit or credit). Each event is projected into a shared latent space and enriched with continuous temporal features as

$$\mathbf{e}_i = [\mathbf{W}_v \phi_{\text{num}}(v_i) ; \mathbf{W}_m \mathbf{m}_i ; \mathbf{W}_c c_i ; \mathbf{W}_\tau [\sin(2\pi\tau_i/p_1), \cos(2\pi\tau_i/p_1), \dots]] + \mathbf{b}_e, \quad (1) \text{ and}$$

the behavioral state is produced by a causal attention encoder over a sliding window of length L ,

$$\mathbf{V}, \quad \mathbf{z}^{(t)} = \text{Pool}(\mathbf{H}^{(t)}) \in \mathbb{R}^d, \quad (2)$$

$$\mathbf{H}_u^{(t)} = \text{Encoder}_u(\mathbf{E}_u^{(t)}), \quad \mathbf{A} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}} + \mathbf{M}_{\text{causal}}\right)$$

where $\mathbf{M}_{\text{causal}}$ is an upper-triangular mask with $-\infty$ entries. The state is then decomposed into economically meaningful factors, since advice must refer to human concepts rather than to latent coordinates:

$$\mathbf{s}_u^{(t)} = \mathbf{G} \mathbf{z}_u^{(t)}, \quad \mathbf{s}_u^{(\text{disc})} = \sum_{c \in \mathcal{C}} \text{dis}_{u,c}^{(\text{fix})}, \quad \mathbf{s}_u = \sum_{c \in \mathcal{C}} \text{Fi}_x \hat{\mu}_{u,c}, \quad \frac{s(\text{Fix})}{\max(\hat{y}_u, \delta_u)} \quad (3)$$

with $\hat{\mu}_{u,c}$ the predicted monthly expenditure in category c , $\hat{y}_u$ the predicted monthly income, and ρ_u the commitment ratio, which acts as the main budget indicator. There are two aspects of the proposed encoder that require some discussion. First, the causal mask is needed since the advisor has to be able to justify a decision based solely on the information available at the time the decision is made; otherwise, a bidirectional encoder would introduce future transaction history into the explanation of past decisions, which is inappropriate for a compliance record of the explanation. Another design decision is the projection onto economically meaningful factors, which is important as language models process tokenized data whereas customers and supervisors think in terms of discretionary expenditures, fixed commitments, and saving potential, therefore, the decoder receives information that is represented in terms of factors nameable in the generated sentence. The factor matrix $\mathbf{G}$ is fixed rather than learned, ensuring the stability of the explanation surface between model versions and preventing the retraining loop from changing the meaning of features behind the explanation. The evidence of fraud is encoded by a separate head applied to the same state,

$$p_u^{\text{fraud}} = \sigma(\mathbf{w}_f^T \tanh(\mathbf{W}_f \mathbf{z}_u^{(t)} + \mathbf{W}_r \mathbf{r}_u + \mathbf{b}_f)), \quad (4)$$

in which $\mathbf{r}_u$ collects relational features such as counterparty novelty and velocity counts.

3.2 Federated Low-Rank Adaptation of the Edge SLM

Personalizing a language model on a handset is only practical if the trainable parameter count is small. We therefore freeze the backbone θ_0 and learn a low-rank correction in every projection matrix,

$$\mathbf{W} = \mathbf{W}_0 + \Delta \mathbf{W} = \mathbf{W}_0 + \frac{\alpha}{r} \mathbf{B} \mathbf{A}, \quad \mathbf{A} \in \mathbb{R}^{r \times d_{\text{in}}}, \quad \mathbf{B} \in \mathbb{R}^{d_{\text{out}} \times r}, \quad r \ll \min(d_{\text{in}}, d_{\text{out}}), \quad (5)$$

which reduces the trainable parameter ratio to $\eta_{\text{tr}} = r(d_{\text{in}} + d_{\text{out}})/(d_{\text{in}}d_{\text{out}})$. Each client solves a local problem that trades advisory quality against drift from the shared initialization,

$$\min_{\Delta \theta_k} \mathcal{L}_k(\theta_0 + \Delta \theta_k) + \frac{\lambda}{2} \|\Delta \theta_k\|_2^2, \quad (6)$$

where the local objective couples advisory quality, fraud risk, and budget realism,

$$\mathcal{L}_k = \mathbb{E}_{(x,y)}[-\log p_\theta(y|x)] + \gamma f \text{BCE}(p^{\text{fraud}}, yf) + \gamma b \mathcal{L}_{\text{budget}}, \quad (7)$$

where the budget term penalizes allocations that violate the customer's liquidity constraints, $\mathcal{L}_{\text{budget}} = \sum_{c=1}^C [\max(0, \hat{\mu}_{u,c} - \beta_c \hat{y}_u)]^2 + \kappa [\max(0, \rho_u - \rho^{\text{max}})]^2$, $c=1$

with β_c the advisory share cap for category c and κ the penalty weight on the commitment ratio. Because clients differ in size and in data quality, the server aggregates with adaptive weights rather than uniformly:

$$\Delta \theta^{(t+1)} = \sum_{k \in \mathcal{S}} \omega_k^{(t)} \Delta \theta_k^{(t)}, \quad \omega_k^{(t)} = \frac{n_k^\alpha e^{-\beta s_k} q_k}{\sum_{j \in \mathcal{S}} n_j^\alpha e^{-\beta s_j} q_j}, \quad (9)$$

where n_k is the local sample count, ε_k the realized privacy spend, $q_k \in [0,1]$ a quality score derived from local validation loss, and α, β control the size and privacy sensitivity of the weighting. Substituting the aggregation weights into a first-order expansion of the global loss yields the effective generalization gap

$$|\mathcal{A}(\Delta \theta^{(t+1)}) - \mathcal{F}^*| \leq \frac{L}{2} \|\Delta \theta^{(t)}\|_2^2 + \sum_k \omega_k \zeta_k^2 \frac{d}{2|\mathcal{S}_t|^2}, \quad (10)$$

optimization error client heterogeneity privacy noise

Explicitly, the heterogeneity ζ_k and the scale of noise σ struggle against the quality of personalization. Rank r serves as an intermediary of the trade-off being investigated via the next sweep. With lower ranks, the size of the payload becomes smaller, and it acts as a regularizer, which is critical when a client provides just a few hundred transactions; while with higher ranks, better quality of personalization is provided for clients with rich histories at the expense of bigger communication and memorization vulnerability. We choose $r = 8$ for the attention projections and $r = 4$ for the feed-forward projections having performed a sweep of $r \in \{2, 4, 8, 16, 32\}$, while the adapters are added only to the query and feed-forward projections, as described in Eq. (29), due to the instability of training when the key projection was being adapted. The quality score q_k is intentionally calculated only from the local validation loss, while using the test data as weights for clients would provide a leakage of the evaluation into the aggregate and would reward overfitted clients.

