MartDE: A Privacy-Preserving and Cost-Efficient Evaluation Framework for Data Marketplaces

Xinyuan Qian[1], Haoyong Wang[2], Hangcheng Cao[3], Shuai Yuan[1], Senkang Hu[3], Qingchuan Zhao[3], Hongwei Li[1], Guowen Xu[1]*

[1] School of Computer Science and Engineering, University of Electronic Science and Technology of China
[2] Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences
[3]Department of Computer Science, City University of Hong Kong
{xinyuan.qian, hongweili, guowen.xu}@uestc.edu.cn, wanghaoyong@cigit.ac.cn, {hangccao, cs.qczhao}@cityu.edu.hk, mk2456mk@gmail.com, senkang.forest@my.cityu.edu.hk,

Abstract

The development of machine learning models increasingly relies on high-quality data that resides in private domains. To enable secure and value-driven data exchange under strict privacy regulations, federated learning (FL) has emerged as a key primitive by enabling the trading of model utilities instead of raw data. Among existing solutions, *martFL* (CCS 2023) represents the state-of-the-art FL-based data marketplace architecture, integrating privacy-preserving model evaluation and verifiable trading protocols to enable robust and fair model utility trading without revealing raw data. Despite its strengths, *martFL* suffers from critical weaknesses at the evaluation layer, including plaintext score exposure and unverifiable and manipulable participant selection. To address these challenges, we propose *MartDE*, a dedicated evaluation framework that builds model-centric data marketplaces with robust, privacy-preserving, and verifiable mechanisms. *MartDE* introduces encrypted utility scoring with client-side decryption to preserve score confidentiality, formally bounded anomaly filtering, adaptive participant selection based on global model performance, and commitment-based verification to ensure consistency between declared and evaluated scores and selection verification. We implement *MartDE* and evaluate it across diverse datasets and adversarial conditions. Results show that *MartDE* achieves superior accuracy, robustness, and cost-efficiency, providing a strong foundation for secure and trustworthy utility-driven data marketplaces.

1 Introduction

The rapid progress of AI in areas such as natural language processing, computer vision, and healthcare relies fundamentally on the availability of large-scale, high-quality training data. However, much of this data resides in siloed, private domains held by organizations, making direct access difficult or impossible. As a result, data acquisition has emerged as a critical bottleneck in modern AI development (Sadilek et al. 2021; Lavin et al. 2022; Chang et al. 2023).

Meanwhile, increasing attention to data privacy, through legal frameworks such as GDPR (GDPR 2016) and PIPL (PIPL 2021), has rendered the traditional practice of raw data trading infeasible or illegal. In this new regulatory environment, there is growing consensus that *data utility*, not raw content, should become the principal medium of exchange. This shift has led to the rise of *data marketplaces* that aim to enable secure, privacy-preserving, and value-driven data sharing among mutually untrusted parties.

Federated Learning (FL) (McMahan et al. 2017; Qian et al. 2022, 2024a) naturally aligns with this vision. By allowing participants to collaboratively train models without sharing their original datasets, FL enables each party to contribute *local model updates* as surrogates for data value. These updates can be selectively aggregated to form a global model. As such, FL provides a compelling technical foundation for utility-oriented data exchange.

A representative system in this domain is *martFL* (Li et al. 2023), which introduces a utility-driven, end-to-end data marketplace architecture that enables secure model evaluation and trading without revealing raw data. By integrating quality-aware evaluation and verifiable trading protocols, *martFL* ensures robustness against malicious participants and fairness in reward distribution. The system integrates homomorphic encryption–based model scoring, hierarchical clustering–based anomaly filtering, and zero-knowledge proof techniques to systematically balance privacy preservation and transactional trust in federated data marketplaces.

However, although *martFL* presents a pioneering framework that integrates both model evaluation and verifiable trading in federated data marketplaces, its evaluation mechanism still suffers from several critical limitations in terms of privacy, robustness, and verifiability. First, while encryption techniques are used during score computation, the final utility scores are ultimately decrypted by the Data Acquirer (DA), exposing them to potential misuse in selectively accepting or rejecting Data Providers (DPs), and undermining the confidentiality of model performance. Second, its anomaly detection strategy relies on static, heuristic clustering over utility scores, lacking formal robustness guarantees. The approach is locally optimized, sensitive to distribution shifts, and easily manipulated by malicious participants through carefully crafted model updates. Third, the participant selection process is neither transparent nor verifiable. It is inherently based on clustering heuristics that are non-verifiable and potentially biased, offering no assurance to DPs that their inclusion or exclusion was fair or justified. Collectively, these issues reveal

*Corresponding author.

fundamental deficiencies in the privacy-preserving evaluation pipeline, calling for a more secure, verifiable, and cost-aware protocol to enable trustworthy utility-based data exchange under federated settings.

To address these challenges, we propose *MartDE*, a dedicated framework that reconstructs the model evaluation layer in federated data marketplaces. *MartDE* focuses exclusively on the secure and utility-driven assessment of local model updates, introducing four key innovations. First, it ensures *privacy-preserving score visibility* by enabling each DP to decrypt its own evaluation result locally, preventing the DA from accessing any utility scores in plaintext. Second, it implements a *formally bounded anomaly filtering mechanism* that flags structurally unsound updates by quantizing squared similarity scores and securely filtering out all abnormal values exceeding a predefined threshold via repeated secure maximum operations. Third, it supports *adaptive participant selection* based on a privacy-preserving utility-price ratio, dynamically adjusting the selection proportion in response to global model performance, thereby balancing diversity and efficiency. Fourth, *MartDE* incorporates *commitment-based verification*, binding scores and declared prices to cryptographic commitments. This allows the DA to verify input consistency without compromising score confidentiality for unselected participants, and enables DPs to verify the privacy-preserving selection process. Together, these components form a robust, fair, cost-efficient, and privacy-preserving evaluation module tailored for secure data marketplaces.

In summary, this work presents three key contributions.

