# INCENTIVIZING INCLUSIVE DATA CONTRIBUTIONS IN PERSONALIZED FEDERATED LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

While data plays a crucial role in training contemporary AI models, it is acknowledged that valuable public data will be exhausted in a few years, directing the world's attention towards the massive decentralized private data. However, the privacy-sensitive nature of raw data and lack of incentive mechanism prevent these valuable data from being fully exploited. Addressing these challenges, this paper proposes inclusive and incentivized personalized federated learning (iPFL), which incentivizes data holders with diverse purposes to collaboratively train personalized models without revealing raw data. iPFL constructs a model-sharing market by solving a graph-based training optimization and incorporates an incentive mechanism based on game theory principles. Theoretical analysis shows that iPFL adheres to two key incentive properties: individual rationality and truthfulness. Empirical studies on eleven AI tasks (e.g., large language models' instruction-following tasks) demonstrate that iPFL consistently achieves the highest economic utility, and better or comparable model performance compared to baseline methods. We anticipate that our iPFL can serve as a valuable technique for boosting future AI models on decentralized private data while making everyone satisfied.

## 1 INTRODUCTION

Training on massive publicly-available data (Raffel et al., 2020; Gao et al., 2020; Schuhmann et al., 2021; 2022), AI models have demonstrated significant proficiency in diverse domains (Brown et al., 2020; Ouyang et al., 2022b; Rombach et al., 2022; Ramesh et al., 2021). As a well-known representative, ChatGPT (Ouyang et al., 2022b; OpenAI, 2023) has swept the world with its exceptional ability to solve general tasks. While it is commonly acknowledged that more data leads to better performance (Kaplan et al., 2020), it has been estimated that available and valuable data in public will be exhausted by the year 2026 (Villalobos et al., 2022; Muennighoff et al., 2023), significantly impeding the continued enhancement of AI models under the current training paradigm.

The gradual depletion of public data starkly contrasts with the private sector, where massive institutions separately hold a wealth of valuable data. For instance, financial institutions such as Bloomberg (Wu et al., 2023) possess high-quality private data to train AI models for finance. Ideally, if these institutions collaborate on their resources, they can create a substantial and diverse database capable of augmenting contemporary AI models (Wu et al., 2023; Singhal et al., 2023; Wang et al., 2023; Chen et al., 2023). Unfortunately, two critical practical issues prevent distributed private data from being fully exploited (Voigt & Von dem Bussche, 2017; Kairouz et al., 2021). Firstly, the sensitivity of private data deters institutions from sharing it readily since this could raise privacy concerns and cause interest conflict (Voigt & Von dem Bussche, 2017; Price & Cohen, 2019; Hathaliya & Tanwar, 2020; Box & Pottas, 2013; Qi et al., 2023; Kaissis et al., 2021). Secondly, the absence of a comprehensive incentive mechanism results in a lack of motivation for institutions to actively and willingly engage in collaboration (Yang et al., 2019; Karimireddy et al., 2022).

Consequently, to enable the utilization of decentralized private data for the continued enhancement of contemporary AI models, it is imperative to establish a harmonious sharing market, which should safeguard privacy and ensure individual interests. In this market, data owners could act as buyers who selectively buy models from others to help train stronger models for their interested tasks; or as sellers who gain revenues from other institutions that have bought their models. Such a guarantee of privacy (i.e., trading models rather than data) and interests can well motivate institutions to partici-

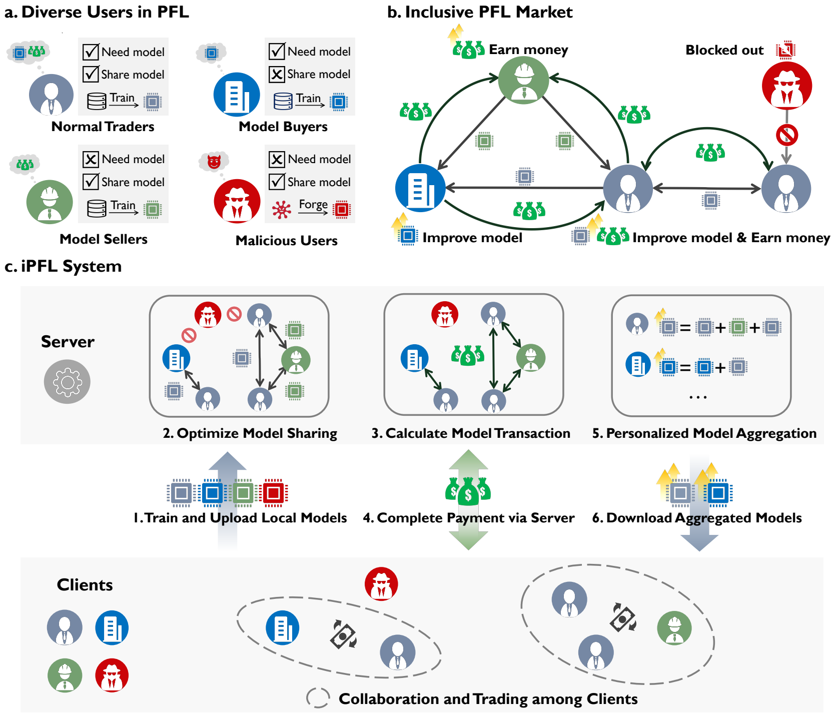


Figure 1: Inclusive PFL market and our iPFL. **a.** The clients have different purposes for entering a PFL system. A client can be: i) a trader who simultaneously buys model and sells their model; ii) a buyer who only buys a model and never shares its own model; iii) a seller who only sells its own model and never buys models; iv) an attacker who intends to ruin the system. **b.** In an inclusive market system, the model and money transaction should satisfy the needs of all the participants and block out attackers. **c.** In our iPFL, all the market behaviors are completed over a neutral server.

pate in the market, forming a virtuous circle as more participants lead to better performance which in turn attracts more participants.