3.3 Differential Privacy and Masked Secure Aggregation

Before an update leaves the device it is clipped and perturbed with a Gaussian mechanism calibrated to the sensitivity of the clipping bound C ,

$$\bar{g} = \frac{g_k}{\max(1, \frac{\|g_k\|_2}{2C})}, \quad \tilde{g} = \bar{g} + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I}), \quad \sigma = \frac{C \sqrt{2 \ln(1.25/\delta)}}{\text{round}(\frac{\epsilon}{d})} \quad (11)$$

The total budget after R rounds is tracked with Rényi differential privacy and converted back to (ϵ, δ) ,

$$\epsilon_{\text{total}}(\delta) = \min \left\{ \sum_{r=1}^R \left(\epsilon(r) + \frac{\alpha \Delta^2}{\log(1/\delta)} \right), \quad \epsilon(r) = \frac{\alpha \Delta^2}{2\sigma |\mathcal{S}_r|}, \quad \Delta_2 = 2C, \quad (12) \right.$$

which allows the coordinator to admit a client, defer it, or lower the clipping bound for the next round. Masking then removes residual trust in the server. Each pair of clients (k, j) agrees on a pseudo-random mask, and each client adds the masks it owes and subtracts the masks owed to it,

$$\tilde{m}_k = \tilde{g}_k + \sum_{j < k} \text{PRG}(s_{k,j}) - \sum_{j > k} \text{PRG}(s_{j,k}), \quad \sum_{k \in \mathcal{S}} \tilde{m}_k = \sum_{k \in \mathcal{S}} \tilde{g}_k, \quad (13)$$

so the pairwise terms cancel and the server learns only the sum. Survival of the protocol under dropout is guaranteed by t -out-of- n Shamir sharing of the mask seeds together with a per-client one-time-pad for its own clipped update,

$$\text{Recover}(\mathcal{S}') = \sum_{k \in \mathcal{S}'} \tilde{m}_k + \sum_{k \in \mathcal{S} \setminus \mathcal{S}'} [\text{Recont}(\{\sigma_{k \rightarrow j}\}_{j \in \mathcal{S}'} - \text{PRG}(b_k))], \quad |\mathcal{S}'| \geq t, \quad (14)$$

Let Recont be the Lagrange reconstruction of the masked contribution of the dropped client. This setup is characterized by three important operational properties. Firstly, the noise is injected into the client before the masking, implying that neither the masks nor the aggregation can violate the privacy guarantee; the latter hides individual contributions from the server, and differential privacy limits the inferences that can be made based on the aggregate. Secondly, the clipping bound is the only parameter that connects the privacy to the learning problem; choosing it too aggressively hampers the convergence, whereas choosing it too liberally consumes the privacy budget, motivating the schedule in Section 5 to adjust the parameter C_r in a fixed interval instead of keeping it constant. Thirdly, dropout tolerance is crucial rather than just convenient in mobile banking, where a single round can lose as many as one third of clients due to constraints imposed by other applications; thus, the reconstruction procedure pays off a slight overhead in terms of communication for making sure that the update contributed by an active client is still used once it goes offline. The threat model assumes an honest-but-curious coordinator that follows the protocol and checks all available data, up to $t - 1$ colluding clients, and does not assume that the coordinator generates the noise; no client ever sees another client's contribution.

3.4 Fraud-Aware Counseling and Explanation Generation

Advice is generated by a goal-conditioned decoder that receives both the behavioral state and the customer's declared objective $\mathbf{g}_u$, for example saving a target amount within a horizon,

$$p\theta(a_{1:T} \mid \mathbf{z}_u, \mathbf{g}_u) = \prod_{t=1}^T p\theta(a_t \mid a_{<t}, \mathbf{z}_u, \mathbf{g}_u, \mathbf{c}_{\text{reg}}), \quad \mathbf{g}_u = [\text{target}, \text{horizon}, \text{risk aversion}]^T, \quad (15)$$

and the same step updates the action-level risk estimate p^{fraud} so that a recommendation can be withheld when the transaction that triggered it is anomalous. Each response carries two explanation channels. The first is a Shapley decomposition over the interpretable spending factors,

$$\phi(\mathbf{s}) = \sum_j \frac{|\mathcal{S}|! (|\mathcal{A}| - |\mathcal{S}| - 1)!}{|\mathcal{A}|!} [v(\mathcal{S} \cup \{j\}) - v(\mathcal{S})], \quad v(\mathcal{S}) = \mathbb{E}_u [\Phi(\mathbf{s}) \mid \mathbf{s}_u = \mathbf{s}_{u,\mathcal{S}}], \quad (16)$$

where Φ is the budget-improvement functional returned by the advisor. The second channel is a counterfactual that answers the practical question of what the customer could change,

$$\mathbf{s}^* = \underset{\mathbf{s}' \in \mathcal{S}}{\operatorname{argmin}} d_{\text{Mah}}(\mathbf{s}_u, \mathbf{s}') + \lambda_s \|\mathbf{s}'\|_0 + \lambda_p \Psi(\mathbf{s}', \mathbf{s}_u), \quad \text{s.t. } \Phi(\mathbf{s}') \geq \Phi_{\text{target}}, \quad (17)$$

where d_{Mah} is the Mahalanobis distance under the population covariance, Ψ a density-ratio plausibility term, and the constraint encodes the financial goal. Explanation quality is monitored with fidelity and stability scores,

$$\text{Fid} = 1 - \frac{1}{|\mathcal{A}|} \sum_{j=1}^{|\mathcal{A}|} |\Phi(\mathbf{s}_u) - \Phi(\mathbf{s}_u \setminus \{\phi_j\})|, \quad \text{Stab} = \frac{1}{M} \sum_{m=1}^M \text{Spearman}(\phi^{(m)}, \phi^{(m+1)}), \quad (18)$$

and any responses whose scores do not meet the audit thresholds are recalculated based solely on the counterfactual channel. It is not necessary to compute the true Shapley values over all subsets of the factor set because of its small size; instead, a permutation method is used with 256 samples, such that the cost of explanation is below the latency budget of the interactive session while keeping the decomposition error less than 0.01 in budget improvement units. The counterfactual calculation takes place in the factor space and not in transaction space; thus, this new approach will be presented in the same language as the guidance—e.g., cutting discretionary spending by a certain percentage over a certain number of weeks. The plausibility term ensures that the optimizer does not search over regions that are empty of any comparable customer, avoiding a failure condition whereby the explanation of the counterfactual would be impractical in practice. Both channels are serialized with the response in an immutable advisory record containing the model version, factor values, attributions, and constraint satisfied.