1. We design a secure and privacy-preserving model evaluation protocol where utility scores are computed over encrypted data and decrypted only locally by each participant. This ensures full confidentiality of individual model quality while enabling formal anomaly detection based on squared scores.
2. We propose an adaptive participant selection mechanism that jointly leverages *model performance dynamics* and *data quality signals*, allowing the system to adjust selection proportion based on global accuracy while prioritizing participants with high utility-price ratios. A commitment-based verification protocol further ensures that declared scores and prices are consistent with and verifiable against the real ones.
3. We conduct comprehensive experiments on multiple datasets and adversarial settings to evaluate *MartDE*. The results show its strong performance in terms of model accuracy, robustness, and privacy protection compared to existing methods.

2 Preliminaries

2.1 Model-centric Data Marketplaces

Traditional data marketplaces (Song et al. 2021; Qian et al. 2024b) often rely on centralized platforms (e.g., BDEX (BDEX 2014), Quandl (Quandl 2011)) to facilitate raw data exchange, which raises significant legal and privacy concerns under regulations like GDPR (GDPR 2016) and PIPL (PIPL 2021). To address these issues, decentralized and federated paradigms (Li et al. 2023) have emerged, enabling local retention of raw data and exchange of derived utilities (e.g., model updates). This shift has led to *model-centric data marketplaces*, where participants contribute utility-bearing representations without disclosing original data.

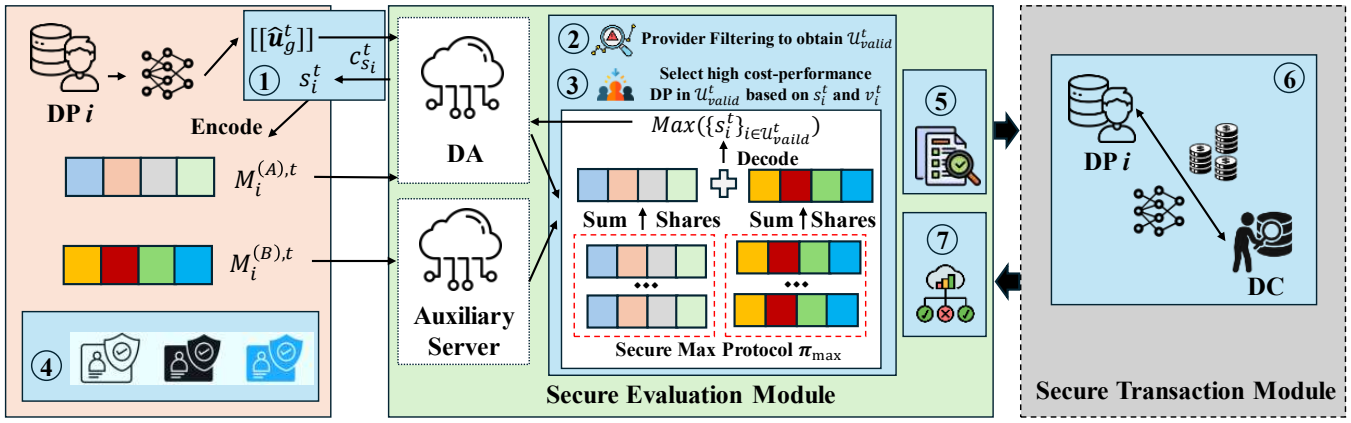
2.2 Threat Model and Assumptions

MartDE mainly focuses on privacy-preserving evaluation other than secure data trading and involves three key roles: *Data Acquirer (DA)*, *Data Providers (DPs)*, and an *auxiliary servers (AS)*. The system operates secure cryptographic to support privacy-preserving model evaluation and behavior verification. To instantiate the non-colluding auxiliary server setting, *MartDE* can adopt infrastructures such as *Divvi Up* (Divvi 2020), a system provided by the Internet Security Research Group (ISRG) (ISRG 2013), known for operating Let's Encrypt (LetsEncrypt 2013). Divvi Up enables privacy-preserving applications that rely on two-party non-collusion assumptions.

Assumptions. We assume that the Data Aggregator (DA) holds a root dataset that is *limited in size and potentially biased*. Unlike many robust federated learning systems that require large, balanced, or trusted data, our design tolerates such limitations, aligning with real-world constraints in low-trust data marketplaces. Additionally, we assume the soundness of standard cryptographic primitives used in our protocol, including homomorphic encryption (e.g., CKKS), commitment schemes, and digital signatures.

Threat Model. Given the above system model, we consider the following adversaries: (1) **Data Providers (DPs): Byzantine and Covert Adversaries.** DPs may submit arbitrary local models and may launch these aforementioned attacks to distrupt the training process, or to earn rewards without actual contributions to training. We also model DPs as *covert adversaries* (Aumann and Lindell 2010)—they behave maliciously only when their misbehavior is unlikely to be detected. (2) **Data Acquirer (DA) and Auxiliary Server (AS): Covert Adversaries under Anytrust.** We adopt the *anytrust* model (Wolinsky et al. 2012), where security holds as long as either DA or AS is honest, and they are assumed not to collude. Both parties may act as *covert adversaries*—deviating from the protocol to gain advantages (e.g., learning private scores, manipulating selection, or biasing aggregation), but only in ways that avoid detection due to potential reputational or legal risks.

Design Goal. This work aims to develop a privacy-preserving, robust, and verifiable model evaluation frFamework tailored for federated data marketplaces. *MartDE* focuses on protecting key privacy-sensitive elements, including each DP's local model updates prior to trading, utility scores during evaluation, and utility-price ratios that reflect economic value. It enables encrypted utility scoring, formal anomaly filtering, and adaptive participant selection under mutual distrust. Notably, secure model trading researches are orthogonal to our research scope.



① Encrypted Model Evaluation; ② Anomaly Filtering; ③ Adaptive Participant Selection; ④ Selection Verification; ⑤ Score Verification; ⑥ Secure Model Trading; ⑦ Normalized Aggregation. DP: Data Provider; DA: Data Acquirer; DC: Data Consumer.