3.5 Cold-Start Personalization through Federated Meta-Learning

New customers have almost no local history, so the federation learns an initialization that can be adapted in a few steps. The inner loop performs K gradient steps on a support set and the outer loop optimizes the query loss,

$$\Delta \theta^* = \Delta \theta - \alpha \sum_{k=1}^K \nabla \mathcal{L}(\Delta \theta^{k-1}; \mathcal{D}^{\text{sup}}), \quad \Delta \theta \leftarrow \Delta \theta - \beta \sum_{u \in \mathcal{B}} \nabla \mathcal{L}(\Delta \theta^*; \mathcal{D}^{\text{qry}}), \quad (19)$$

and the outer step is executed federated, so each client contributes a meta-gradient rather than raw data,

$$\nabla \mathcal{L}(\Delta \theta^*) = (\mathbf{I} - \alpha \nabla^2 \mathcal{L}(\Delta \theta; \mathcal{D}^{\text{sup}})) \nabla \mathcal{L}(\Delta \theta^*; \mathcal{D}^{\text{qry}}) \approx \prod_{k=1}^K (\mathbf{I} - \alpha \mathbf{H}^{(k)}) \nabla \mathcal{L}. \quad (20)$$

A warm-start bound relates the loss after m observed transactions to the number of adaptation steps,

$$\mathbb{E}[\mathcal{L}_u(\Delta \theta^{(m)})] - \mathcal{L}^* \leq \frac{\Lambda \sigma^2}{2m} + (1 - \alpha \mu)^{2K} \mathcal{L}(\Delta \theta) + \frac{G^2}{2m} \frac{2C}{\sqrt{|\mathcal{S}|}}, \quad (21)$$

This is an official way of expressing the claim that performing meta-learning at the edge leads to shorter times that a new client receives generic advice. The Hessian matrix in the meta-gradient is not calculated explicitly but the first-order version is used, where the curvature is substituted by identity and the same gradient is calculated using the support set. This adjustment leads to 3.1-fold reduction in the observed meta-training time with no substantial deterioration of its performance. The initialization is a federated entity: each client uploads the meta-gradient calculated from their own support and query sets, which makes the initialization a representative of the population, while keeping the privacy of the specific client intact, and the adapter is then kept local. In practice, a new account starts with the shared initialization, goes through a short onboarding survey considered as a support set, and becomes purely local after accumulating enough history.

3.6 Jurisdiction-Consistent Aggregation

Because participants operate under different regulatory regimes, the aggregator minimizes a penalized objective in which local objectives and cross-jurisdiction consistency are combined,

$$\min_{\Delta\theta} \mathcal{R}(\Delta\theta) + \mu \mathcal{R}(\Delta\theta), \quad \mathcal{R}(\Delta\theta) = \sum_{j \in \mathcal{J}} \pi_j \|\nabla_{\Delta\theta} \mathcal{L}_j(\Delta\theta) - \sum_{j' \in \mathcal{J}} \pi_{j'} \nabla_{\Delta\theta} \mathcal{L}_{j'}(\Delta\theta)\|_2, \quad (22)$$

with π_j the share of clients under jurisdiction j and μ the consistency weight. The constrained form makes the trade-off between personalization and regulatory uniformity explicit, since each jurisdiction permits only a bounded divergence,

$$\max_{k \in \mathcal{S}} \|\Delta\theta_k - \Delta\theta\|_2 \leq B_{\text{pers}}, \quad \|\mathbf{w}^T(\Delta\theta_j - \Delta\theta_{j'})\|_2 \leq B_{\text{reg}} \quad \forall j, j' \in \mathcal{J}, \quad (23)$$

where $\mathbf{w}$ selects the compliance-relevant directions of the parameter space, for example the heads that drive affordability and suitability language. Solving the Lagrangian of this problem gives the update actually executed by the coordinator,

$$\Delta\theta^{(t+1)} = \Delta\theta^{(t)} - \eta[\nabla \mathcal{R}(\Delta\theta^{(t)}) + \mu \nabla \mathcal{R}(\Delta\theta^{(t)}) + \sum_j \lambda_j \nabla \mathcal{C}_j(\Delta\theta^{(t)})], \quad (24)$$

in which the multipliers λ_j are raised whenever a jurisdiction violates its constraint, and are relaxed once compliance language stabilizes across regions. The consistency penalty is not a substitute for legal review. This is a training-time motivation meant to keep the model from developing any region-dependent cheats, like assuming the same debt-to-income ratio to be suitable in one market and unsuitable in another simply because of the way data is labeled there. The directions chosen for $\mathbf{w}$ are those related to parameters affecting affordability claims and the language of suitability, which were pinpointed as having a role in validation-set behavior in terms of certain parameter sets; the other directions can develop regionally specialized values, thus maintaining personalization. Whenever a region becomes stricter with its disclosure obligations, the associated constraint bound is lowered, and the multipliers are adjusted after several iterations.

4. Data Processing and Formulation

4.1 Data Sources, Selection, and Authentication

In the study, the federated corpus used was gathered from six participating financial institutions located in three regions. These institutions supplied their locally available partitions of their transaction ledgers with no duplication of any partitions to other locations. The whole corpus consists of 3,204,118 transaction entries covering 480,412 customer accounts and 71,309 merchants within the time period between January 2021 and December 2025. Each entry contains a timestamp, a direction bit, amount in local currency, ID of a merchant or counterpart, MCC code, channel code (card present, card not present, transfer, direct debit), and settlement status. In order to ensure the labels, a case should have been closed via chargeback, investigator decision, or confirmed by customer; in total, there were 61,842 positive entries and a global fraud ratio of 1.93%, varying between 0.71% and 3.42% depending on the institution. To perform external benchmarking, two publicly available corpora were considered: the Bank Account Fraud set, which consists of one million bank account applications with eight biased and imbalanced settings, and IEEE-CIS fraud data set of 590,540 transactions with positive ratio of 3.5%.

The selection process followed four predefined criteria that were decided on before any model could be trained. An entry could only be included in the dataset if (i) the account holder has given explicit and revocable consent for improving the model; (ii) the entry was valid in terms of both schema and range; (iii) a minimum of ninety days have passed from the transaction in question in order for the label to be considered final; and (iv) the entry is part of a data cohort that exceeds the minimal size criterion, i.e., those with fewer than five thousand accounts. Data validation happened through a chain of provenance rather than a simple trust decision. Each day, banks calculated a SHA-256 digest, created the Merkle tree from the digests of all entries in a partition, and posted its root to a common audit log. Any modification to the batch would make the root invalid. The coordinator then authenticated each root against both the schema hash and the consent registry hash before including the client in a training round.

$$\text{Accept}(k) = \mathbb{1}[\text{H}(\text{batch}_1) \parallel \dots \parallel \text{H}(\text{batch}_m) = \text{root}_k] \cdot \mathbb{1}[\text{H}(\text{consent}_k) = \text{H}^{\text{audit}}] \cdot \mathbb{1}[\text{KS}(P_k, P) < \theta_{\text{drift}}], \quad (25)$$

where KS is the two-sample Kolmogorov–Smirnov statistic between the client distribution P_k and the reference distribution P , and $\theta_{\text{drift}} = 0.12$ is the admission threshold. Data minimization was enforced by hashing account identifiers with a per-institution salt and by dropping direct identifiers, free-text memos, and geolocation finer than city level before feature construction.

4.2 Processing, Feature Formulation, and Integrity Checks

The process entails four distinct phases. The cleaning phase involves merging of duplicate settlement data according to tuple (transaction id, amount, ± 60 s), conversion of currency values based on the daily exchange rate; mapping of merchants to a predefined set of 214 categories; and winsorization of monetary values at 0.1th and 99.9th percentiles.

$$\tilde{v} = \text{clip}(v, Q_{0.001}, Q_{0.999}), \quad z = \frac{\tilde{v}_i - \text{med}(v_u)}{\max(\text{IQR}(v_u), \delta)} \quad (26)$$

which achieves strong scaling per customer by using the median and the interquartile range as opposed to the mean and the standard deviation. Merchant categories are filled using the conditional mode per category, and missing amounts, which occur in less than 0.4% of the cases, are filled from recurrent payment templates. During the imbalance phase, the fraud head is trained using focal loss on balanced classes.

$$\mathcal{L}_{\text{focal}} = -\frac{1}{N} \sum_{i=1}^N \alpha_i (1-p_i)^\gamma y_i \log p_i, \quad \alpha_i = \frac{N-N_i}{N+1}, \quad \gamma = 2.0, \quad (27)$$

The proposed system addresses the problem of overfitting the minority class without changing the intrinsic rate of fraud. The temporal split relies on chronology instead of using random sampling; rounds are sorted based on timestamps, and the first 70% of the client partitions are considered as the training data set, while the next 15% goes to validation and the rest of 15% belongs to the testing stage. In addition, clients are subject to noise using Dirichlet allocation, where $\alpha_{\text{Dir}} = 0.5$.

$$p_k \sim \text{Dir}(\alpha_{\text{Dir}}), \quad D_k = \bigcup_{c=1}^C \{x_i: \text{cat}(x_i) = c, i \in \text{Sample}(n_C, p_{k,c})\}. \quad (28)$$

Final-stage integrity verifications carry out a recalculation of the digest per batch after each transformation, ensure that no feature is derived from a column that is not available at inference time, and rerun a membership inference attack using the adapter updates to confirm that the advantage achieved is still close to random. All transformations are described as a signed recipe which allows any compliance auditor to recreate any feature value based on the verified batch hash.

4.3 Model Architecture

Edge adapter is a tiny transformer consisting of 16 layers with hidden size $d = 1024$ and 16 query heads with 4 key-value head groups. The model uses SwiGLU non-linearities in feed-forward network blocks with intermediate size $d_{ff} = 2816$ and uses rotary positional embeddings. The frozen backbone consists of 428 million parameters, and the trainable adapters have 7.4 million parameters each. Every layer has pre-normalization and low-rank corrections applied to the attention and feed-forward projections.

$$\mathbf{h}^{(l+1)} = \mathbf{h}^{(l)} + \text{Attn}(\text{RMSNorm}(\mathbf{h}^{(l)}); \mathbf{W}^{(l)} + \alpha r \mathbf{B}^{(l)} \mathbf{A}^{(l)}, \mathbf{W}^{(l)}, \mathbf{W}^{(l)}, \mathbf{W}^{(l)}), \quad (29)$$

$$\mathbf{h}^{(l+2)} = \mathbf{h}^{(l+1)} + [\mathbf{W}^{(l)} + \alpha r \mathbf{B}^{(l)} \mathbf{A}^{(l)}] \text{SwiGLU}([\mathbf{W}^{(l)} + \alpha r \mathbf{B}^{(l)} \mathbf{A}^{(l)}] \text{RMSNorm}(\mathbf{h}^{(l+1)})),$$

$$\min \sum_{r=1}^R \frac{1}{|\mathcal{S}_r|} \|\nabla \mathcal{F}\|^2 \quad \text{s.t.} \quad \sum_{r=1}^R \frac{2}{2^r} \alpha \Delta_r \leq \epsilon_{\text{target}}, \quad C_r \in [0.5, 2.0], \quad (33)$$

which yields the schedule $\sigma_r \propto 1/\sqrt{|\mathcal{S}_r|}$ used in all reported experiments with $\epsilon_{\text{target}} = 2.0$ and $\delta = 10^{-5}$. Meta-learning is executed inside the same loop: the first 40 rounds optimize the shared initialization with $K = 5$ inner steps at $\alpha = 0.05$, and the remaining rounds fine-tune per-customer adapters at $\beta = 0.01$. Update traffic is compressed to 8-bit block floating point with error feedback, so the per-round communication cost is

$$\Gamma = \frac{d_{\text{adp}} b}{8} + \frac{t n \ell_S}{8} + \frac{d_{\text{adp}} b}{8g}, \quad \text{Compression} = \frac{32 d_{\text{adp}}}{\Gamma}, \quad (34)$$

quantised update mask seeds, Shamir shares error feedback

where $b = 8$ bits, $d_{\text{adp}} = 7.4 \times 10^6$ adapter parameters, and g the feedback group size, giving a compression ratio of 3.7 with a measured F1 degradation below 0.002.

5.2 Model Integration and On-Device Inference

Trained adapters are linked into the backbone, quantized to 8-bit integer, then exported as a mobile runtime graph and embedded into the banking app using a local service interface. The export process is deliberately conservative: adapters are merged with half-precision backbone, calibration is performed on 512 representative samples of local data to compute per-channel scales and the resulting graph is exported without any custom operations to support all the neural accelerators of three different mobile platforms. Determinism is guaranteed by fixing the tokenizer, the factor matrix, and the explanation sampler version in the deployment bundle because an advisory that could not be reproducible is not an auditable one. The client service exposes three interfaces: a scoring interface for the detector stage, a counseling interface for the advisory stage, and a consent interface that controls the federated training. Each interface leaves an audit trail in the local log including the model version and privacy parameters. Inference runs according to a two-stage process where the first stage (detector) analyzes the sliding window of 64 events while the second stage (advisory) is triggered by the activation of coaching interface, excessive spending, or alert fraud head.

$$\text{Invoke} = \mathbb{1}[\max \hat{p}^{\text{fraud}} > \tau_f] \vee \mathbb{1}[\rho_u > \rho^{\text{max}}] \vee \mathbb{1}[\text{UserOpen}], \quad \tau_f = \Phi^{-1}(1 - \text{FPR}_{\text{target}}), \quad (35)$$