Figure 1: Overview of *MartDE* framework.

3 *MartDE* Framework

3.1 Overview

MartDE is a privacy-preserving evaluation framework tailored for federated data marketplaces, with a core focus on confidentiality, robustness, and verifiability in the evaluation process. As illustrated in Figure 1, the main component of *MartDE* is the *Secure Evaluation Module*, which enables collaborative and private evaluation and selection of model utility. In this module, each DP submits an encrypted utility score $[\![\hat{u}^t]\!]$ and evaluation score s_i^t (Step ①). The DA and an auxiliary server jointly perform anomaly filtering (Step ②) and adaptive participant selection (Step ③), using the *Secure Max Protocol* $\pi_{\max}$ and *Winner Identification Protocol* π_{win} to identify the optimal providers. The selection and evaluation results are then verified through cryptographic commitment schemes and related techniques (Steps ④ and ⑤) to ensure correctness and completeness. The *Secure Transaction Module* shown on the right side of the figure depicts the trading process between DP and DC (Step ⑥), which, although not the focus of this work, is included to present a complete system workflow. Additionally, *MartDE* supports a normalized aggregation mechanism (Step ⑦) to accommodate diverse integration needs. Compared to existing approaches such as *martFL*, *MartDE* achieves comprehensive improvements in privacy protection, robustness against anomalies, and verification capability.

3.2 Framework Details

Encrypted Model Evaluation with Client-side Decryption. To protect individual DPs' privacy and prevent utility score leakage, we design an encrypted evaluation protocol that homomorphically computes cosine similarity between each DP's model update and the DA's baseline. The DA does not see plaintext scores at the begining; each DP decrypts its own result locally to assess model quality and decide whether to participate in aggregation. This enables informed participation while preserving full score confidentiality. Notably, each DP encrypts and uploads its model update, not the utility score. The DA then homomorphically derives encrypted scores using its normalized baseline vector.

Secure Max Protocol $\pi_{\max}$:

Input: n clients each hold private input $x_i \in \{0, 1, \ldots, \ell\}$. Two non-colluding servers $\mathcal{S}_A$ and $\mathcal{S}_B$ aim to compute $m = \max(x_1, \ldots, x_n)$ over a field $\mathbb{Z}_p$ ($p \gg \ell$), without learning x_i.

Output: Only m is revealed; all x_i remain private.

Protocol Steps:

(1) **Client-side:** Each client $\mathcal{C}_i$ encodes x_i as unary vector $\beta_i \in \{0,1\}^\ell$: $\beta_{ij} = 1$ if $j \leq x_i$, else 0. Then sample: $v_{ij} = r_{ij} \sim \mathbb{Z}_p^*$ if $\beta_{ij} = 1$, else 0. Define $M_i = (v_{i1}, \ldots, v_{i\ell})$, split into shares $M_i^{(A)}, M_i^{(B)}$ such that $M_i = M_i^{(A)} + M_i^{(B)}$, and send them to $\mathcal{S}_A$ and $\mathcal{S}_B$ respectively.

(2) **Server-side:** Each server sums received shares: $S^{(A)} = \sum_i M_i^{(A)}$, $S^{(B)} = \sum_i M_i^{(B)}$. They exchange and reconstruct $S = S^{(A)} + S^{(B)} = (s_1, \ldots, s_\ell)$. Then, decode S: For each j, set $\gamma_j = 1$ if $s_j \neq 0$, else 0. Output $m = \max\{j \mid \gamma_j = 1\}$.

Figure 2: Secure Max Protocol $\pi_{\max}$: Privacy-preserving two-server computation of $\max(x_1, \ldots, x_n)$.

The protocol proceeds as follows: (1) *Key Generation:* Each DP is distributed a CKKS key pair $(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow \texttt{KeyGen()}$, sharing pk_i with the DA while keeping sk_i private. (2) *Baseline Encryption:* The DA computes its normalized baseline update vector $\hat{\mathbf{u}}_g^t = \frac{\mathbf{u}_g^t}{\|\mathbf{u}_g^t\|}$, where $\mathbf{u}_g^t = \texttt{Flatten}(W_g^{t'} - W_g^t)$. (3) *Encrypted Evaluation:* Each DP normalizes its local model W_i^t and its local update $\hat{\mathbf{u}}_i^t = \frac{\mathbf{u}_i^t}{\|\mathbf{u}_i^t\|}$, encrypts it as $c_{\mathbf{u}_i}^t = \texttt{Enc}(\mathsf{pk}_i, \hat{\mathbf{u}}_i^t)$, and sends it to the DA. The DA computes encrypted cosine similarity via homomor-

Winner Identification Protocol π_{win}:

Input: n users submit private quantized scores $x_i^t \in \{0, 1, \ldots, \ell\}$, encoded as secret-shared unary vectors M_i. Two non-colluding servers $\mathcal{S}_A$ and $\mathcal{S}_B$ know the maximum $m^t = \max(x_1^t, \ldots, x_n^t)$, but not which user submitted it.

Output: One index i^* such that $x_{i^*}^t = m^t$, with all other x_i^t remaining hidden.

Protocol:

(1) **Seed agreement:** $\mathcal{S}_A$, $\mathcal{S}_B$ decommit to random seeds $\texttt{secret}_A$, $\texttt{secret}_B$ they generated before selection, then compute shared pseudorandom seed: $\texttt{seed} = \texttt{secret}_A \oplus \texttt{secret}_B$.

(2) **Index permutation:** Derive a random permutation π over $[n]$ using PRG initialized with $\texttt{seed}$.

(3) **Winner check:** Iterate over $j = \pi(i)$:

- Reconstruct $M_j = M_j^{(A)} + M_j^{(B)} = (v_{j,1}, \ldots, v_{j,\ell})$.
- If $v_{j,m^t} \neq 0$ (i.e., user j has $x_j^t = m^t$), set $i^* = j$ and terminate.