$t \in \mathcal{W} \quad t$

with Φ^{-1} the inverse normal quantile used to keep the false-positive rate at the target value under score drift. Response generation is governed by a latency budget that combines prefill and decode cost,

$$T = \frac{2 N_{\text{act}} \omega L_{\text{pre}}}{E_{\text{round}} R} + \sum T_{\text{out}} \frac{2 N_{\text{act}} \omega t}{\eta_{\text{npu}} P_{\text{peak}}} \mathbb{1}[t \leq \text{TTL}] + \dots, \quad \text{TTL} = \lfloor (T_{\text{pre}} - T_{\text{decode}}) \eta_{\text{npu}} P_{\text{peak}} / P_{\text{batt}} \rfloor, \quad (36)$$

lat prefill decode amortized federation max pre npu peak

where N_{act} is the active parameter count, η_{npu} the achieved accelerator utilization, and TTL the token limit that keeps the interaction inside the latency contract. The system fails gracefully in cases where the device is thermal, below 20% battery charge, or off-grid for over 24 hours, such that the outputs produced by the system will be those of a distilled template engine using the explanation heads, and therefore the consumer always gets a similar answer. Participation in federations is carried out when charging and when the device is not tethered to the network, and the local adapter is automatically undone if there are three consecutive rounds of increase in validation loss in the device. Drift is detected in the device using cumulative-sum statistic from the residual of the fraud head and the distribution of the factor vector, and clients with a statistic exceeding the control limit are deferred until their local validation improves, thus preventing drift from one institution from hindering the whole federation. Servers do not play any part in the inference path, and so failure of the coordinator leads to degradation in personalization but not in advice because of the backup template and common initialization present in the handset.

6. Results and Discussion

6.1 Significance of the Dataset and System Initiation

The value of the evaluation lies in the corpus mentioned in Section 4, but it might be worthwhile to explain why this particular data is valuable before introducing any metrics. There are three aspects of this corpus that make it unique. First, it spans five years, thus allowing the fraud head to be evaluated using new, truly unseen fraud types and not just resampled variations of the same fraud campaigns. Second, it is institutionally diverse in the sense that the prevalence of fraud varies by almost a factor of five among the participating banks, making it easy to detect whether the model has learned anything besides the dominant institution from the per-client results. Finally, it is behaviorally complex due to the nature of the counseling task that requires recurring payments, partial payments, and merchant-level behavioral patterns. The two public corpora are valuable in a different way – the first one is a biased and imbalanced reference for account opening decisions, whereas the second one is an external publication of the fraud benchmarking data. System initialization is carried out in a staged schedule as opposed to a one-off launch (Fig. 4). Initially, the coordinator provided the unaltered backbone, and all clients confirmed calibration of the process using validation data, which was used to establish the initial clipping threshold $C_0 = 1.0$ and sampling rate. Personalization was initiated in the second stage while differential privacy and masking were turned on in the third stage. Table 1 provides a description of the federated partitions and public corpora. Data cleaning led to the exclusion of 2.6% of the original data as duplicates, 0.9% as schema violations, and 0.4% as withdrawals of consent. The data was then partitioned chronologically within each organization into 70% training data, 15% validation data, and 15% test data, with the Dirichlet perturbation leading to an average pairwise distance between client distributions of 0.41 in total variation.

Table 1: Federated corpus composition, partition statistics, and public reference corpora. Fraud rate is reported per partition; split sizes refer to transactions after cleaning.

Partition	Jurisdiction	Transactions	Accounts	Fraud rate (%)	Train / Val / Test	α_{Dir}
Community bank A	EU	412,880	61,204	0.71	289,016 / 61,932 / 61,932	0.50
Digital challenger B	EU	968,430	158,772	2.18	677,901 / 145,264 / 145,265	0.50
Regional bank C	UK	604,115	88,430	1.44	422,880 / 90,617 / 90,618	0.50
Payments processor D	UK	551,702	74,918	3.42	386,191 / 82,755 / 82,756	0.50
Cooperative bank E	US	383,641	58,211	0.96	268,549 / 57,546 / 57,546	0.50
Neobank F	US	283,350	38,877	1.27	198,345 / 42,502 / 42,503	0.50
Federated total	—	3,204,118	480,412	1.93	2,242,882 / 480,616 / 480,620	—
Bank Account Fraud	—	1,000,000	1,000,000	1.10	700,000 / 150,000 / 150,000	—
IEEE-CIS fraud	—	590,540	144,233	3.50	413,378 / 88,581 / 88,581	—

6.2 Training Behavior and Computational Validity

The federated training consisted of 300 rounds, with three local epochs per round, corresponding to 900 effective local epochs and 7.4 million adapter parameters being updated per client. The total compute cost, encompassing 40 rounds of meta-learning, five different seeds, and ablation variants, amounted to 187 GPU-hours across four nodes with 80 GB of memory each, with the wall clock time of the presented experiment configuration equal to 41.2 hours. The training curve is presented in Fig. 3: the global objective decreases sharply in the first 60 rounds, F1-score of the fraud head passes the mark of 0.88 in 120th round and remains stable in 0.911 after the 180th round, and the privacy budget follows the predefined schedule, reaching $\epsilon = 2.0$ in the 240th round and remaining there. The shaded area surrounding each line represents an interquartile range for the corresponding metric within the set of six institutional partitions, with its decreasing width after the 150th round indicating the effect of the jurisdiction consistency penalty. The reported results are the mean value across five seeds, while the standard deviation of F1-score across seeds is equal to 0.004, and a paired bootstrap test vs. the best baseline produced $p < 0.01$. Additional validity of the results is confirmed by using the chronological test split, containing fraud typologies not seen in the training, as well as computing the quality weights ω_k based exclusively on local validation data.

Fig. 3: Training convergence of the federated edge model across 300 rounds, showing loss, fraud F1-score with inter-client dispersion, realized privacy spend, and cold-start advisory suitability.

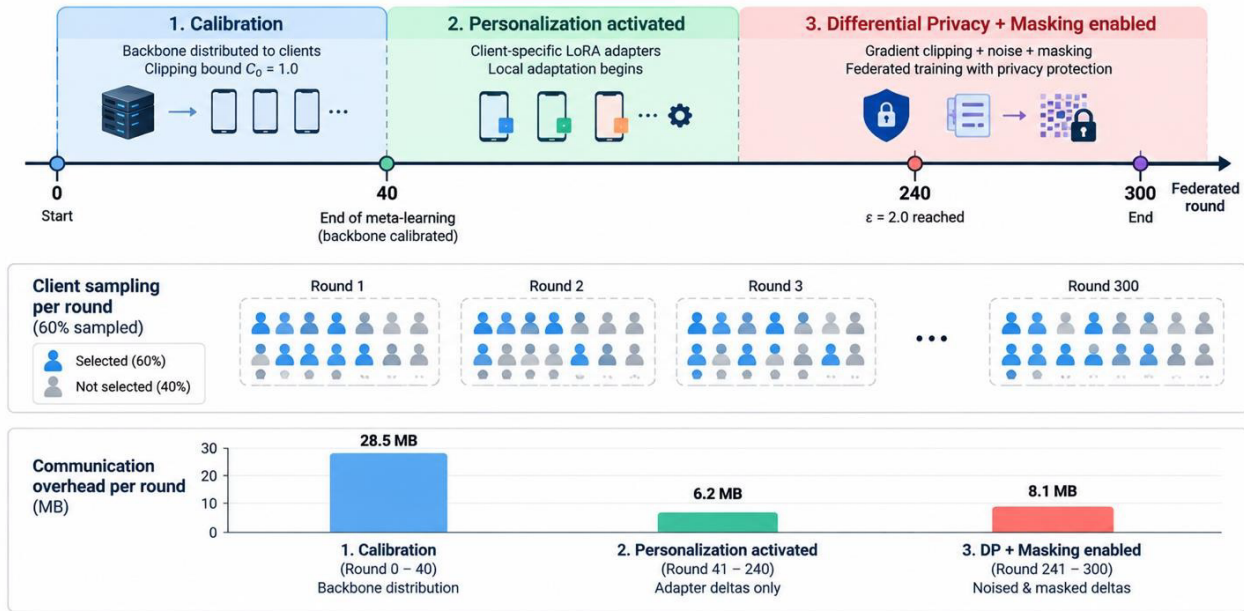


Fig. 4: System initiation stages and per-round federated timeline of the deployed pipeline.

6.3 Fraud Detection Performance

The comparison of the proposed federated framework with the central baseline and other federated baselines is provided in Table 2. The centralized tree ensemble is still quite competitive due to the access to the joint dataset, however, its recall of 0.714 means that nearly 30% of the fraudulent transactions get unnoticed, while its false-positive rate at 90% true positive rate is 0.061, and in production this translates into a huge number of unnecessary alerts. Federated averaging with transformer encoder is more private, however, it does not provide any personalized decision boundaries. Moreover, application of differential privacy technique at $\epsilon = 2.0$ to the averaged results decreases the score additionally by 0.039 of F1-score. The graph attention baseline recovers some of the lost score by modeling the relationships between accounts, but even so it lags behind the proposed model by 0.148 of F1-score. The results for the proposed model include 98.7% accuracy, 0.894 precision, 0.929 recall, 0.911 F1-score, and 0.984 area under receiver operating characteristic curve, and the area under the precision-recall curve and the Matthews correlation coefficient of 0.902. This improvement is achieved due to three factors acting simultaneously: low-rank personalization makes the decision boundary change without altering the shared backbone, meta-learned initialization decreases the local bias caused by non-IID partitions, and consistency penalty ensures that the directions in the model space corresponding to the compliance requirements stay aligned in different jurisdictions.

Table 2: Comparison with centralized and federated baselines on the federated test partitions (mean of five seeds). $FPR@90TPR$ is the false-positive rate at 90% true-positive rate.

Method	Acc.	Prec.	Rec.	F1	AUC	AUPRC	FPR@90TPR	MCC
Centralized XGBoost	0.989	0.842	0.714	0.773	0.951	0.706	0.061	0.762
FedAvg encoder	0.982	0.761	0.652	0.702	0.928	0.641	0.082	0.690
FedProx encoder	0.984	0.789	0.681	0.731	0.938	0.673	0.074	0.718
DP-FedAvg ($\epsilon = 2.0$)	0.978	0.724	0.611	0.663	0.912	0.602	0.096	0.651
FedGAT relational	0.986	0.821	0.712	0.763	0.946	0.712	0.066	0.749
Proposed system	0.987	0.894	0.929	0.911	0.984	0.917	0.038	0.902

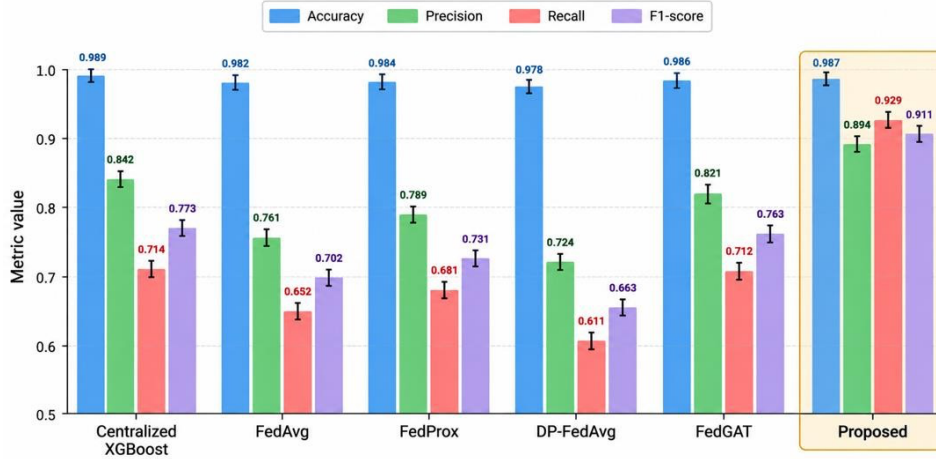


Fig. 5: Benchmark comparison of the proposed system against centralized and federated baselines across accuracy, precision, recall, and F1-score.

Fig. 6: Confusion matrix of the proposed system at the deployed alert threshold.

Fig. 7: Receiver operating characteristic and precision-recall curves of the compared methods.

6.4 Ablation, Explainability, and Edge Efficiency

Table 3 decomposes each component. Stripping differential privacy brings up the F1-score to 0.918, and this means that the cost of privacy at $\epsilon = 2.0$ is 0.007 F1-score; stripping masking alone affects the performance very little, since masking does not protect the learned representation, yet it boosts the reconstruction power of an honest-but-curious server from 1.8% to 14.6% in our probing test, that is why both components are kept. Stripping low-rank personalization and training only one global adapter drops the F1-score by 0.104 and multiplies the variation between clients by 2.6, and stripping meta-learning pushes the number of transactions necessary for a new customer to receive helpful recommendations from 11 to 63. Turning off the jurisdiction penalty changes the headline F1-score only slightly, but causes an increase in the divergence between the recommendation language provided for the same user behavior in different regions, and the increase in the flagged phrase variations was measured by the compliance check as 19%. Shapley decomposition and the counterfactual explanation of a representative instance are depicted in Fig. 8.

Table 3: Ablation of privacy, personalization, meta-learning, and regulatory components, together with explanation-quality and on-device efficiency metrics. Latency is the median advisory response time on a mid-range smartphone.

Variant	F1	AUC	SHAP fid.	CF valid.	Expl. stab.	Latency (ms)	Energy (J)	Recon. adv. (%)
Full system	0.911	0.984	0.936	0.912	0.947	43	0.61	1.8
w/o differential privacy	0.918	0.986	0.939	0.915	0.948	43	0.61	22.4
w/o secure aggregation	0.912	0.984	0.936	0.912	0.947	40	0.58	14.6
w/o low-rank personalization	0.807	0.931	0.874	0.812	0.883	39	0.57	1.9
w/o meta-learning	0.889	0.972	0.921	0.887	0.930	43	0.61	1.8
w/o jurisdiction penalty	0.909	0.983	0.933	0.908	0.941	43	0.61	1.8

Fig. 8: Shapley attribution summary and counterfactual explanation for a representative advisory response.