Figure 3: Winner Identification Protocol π_{win}: Private identification of the user with maximum score.

phic dot product: $c_{s_i}^t = \texttt{EvalMultPlain}(\hat{\mathbf{u}}_g^t, c_{\mathbf{u}_i}^t)$, and returns the result. (4) *Score Decryption:* The DP decrypts its result locally as $s_i^t = \texttt{Dec}(\mathsf{sk}_i, c_{s_i}^t)$, and uses it to decide on participation. Each s_i^t remains private to the DP and is later verified for consistency via a commitment-based mechanism (see Sec. 3.2), which prevents potential manipulation during subsequent filtering or selection.

Anomaly Filtering via Squared Score Screening. To exclude adversarial or numerically unstable updates from aggregation, *MartDE* employs a squared similarity-based anomaly filter. Since each decrypted cosine similarity score satisfies $s_i^t \in [-1, 1]$, its squared value $z_i^t = (s_i^t)^2$ should theoretically lie in $[0, 1]$. However, due to finite-precision errors or malformed ciphertexts (e.g., from poisoned inputs), z_i^t may exceed 1. Therefore, any $z_i^t > 1$ is conservatively treated as an anomaly, and the corresponding update is discarded.

Protocol. Each DP decrypts its score $s_i^t = \texttt{Dec}(sk_i, c_{s_i}^t)$, computes $z_i^t = (s_i^t)^2$, and quantizes it as $x_i^t = \lfloor 10^d \cdot z_i^t \rfloor$. This is encoded as a unary vector $\beta_i^t \in \{0, 1\}^\ell$ with $\beta_{ij}^t = 1$ if $j \leq x_i^t$, and 0 otherwise. The vector is secret-shared to two non-colluding servers $\mathcal{S}_A$ (DA) and $\mathcal{S}_B$. Using the *secure max protocol* (Figure 2), the servers compute $m^t = \max(x_1^t, \ldots, x_n^t)$. If $m^t > x^* = 10^d$, anomaly is detected, and all DPs with $x_i^t > x^*$ are excluded. The remaining valid participants form the set $\mathcal{U}_t^{\text{valid}} = \{i \in [n] \mid x_i^t \leq x^*\}$. It is important to note that the correctness of each reported x_i^t will later be verified through the commitment-based verification phase, ensuring that DPs cannot manipulate their squared scores without detection. Additionally, the *winner identification protocol* (Figure 3) can be directly applied to identify the DP with invalid scores, enabling secure attribution in anomaly filtering.

Adaptive Utility-based Participant Selection. To balance utility and cost, *MartDE* selects participants based on their utility-price ratio $\rho_i^t = s_i^t / v_i^t$. Rather than fixing the number of selected users, we adopt an adaptive selection proportion $p \in (p_{\min}, p_{\max})$ that decreases as model accuracy improves. Given previous epoch accuracy a, the selection rate is:

$$p = p_{\max} - \left(\frac{a - a_{\min}}{a_{\max} - a_{\min}} \right)^{1/q} \cdot (p_{\max} - p_{\min}).$$

Protocol. For each $i \in \mathcal{U}_{\text{valid}}^t$, compute $y_i^t = \lfloor 10^d \cdot \rho_i^t \rfloor$, encode it as a unary vector, and secret-share it to DA ($\mathcal{S}_A$) and an auxiliary server ($\mathcal{S}_B$). The Secure Max Protocol is run iteratively k times to extract the top-k values $m_1^t, \ldots, m_k^t$, each identifying a new user i_r^* with highest remaining score. The final selected set is $\mathcal{U}_{\text{top}}^t = \{i_1^*, \ldots, i_k^*\}$, where $k = \lceil p \cdot |\mathcal{U}_{\text{valid}}^t| \rceil$. The integrity of the submitted values y_i^t and x_i^t is later verified through commitments to prevent misreporting, and the *public verification protocol* (Fig. 4) additionally guarantees the correctness of the maximum-value computation and ensures fairness in the participant selection process.

Remarks. This framework enables *MartDE* to adapt to model performance—broadening participation when accuracy is low, and focusing on efficiency as it improves—while preserving privacy via secure computation and controlling cost through utility-price ranking.

Commitment-based Score Verification. To ensure the correctness of selection, *MartDE* designs a commitment-based verification mechanism before model trading. This enables the DA to validate the evaluation scores and utility-price ratios of selected participants after selection. In round t, each DP i submits to the DA: (1) a commitment to its score s_i^t, denoted as $\text{Com}_i^t = \texttt{Commit}(s_i^t; r_i^t)$; (2) secret-shared, unary-encoded vectors for the quantized squared score $x_i^t = \lfloor 10^d (s_i^t)^2 \rfloor$ and utility-price ratio $y_i^t = \lfloor 10^d (s_i^t / v_i^t) \rfloor$; and (3) a digital signature $\sigma_i^t = \texttt{Sign}_{\text{DP}_i}(v_i^t)$ over the declared price. If selected ($i \in \mathcal{U}_{\text{top}}^t$), the DP discloses the following values only to DA: score s_i^t, declared price v_i^t, and randomness r_i^t. The DA verifies the commitment by checking $\texttt{ComVerify}(\text{Com}_i^t, s_i^t, r_i^t) = \texttt{true}$. Next, the DA verifies the authenticity of the declared price by checking $\texttt{VerifySig}(\text{PK}_i^t, v_i^t, \sigma_i^t) = \texttt{true}$.

Finally, the DA reconstructs the AFE vector M_i^t, infers $\tilde{x}_i^t = \max\{j \mid M_{ij}^t \neq 0\}$, and verifies $\tilde{x}_i^t \stackrel{?}{=} \lfloor 10^d (s_i^t)^2 \rfloor$. To confirm utility-price consistency, the DA reconstructs $\tilde{M}_i^t$, infers $\tilde{y}_i^t = \max\{j \mid \tilde{M}_{ij}^t \neq 0\}$, and verifies $\tilde{y}_i^t \stackrel{?}{=} \lfloor 10^d (s_i^t / v_i^t) \rfloor$. The DPs can additionally verify the selection outcome via protocol π_{ver}.