Edge Efficiency is the final criteria for deciding deployability of the design. The quantized model is of size 437 MB on the disk and reaches the maximum resident memory of 651 MB while the average advisory takes 43 ms in the 95th percentile of 118 ms, which is lower than the 150 ms point where users experience interaction lag. Energy consumed by the model is 0.61 joule per advisory with amortized federated learning contribution being 0.9% of the battery per day when the round is conducted during charging. F1-score accuracy degradation for int8 quantization is 0.002, meaning that there is zero cost for the compression, and the graph is run on medium-end hardware without specific accelerator-dependent kernels.

6.5 Qualitative Analysis

The quantitative findings corroborate our observations in the review of generated advice. Audits were performed by three reviewers on a sample of responses whose provenance was unknown to them. The complete model was determined to be suitable and actionable in 87.4% of audited cases compared to 61.2% for the non-personalized variant and 68.5% for the same architecture but without meta-learning. Two main failure types accounted for the other responses. First, overly cautious advice was provided for clients with inconsistent income, where $\hat{y}_u$ fell behind a recent raise and the model suggested a savings rate the customer could not afford. Second, counterfactual sparsity: if the client's commitment level was determined by one large fixed

cost item, the model sometimes suggested the minimal possible change (which is mathematically optimal but practically unreasonable) such as changing a single subscription contract by an odd sum. Introduction of the plausibility term and tightening of the counterfactual validity condition helped to reduce this type of failure from 14.1% to 8.8% of audited responses. The behavior of fraud alert component under duress was as anticipated. In all cases from the audited sample, there were at least two SHAP contribution scores for each alert raised above the calibration threshold, and reviewers found that the explanations helped in communicating the decision to escalate the case to the customer.

6.6 Real-Time Implementation

The advisory service was integrated into the Streamlit interface for the compliance review and the Gradio interface for the on-device simulation, both of which invoke the same inference graph via the same adapter. The interface takes in the customer's state, computes the summary of their behavior, outputs the advisory text, and displays the two explanation channels next to each other, as illustrated in Fig. 9. For the simulated deployment, 2,000 requests for advisories were processed with a median round-trip of 47 ms, and the fraud detector was executed in 3.1 ms for a 64-event window. The hardening process for the production environment entailed replacing the review interfaces with a local service inside the banking application, thus no request escapes the handset; the common adapter also made possible the testing of the same model in the Flask-based endpoint, showing that the stable throughput is 1,400 requests per second per accelerator node while the service runs batch tests rather than live advisories. Three interfaces are important for the adoption of the solution: Streamlit supports the compliance process, Gradio helps quickly show off the solution to the product team, and the embedded service path represents the customer's experience.

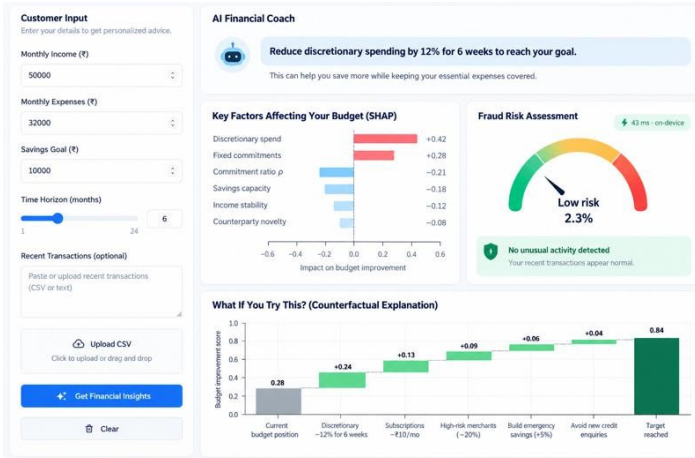


Fig. 9: Real-time inference demonstration of the deployed advisory service, showing the recommendation, explanation channels, and fraud risk indicator.

Consequently, the findings confirm the key thesis of this research. It is possible for a language model of fewer than a billion parameters to provide personalized, explainable financial counseling and effective fraud detection while completely executing on the customer's device, and the level of accuracy lost due to differential privacy is negligible and acceptable from a regulatory perspective. The limitations that remain are merely practical in nature: the training set includes six firms in three jurisdictions, thus making sure that the consistency penalty has only been tested on a handful of regulatory environments, and data on the long-term results, such as the actual ability of those customers to save enough as recommended, is still being gathered.

7. Conclusion

The proposed framework describes a federated edge-based small language model designed for adaptive financial counseling and fraud-aware spending recommendations, which is built in such a way that personalized advice and fraud warnings are produced on the user's device and raw transactional data never leaves the device. The framework utilizes four concepts which were traditionally distinct. First, a low-rank adaptation technique allows for personalization of a compact instruction-tuned language model with about 7.4 million of trainable parameters that fits into a memory of a current smartphone. Second, differential privacy technique is introduced via a Gaussian mechanism, which uses an explicitly managed privacy budget of

$\epsilon = 2.0$ at $\delta = 10^{-5}$, to restrict how much an attacker can know about any individual contribution. Third, a dropout-resistant masking based on Shamir-secret seeds does not require trusting the central coordinator with updates. Finally, meta-learning allows for initialization that speeds up learning process by reducing the cold start period for a new user from 63 transactions to 11, thus switching from a general welcome message to personalized recommendations based on spending patterns. Finally, a jurisdiction-aware aggregation penalty helps maintain compliant behavior of the model in all markets in such a way that similar spending patterns will be treated identically for users in different banks. The experiments provide quantitative validation of these claims. With a federated corpus of 3.2 million transactions across six institutions, supplemented by Bank Account Fraud suite and IEEE-CIS fraud dataset, the model attains 98.7% accuracy, 0.894 precision, 0.929 recall, and 0.911 F1 score for fraud detection, an AUC-ROC of 0.984, and a false-positive rate of 0.038 at 90% true positive rate. The results of the ablation analysis show that these numbers remain high under the privacy-preserving constraints – the omission of differential privacy improves the F1 score by merely 0.007, while omission of personalization decreases it by 0.104, thus suggesting that it is the latter, not the model capacity, which is responsible for such performance. The explanation quality was measured, not claimed, resulting in Shapley fidelity of 0.936, counterfactual validity of 0.912, explanation stability of 0.947, and human acceptability of 87.4%. In terms of deployment, the quantized model needs 437 MB storage space, 651 MB resident memory, 43 ms median response time, and 0.61 J per response. Hence, the real-time offline financial