Secure Model Trading and Normalized Aggregation. Upon completion of the participant selection, the protocol enters the final phase, which securely trades model updates and aggregates them using a normalized weighting scheme.

Public Verification Protocol π_{ver}:
Input: Aggregate vector $T = (t_1, \ldots, t_\ell) \in \mathbb{Z}_p^\ell$, randomness vector $R = (r_1, \ldots, r_\ell) \in \mathbb{Z}_p^\ell$, and each user's commitment vector $\mathbf{C}_i = (c_{i1}, \ldots, c_{i\ell}) \in \mathbb{G}^\ell$, all signed by the corresponding public key.
Output: Confirmation that T correctly aggregates committed values, and that $m = \max\{j \mid t_j \neq 0\}$ is valid.
Steps:

(1) **Signature Check:** Verify digital signature on each $\mathbf{C}_i$ using the user's public key.

(2) **Aggregate Commitments:** For each $j \in [\ell]$, compute $C_j = \prod_{i=1}^{n} c_{ij}$.

(3) **Consistency Check:** Verify $C_j \stackrel{?}{=} g^{t_j} h^{r_j}$, where $r_j = \sum_{i=1}^{n} r_{ij}$, for $j \in [\ell]$.

(4) **Decode MAX:** Let $\gamma_j = 1$ if $t_j \neq 0$, else 0; output $m = \max\{j \mid \gamma_j = 1\}$.

If all checks pass, the result is deemed correct and consistent with committed inputs; no individual input is revealed.

Figure 4: Public verification protocol π_{ver} for validating the computed maximum.

It is worth noting that this work primarily focuses on selecting cost-effective participants through encrypted evaluation and robust filtering, while the actual reward distribution and model update exchange can rely on existing verifiable trading frameworks (e.g., (Li et al. 2023)).

Score Normalization. To enhance stability and fairness in aggregation, we apply a min-max-based normalization to the similarity scores s_i^t. Let $m = \min_{j \in \mathcal{U}_{\text{top}}^t} s_j^t$ and $\tilde{w}_i^t = \frac{s_i^t - m}{\sum_{j \in \mathcal{U}_{\text{top}}^t} s_j^t - n \cdot m + \varepsilon}$, where ε is a small constant (e.g., $\varepsilon = 10^{-6}$) to avoid division by zero. This ensures all weights are non-negative and appropriately scaled.

Weighted Aggregation. The DA aggregates the selected local updates using the normalized weights: $\mathbf{u}_{\text{global}}^t = \sum_{i \in \mathcal{U}_{\text{top}}^t} \tilde{w}_i^t \cdot \mathbf{u}_i^t$. The global model W_g^t is then updated by applying $\mathbf{u}_{\text{global}}^t$. This concludes the training process for the current epoch.

4 Experiment

4.1 Experimental Setup

We conduct experiments on a PC with an NVIDIA GeForce RTX 4060 Ti (8 GB), an Intel® Core™ i9-12900H CPU, and 32 GB RAM. Our code is publicly available[1] for reproducibility and further research.

Datasets and Implementation. We evaluate *MartDE* on three benchmark datasets: MNIST (LeCun 1998), a collection of handwritten digit images; Fashion-MNIST (FMNIST) (Xiao, Rasul, and Vollgraf 2017), which contains clothing and accessory images; and CIFAR-10 (Krizhevsky, Hinton et al. 2009), which is more challenging due to its higher visual complexity. The hyper-parameter settings is the same as *martFL* (Li et al. 2023). Following prior works (Blanchard et al. 2017; Cao et al. 2021), all datasets are partitioned in a non-iid manner with parameter $q = 0.5$, where larger q indicates stronger heterogeneity.

Baselines and Default Setting. We compare our scheme with other three methods in terms of data evaluation performance and efficiency: (1) *Original*, the standard FedAvg (McMahan et al. 2017) without defenses; (2) *martFL* (Li et al. 2023), a robust aggregation framework using cosine similarity and server-side filtering; (3) *DDFed* (Xu et al. 2024), a privacy-preserving scheme based on secure aggregation with HE. To support ciphertext-domain comparison, client scores are scaled (by 10^4) before encryption and rescaled after decryption.

Additionally, we include popular robust aggregation strategies for robustness comparison, including *Krum* (Blanchard et al. 2017), and mean/median-based approaches: *Median*, *Clipping Median*, and *Trimmed Mean* (Yin et al. 2018).

Attack Models and Training Settings. We evaluate defenses against three typical model-poisoning attacks: *Inner Product Manipulation (IPM)* (Fang et al. 2020), *Scaling* (Bagdasaryan et al. 2020), and *A Little Is Enough (ALIE)* (Baruch, Baruch, and Goldberg 2019). Unless otherwise stated, the *attacker ratio is 0.2* (i.e., 20% malicious clients) starting from the 50th round. Evaluation and training involves 10 randomly selected within 100 clients per round, local batch size 64, 3 local epochs per training round, and SGD with momentum 0.9 and learning rate 0.01.

4.2 Evaluation Performance

Overall Performance and Cost Efficiency under Adversarial Settings. We evaluate *MartDE* framework under five random seeds, with 10 participants per aggregation round including 2 adversarial clients (20%). The privacy-preserving data evaluation (PPDE) module is profiled over 50 runs, and the average runtime is reported. To ensure realistic cost settings, each participant's declared price v_i^t is randomly sampled from $[1, 2]$ during all experiments. Table 1 summarizes the average performance across three datasets (MNIST, FMNIST, CIFAR10), comparing *MartDE* with baseline methods including *martFL* and *DDfed*.

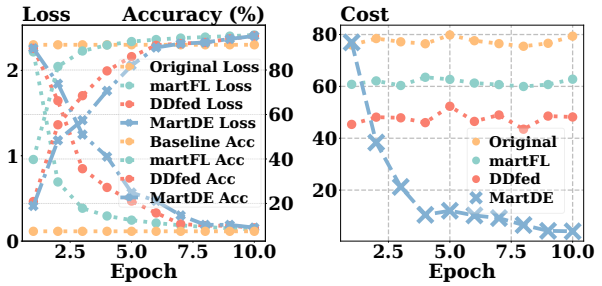
MartDE consistently achieves competitive or superior model performance (in terms of Accuracy and F1-score) compared to existing approaches, while significantly outperforming them in terms of cost saving. For example, on the FMNIST dataset, *MartDE* achieves 77.07% accuracy with a 70% cost saving, whereas *DDfed* achieves a similar accuracy (76.97%) but only 41% cost saving.

This demonstrates that *MartDE*'s utility-price-driven participant selection is more cost-effective. The key reason lies in our *adaptive participant selection mechanism*, which adjusts the number of selected users based on model performance in each epoch. In contrast, *DDfed* employs a fixed selection strategy, which may introduce redundancy or fail to adequately include useful participants under performance variation. On the more challenging CIFAR10 dataset, *MartDE* improves accuracy by 1.5% over *DDfed* and reduces the cost by over

[1] https://github.com/cytus-kira/martDE

Dataset	Method	Acc (%)	F1 (%)	Cost Saving (%)
MNIST	*Original*	10.00 ± 0.08	10.00 ± 0.05	0.0
	martFL	95.76 ± 0.08	95.72 ± 0.05	21 ± 0.5
	DDfed	95.21 ± 0.18	95.25 ± 0.15	41 ± 0.2
	Ours	95.31 ± 0.14	95.27 ± 0.16	**72 ± 0.5**
FMNIST	*Original*	10.00	0.00	0.0
	martFL	23.76	19.66	21± 0.5
	DDfed	76.97 ± 0.28	75.85 ± 0.15	41 ± 0.4
	Ours	77.07 ± 0.58	75.85 ± 0.75	**73 ± 0.5**
CIFAR10	*Original*	10.00	10.00	0.0
	martFL	10.00	1.82	21 ± 0.5
	DDfed	70.16 ± 0.11	70.06 ± 0.28	39 ± 0.4
	Ours	71.67 ± 0.24	71.73 ± 0.38	**74 ± 0.6**

Table 1: Robustness and Cost Efficiency Comparison under Malicious Participant Scenarios.



(a) Acc. & Loss (b) Propotion of Cost Saving

Figure 5: Convergence Performance and Cost Efficiency over 10 Training Epochs.

40%, confirming the adaptability and economic advantage of our design in adversarial settings.

Epoch-wise Convergence and Cost Dynamics Analysis. To further evaluate the temporal performance of our *MartDE* framework, we analyze its convergence behavior and cost-saving dynamics over 10 training epochs, as illustrated in Figure 5. In this set of experiments, involves 50 randomly selected within 100 clients per round, among which 20% (i.e., 10 users) are designated as adversarial participants.

Figure 5a shows the evolution of accuracy and loss across epochs. *MartDE* leverages dynamic client selection, resulting in slightly slower mid-stage convergence because each round incorporates fewer effective updates. Nevertheless, it ultimately achieves accuracy and loss comparable to *MartFL* and *DDFed*, while maintaining robust training dynamics under reduced participation. In contrast, the *Original* setting, which indiscriminately aggregates all updates, fails to converge due to the overwhelming influence of adversarial clients.

Figure 5b illustrates the proportion of cost saving achieved by each method over time. Our approach consistently maintains the highest cost-saving rate, starting above 75% and stabilizing around 70% by epoch 10. This performance is attributed to our adaptive selection mechanism, which dynamically reduces redundancy as the model improves, selecting only the most cost-efficient updates. In contrast, *martFL* uses a clustering-based selection method and therefore incurs relatively stable costs, while *DDfed* applies a fixed top-k selection strategy, which may include redundant participants especially when model performance varies.

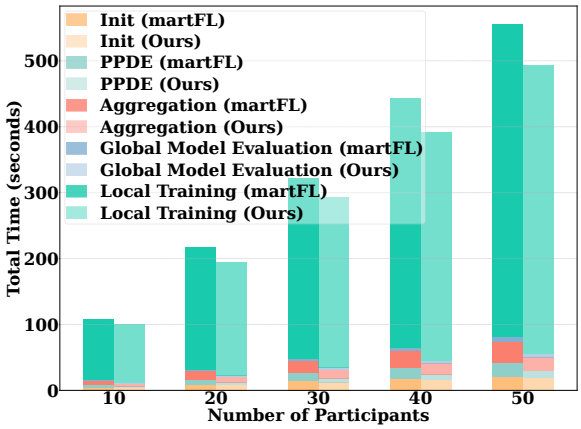


Figure 6: Stage-wise computation time comparison between *MartDE* and *martFL* under varying numbers of participants (10 to 50). Each bar represents the average runtime across five system components.

4.3 Efficiency

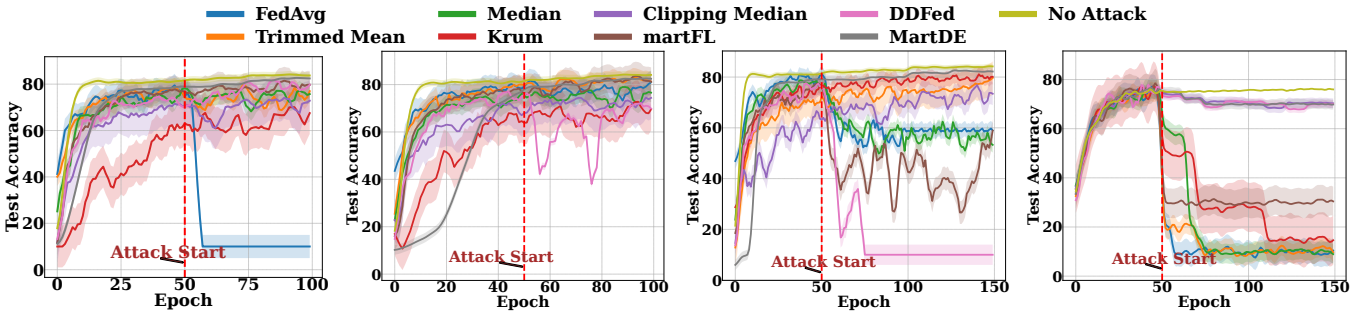
To assess system efficiency, we compare the per-epoch runtime of *MartDE* and *martFL* across aggregation scales (10–50 clients) using a fine-grained breakdown of execution stages. As shown in Figure 6, local training dominates total runtime (>80%) under all configurations, confirming that computation is primarily client-side rather than server-side.