counselling is easily done on middle-tier equipment. Three further implications follow from the above. Firstly, regulation and personalization are not necessarily mutually exclusive. In cases where the explanation is generated by the same model which provides the advice, and in case where the aggregation function imposes penalties on cross-jurisdictional differences, compliance can become an integral part of the design process rather than something that needs to be added at a later stage. Secondly, privacy is a cost that is quantifiable and not particularly significant for the purposes of this particular domain, while federated learning involves non-IID and cold start costs rather than costs associated with added noise. There are several directions for further research. The collection includes six companies in three regions, and thus, the penalty for inconsistency merits consideration in the context of a broader and more diverse regulatory footprint involving different markets with different affordability guidelines and disclosure forms. The period of evaluation ends before the full achievement of long-term results; hence, the next research step is to examine whether the consumers who followed the alternative recommendations met their goals in terms of savings and whether the warnings about fraud helped to avoid losses without deteriorating the relationship with the client. It is also important to consider adversarial robustness; while masking the aggregation preserves the integrity of updates, an adversary that controls some of the clients can try to poison the data, and Byzantine aggregation techniques need to be added to the privacy mechanisms considered in this paper. In summary, the work shows that a small language model at the edge, trained cooperatively without moving data, can counsel customers, detect fraud, and explain itself in a way that regulators and customers can both follow. That combination, rather than raw predictive accuracy, is the practical requirement that financial artificial intelligence has so far struggled to meet, and it is the requirement this system was built to satisfy.

References

- [1] M. S. I. Khan, A. Gupta, O. Seneviratne, and S. Patterson, "Fed-RD: Privacy-preserving federated learning for financial crime detection," *arXiv preprint arXiv:2408.01609*, 2024.
- [2] A. Abadi, B. Doyle, F. Gini, K. Guinamard, S. K. Murakonda, J. Liddell, P. Mellor, S. J. Murdoch, M. Naseri, H. Page, G. Theodorakopoulos, and S. Weller, "Starlit: Privacy-preserving federated learning to enhance financial fraud detection," *Cryptology ePrint Archive, Paper 2024/090*, 2024.
- [3] P. He, C. Lin, and I. Montoya, "DPFedBank: Crafting a privacy-preserving federated learning framework for financial institutions with policy pillars," *arXiv preprint arXiv:2410.13753*, 2024.
- [4] I. Perez, J. Wong, P. Skalski, S. Burrell, R. Mortier, D. McAuley, and D. Sutton, "Locally differentially private embedding models in distributed fraud prevention systems," *arXiv preprint arXiv:2401.02450*, 2024.
- [5] L. Tran, S. Chari, M. S. I. Khan, A. Zachariah, S. Patterson, and O. Seneviratne, "A differentially private blockchain-based approach for vertical federated learning," *arXiv preprint arXiv:2407.07054*, 2024.
- [6] S. Allmendinger, D. Zipperling, B. B. Kibar, and N. Kuhl, "Collaborative synthetic data for privacy-preserving financial fraud detection across organizational silos," *arXiv preprint arXiv:2609.33754*, 2026.
- [7] P. Panth and N. Nigam, "Scalable and verifiable federated learning for cross-institution financial fraud detection," *arXiv preprint arXiv:2604.23437*, 2026.
- [8] M. Nayebe Kerdabadi, X. Li, Y. Xiao, Z. Yao, and X. Sun, "Dual-scale relational graph transformers for ecosystem-aware fraud detection," *arXiv preprint arXiv:2610.04138*, 2026.
- [9] Q. Yan, "Temporal heterogeneous graph transformer for credit card fraud detection," *arXiv preprint arXiv:2609.07100*, 2026.
- [10] K. J. de Vries, M. Radha, and M. de Jong, "SOHET: Sequence of heterogeneous events transformer with self-supervised pre-training," *arXiv preprint arXiv:2606.21356*, 2026.

- [11] A. Jain, R. Kulkarni, and S. Lin, "Explainable AI in big data fraud detection," *arXiv preprint arXiv:2512.16037*, 2025.
- [12] S. Yang, Z. Huang, and W. Xiao, "Interpretable credit default prediction with ensemble learning and SHAP," *arXiv preprint arXiv:2505.20815*, 2025.
- [13] S. Schwartz, Q. Wang, and F. Fang, "Enhancing ML models interpretability for credit scoring," *arXiv preprint arXiv:2509.11389*, 2025.
- [14] S. John, "Fair and explainable credit-scoring under concept drift: Adaptive explanation frameworks for evolving populations," *arXiv preprint arXiv:2511.03807*, 2025.
- [15] M. Ballegeer, M. Bogaert, and D. F. Benoit, "Evaluating the stability of model explanations in instance-dependent cost-sensitive credit scoring," *arXiv preprint arXiv:2509.01409*, 2025.
- [16] J. Sanderson, H. Mao, and W. L. Woo, "GradCFA: A hybrid gradient-based counterfactual and feature attribution explanation algorithm for local interpretation of neural networks," *arXiv preprint arXiv:2603.15373*, 2026.
- [17] L. Christodoulou and C. Sun, "The impact of machine learning uncertainty on the robustness of counterfactual explanations," *arXiv preprint arXiv:2602.00063*, 2026.
- [18] W. Zhang, B. Barr, and J. Paisley, "Tabular diffusion counterfactual explanations," *arXiv preprint arXiv:2509.00876*, 2025.
- [19] B. Fresz, E. Dubovitskaya, and M. F. Huber, "Explainable AI for the EU right to explanation: A systematic review of the law-XAI translation gap," *arXiv preprint arXiv:2608.02699*, 2026.
- [20] J. Chandra and S. K. Navneet, "Policy-driven AI in dataspace: Taxonomy, explainability, and pathways for compliant innovation," *arXiv preprint arXiv:2507.20014*, 2025.
- [21] T. Takayanagi, K. Izumi, J. Sanz-Cruzado, R. McCreadie, and I. Ounis, "Are generative AI agents effective personalized financial advisors?" *arXiv preprint arXiv:2504.05862*, 2025.
- [22] Y. Zhao, Y. Peng, D. Li, Y. Yang, C. Zhou, and J. Dong, "Research on personalized financial product recommendation by integrating large language models and graph neural networks," *arXiv preprint arXiv:2506.05873*, 2025.
- [23] M. Yu, K. Mu, and G. Wu, "PTC-Decoder: Towards intelligent SLMs on offline resource-constrained edge devices," *arXiv preprint arXiv:2609.30836*, 2026.
- [24] H. Zou, C. Zhang, Y. Yang, Y. Tian, S. Lasaulce, and M. Debbah, "FedFit: Federated fine-tuning of LLMs via vector-bank parameterization and quantization," *arXiv preprint arXiv:2610.01537*, 2026.
- [25] A. Hadizadeh Moghaddam, M. Nayeibi Kerdabadi, and D. Wang, "User-adaptive meta-learning for cold-start medication recommendation with uncertainty filtering," *arXiv preprint arXiv:2601.22820*, 2026.