Importantly, *MartDE* demonstrates significant advantages over *martFL* in both the *privacy-preserving evaluation* and *aggregation* stages. The red bars in the figure (representing martFL's encrypted evaluation process) are visibly taller, indicating higher computational complexity due to heavier homomorphic encryption workloads. In contrast, *MartDE* leverages a lightweight encrypted scoring scheme and a two-server-based adaptive participant selection protocol. This design greatly reduces redundant evaluations on low-utility or adversarial updates, allowing the system to select more cost-effective contributions. As a result, both the computational burden on the server and the overall cost of model training are substantially reduced.

Across all participant scales, *MartDE* improves server-side computational efficiency by approximately 1.50× over *martFL*. Specifically, in the PPDE module, the efficiency gain reaches about 1.58×. These improvements highlight the enhanced throughput and scalability of our system. Notably, these gains are achieved without revealing the actual model scores or utility-price ratios, reinforcing the practicality and deployability of *MartDE* in real-world federated learning and data marketplace scenarios.

4.4 Robustness Against Various Attacks

Defense Effectiveness of *MartDE* under Diverse Attack Settings. Figure 7 compares the robustness of various de-



(a) IPM Attack (b) ALIE Attack (c) Scaling $\ll 1$ (d) Scaling $\gg 1$

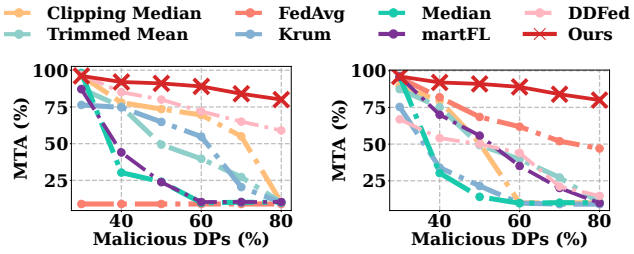
Figure 7: Commparison of robustness across various robust approaches, evaluated on FMNIST, under IPM attack, ALIE attack, and Scaling attacks.

fense strategies against model-poisoning attacks on the FMNIST dataset, including IPM, ALIE, and **two Scaling attacks**. Scaling with a factor greater than 1 amplifies malicious updates and misleads direction-based defenses, while a factor less than 1 attenuates updates and slows convergence. For consistency, the scaling factor is set to the number of participants count with a small random offset. Each experiment involves 100 clients with an attacker ratio of 0.2, and attacks begin at the 50th training round. Under the IPM attack (Figure 7a), *MartDE* quickly recovers and maintains stable performance, achieving post-attack accuracy comparable to the clean training scenario. While traditional defenses such as *Trimmed Mean*, *Clipping Median*, and *martFL* partially mitigate the degradation, their performance remains consistently lower than our method. In contrast, *FedAvg* fails entirely after the attack. For the ALIE attack (Figure 7b, our method again demonstrates strong resilience, preserving over 75% test accuracy with minimal fluctuation. Notably, *MartDE* outperforms *DDFed* and Median-based methods, which exhibit significant accuracy drops after attack initiation.

Figures 7c and 7d show the defense performance under SCALING attacks with different magnitudes. When the scaling factor is small ($\ll 1$), most defenses experience a noticeable decline. However, *MartDE* remains robust, outperforming other methods across nearly the entire training process. When the scaling factor is large ($\gg 1$), the attack severely disrupts baseline defenses, reducing accuracy to below 20%. Notably, only *MartDE* and Clipping Median remain robust, and *MartDE* consistently outperforms.

Resilience of *MartDE* Against Scaling Attacks Targeting Cosine-Similarity Defenses. Figure 8 presents the post-convergence model accuracy of various defense strategies under increasing proportions of malicious clients. All experiments are conducted with 10 participants for 10 epochs, and the reported accuracy corresponds to the final round after convergence.

This experiment specifically evaluates resistance against the scaling attack, which is designed to break cosine-similarity-based defense mechanisms by amplifying or shrinking the vector norm while preserving direction. Subfigure 8a simulates over-scaling (scaling > 1) by setting the scaling factor to the number of clients (10), while Subfigure 8b



(a) scaling>1 (b) scaling<1

Figure 8: Model accuracy under different adversary proportions after 10 epoachs training.

models under-scaling (scaling < 1) using the reciprocal of client count and a random offset.

While cosine-based defenses such as *DDFed* and *martFL* degrade rapidly as the malicious proportion increases, *MartDE* remains consistently robust across all scenarios. Specifically, when the attacker ratio exceeds 60%, most defenses collapse in the over-scaling case, yet *MartDE* still maintains over 75% accuracy at 80% malicious clients. Similarly, under under-scaling, *MartDE* significantly outperforms all baselines across the full range.

The key reason lies in our gradient norm normalization module, which re-scales all updates to a fixed norm prior to aggregation. This step eliminates the norm-based exploitability inherent to cosine similarity, enabling *MartDE* to benefit from similarity-based evaluation without being vulnerable to direction-preserving norm attacks.

5 Conclusion

We presented *MartDE*, a privacy-preserving and utility-aware evaluation framework for data marketplaces. By combining encrypted scoring, secure anomaly filtering, adaptive participant selection, and commitment-based verification, *MartDE* enhances privacy, robustness, cost-efficiency, verifiability, and fairness in model update evaluation. Future work will integrate lightweight secure computation to enable efficient, verifiable data trading and support trustworthy data marketplaces.

Acknowledgements

This work is supported by the Sichuan Science and Technology Program under Grant 2024ZHCG0188.

References

Aumann, Y.; and Lindell, Y. 2010. Security against covert adversaries: Efficient protocols for realistic adversaries. *Journal of Cryptology*, 23(2): 281–343.

Bagdasaryan, E.; Veit, A.; Hua, Y.; Estrin, D.; and Shmatikov, V. 2020. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, 2938–2948. PMLR.

Baruch, G.; Baruch, M.; and Goldberg, Y. 2019. A little is enough: Circumventing defenses for distributed learning. *Advances in Neural Information Processing Systems*, 32.

BDEX. 2014. BDEX: The First Decentralized Data Exchange Platform. https://www.bdex.com. Accessed: July 2025.

Blanchard, P.; El Mhamdi, E. M.; Guerraoui, R.; and Stainer, J. 2017. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30.

Cao, X.; Fang, M.; Liu, J.; and Gong, N. 2021. FLTrust: Byzantine-robust Federated Learning via Trust Bootstrapping. In *Proceedings of NDSS*.

Chang, Q.; Yan, Z.; Zhou, M.; Qu, H.; He, X.; Zhang, H.; Baskaran, L.; Al'Aref, S.; Li, H.; Zhang, S.; et al. 2023. Mining multi-center heterogeneous medical data with distributed synthetic learning. *Nature communications*, 14(1): 5510.

Divvi. 2020. Divvi Up: Privacy-Preserving Data Aggregation Service. https://divviup.org. Accessed: 2025-07-28.

Fang, M.; Cao, X.; Jia, J.; and Gong, N. Z. 2020. Local model poisoning attacks to Byzantine-robust federated learning. In *USENIX Security Symposium*.

GDPR. 2016. General Data Protection Regulation (GDPR). https://eur-lex.europa.eu/eli/reg/2016/679/oj. EU Regulation 2016/679, Accessed: July 2025.

ISRG. 2013. Internet Security Research Group. https://www.abetterinternet.org/. Accessed: 2025-07-28.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.(2009).

Lavin, A.; Gilligan-Lee, C. M.; Visnjic, A.; Ganju, S.; Newman, D.; Ganguly, S.; Lange, D.; Baydin, A. G.; Sharma, A.; Gibson, A.; et al. 2022. Technology readiness levels for machine learning systems. *Nature Communications*, 13(1): 6039.

LeCun, Y. 1998. The MNIST Database of Handwritten Digits. http://yann.lecun.com/exdb/mnist/. Accessed: 2025-07-29.

LetsEncrypt. 2013. Let's Encrypt - Free SSL/TLS Certificates. https://letsencrypt.org/. Accessed: 2025-07-28.

Li, Q.; Liu, Z.; Li, Q.; and Xu, K. 2023. martFL: Enabling Utility-Driven Data Marketplace with a Robust and Verifiable Federated Learning Architecture. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, CCS '23, 1496–1510. New York, NY, USA: Association for Computing Machinery. ISBN 9798400700507.

McMahan, H. B.; Moore, E.; Ramage, D.; Hampson, S.; and Aguera y Arcas, B. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54 of *Proceedings of Machine Learning Research*, 1273–1282. PMLR.

PIPL. 2021. Personal Information Protection Law of the People's Republic of China (PIPL). https://www.gov.cn/xinwen/2021-08/20/content_5632486.htm. Adopted by the Standing Committee of the National People's Congress, Effective Nov. 1, 2021. Accessed: July 2025.

Qian, X.; Li, H.; Hao, M.; Xu, G.; Wang, H.; and Fang, Y. 2024a. Decentralized multi-client functional encryption for inner product with applications to federated learning. *IEEE Transactions on Dependable and Secure Computing*, 21(6): 5781–5796.

Qian, X.; Li, H.; Hao, M.; Yuan, S.; Zhang, X.; and Guo, S. 2022. CryptoFE: Practical and privacy-preserving federated learning via functional encryption. In *GLOBECOM 2022-2022 IEEE Global Communications Conference*, 2999–3004. IEEE.

Qian, X.; Li, H.; Xu, G.; Wang, H.; Zhang, T.; Chen, X.; and Fang, Y. 2024b. Privacy-preserving data evaluation via functional encryption, revisited. In *IEEE INFOCOM 2024-IEEE Conference on Computer Communications*, 11–20. IEEE.

Quandl. 2011. Quandl: Financial, Economic and Alternative Data. https://www.quandl.com. Accessed: July 2025.

Sadilek, A.; Liu, L.; Nguyen, D.; Kamruzzaman, M.; Serghiou, S.; Rader, B.; Ingerman, A.; Mellem, S.; Kairouz, P.; Nsoesie, E. O.; et al. 2021. Privacy-first health research with federated learning. *NPJ digital medicine*, 4(1): 132.

Song, Q.; Cao, J.; Sun, K.; Li, Q.; and Xu, K. 2021. Try before you buy: Privacy-preserving data evaluation on cloud-based machine learning data marketplace. In *Proceedings of the 37th Annual Computer Security Applications Conference*, 260–272.

Wolinsky, D. I.; Corrigan-Gibbs, H.; Ford, B.; and Johnson, A. 2012. Scalable anonymous group communication in the anytrust model. *European Workshop on System Security (EuroSec)*.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms.

Xu, R.; Gao, S.; Li, C.; Joshi, J.; and Li, J. 2024. Dual defense: Enhancing privacy and mitigating poisoning attacks in federated learning. *Advances in Neural Information Processing Systems*, 37: 70476–70498.

Yin, D.; Chen, Y.; Kannan, R.; and Bartlett, P. 2018. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International conference on machine learning*, 5650–5659. Pmlr.