Following this vision, we adopt personalized federated learning (PFL) (Wu et al., 2022; T Dinh et al., 2020; Fallah et al., 2020) as the technical foundation for model training in this market, due to PFL's properties on preserving data privacy (i.e., sharing models) and catering to personal interests (i.e., improving personalization performance). In this PFL-based market, coordinated by a central server, participants share their locally-trained models to achieve personalization through collaboration (Ye et al., 2023b; Li et al., 2021a; Huang et al., 2021). This approach has shown promising personalized performance through techniques like model regularization (Li et al., 2021a), meta-learning (Fallah et al., 2020), and clustering (Sattler et al., 2020). However, existing methods mainly focus on personalization techniques, overlooking participants' economic conditions and motivations, which are two key factors in market dynamics.

Therefore, in this paper, we introduce an inclusive PFL system that accommodates individual model preferences and economic conditions, where we specifically consider four types of participants as shown in Figure 1 (a). We model the overall system as a graphical game, with participants as nodes and their exchange relationships as asymmetrically weighted edges, enabling a nuanced model-sharing network; see illustration in Figure 1 (b). To achieve this, we propose a novel graph-based PFL optimization objective that captures an individual's model preference via model similarity and economic conditions via reserving personalized utility functions. Specifically, we pursue personalized models by minimizing loss on interested tasks while maximizing the pair-wise model similarity among participants and the total social welfare within the overall collaboration graph. In this way, participants are allowed to select models based on their preferences and affordability, improving personalization performance, enhancing system robustness against inauthentic models and promoting cost efficiency.

While the graph-based PFL provides the technical foundation, the market's success also depends on an effective incentive mechanism to motivate participation. This mechanism must fairly reward contributions and ensure those benefiting from contributions compensate accordingly, while also promoting honest participation and deterring dishonest or malicious behavior (Li et al., 2021b; Zhan et al., 2020). To achieve this, we design a payment mechanism in our PFL system (we term our overall system iPFL where i denotes incentivized and inclusive) that encourages willing and honest participation. This mechanism sets specific prices for model transactions, calculated using game theory principles and considering both the buyer's economic utility and the seller's model quality. This ensures mutual benefit from each transaction. Through theoretical analysis, we show that iPFL adheres to two key incentive principles: individual rationality, ensuring that all participants benefit from each training round, and truthfulness, incentivizing clients to disclose their true training costs, fostering a collaborative and honest market environment.

To verify the effectiveness of our proposed iPFL (see system overview in Figure 1), we conduct extensive experiments, covering comprehensive comparisons with baselines, diverse scenarios and tasks. Results show that iPFL consistently achieves higher economic utility, and better or comparable personalization performance compared to state-of-the-art PFL methods. Remarkably, in a scenario of training large language models (Touvron et al., 2023), iPFL can achieve $49\%$ higher economic utility and $9\%$ higher model utility than the best baseline method. We anticipate that our proposed iPFL can serve as a valuable technique for boosting future AI models on decentralized private data while making everyone satisfied.

## 2 Related Works

**Personalized federated learning.** The latest developments in personalized federated learning (PFL) have made significant strides in enhancing individual performance. Present approach to attain personalized model for the clients in FL includes model regularization (Li et al., 2021a; T Dinh et al., 2020), meta-learning (Fallah et al., 2020), clustering (Sattler et al., 2020) techniques, and more.

Particularly, our work is related to the PFL methods that incorporate graph regularization (Huang et al., 2021; Ye et al., 2023b; Zhang et al., 2021). Huang et al. (2021) proposes FedAMP, which introduces regularization with a predefined attention-inducing function capturing the pairwise collaboration among the clients. In pFedGraph proposed by Ye et al. (2023b), a collaboration graph is learned from clients' model similarity and then used to regularize the update of personalized models. These methods are more interpretable as they explicitly describe the collaboration relationships among the clients, which are suitable for the basis of a market system. The collaboration topology in previous performance-prior graph-based PFL methods is usually determined by model attributes. In our works, we additionally introduce a market system into the formulation of the collaboration graph, so that our PFL system can balance performance and economy simultaneously.

We defer literature review on **incentivized federated learning** in Section E.4.

## 3 Problem Formulation

We consider the popular PFL settings, where $m$ institutions join the system as clients and are managed by a central server. Each client $i$ holds a private dataset $\mathbf{Z}_i = (\mathbf{z}_{i,1}, ..., \mathbf{z}_{i,N_i})$ with $N_i$ data points sampled from client $i$'s local data distribution $\mathcal{D}_i$. Each client $i$ maintains its own model parameters $\theta_i$. Given a common loss criterion $l(\cdot, \cdot)$, the empirical loss of client $i$ on its own dataset $\mathbf{Z}_i$ is: $L_i(\theta_i) = \frac{1}{N_i}\sum_{k\in[N_i]} l(\theta_i; \mathbf{z}_{i,k})$. The clients hope to train a personalized model that performs well on its local data distribution $\mathcal{D}_i$. In this case, the population loss (testing loss) for client $i$ is $L_i^*(\theta_i) = \mathbb{E}_{\mathbf{z}\sim\mathcal{D}_i} l(\theta_i; \mathbf{z})$.

Due to privacy concerns and communication constraints in multi-institutional scenarios, the clients cannot directly send their data to other clients. In our work, we consider the model-sharing strategy in PFL. As a member of the federation, each client can refer to others' model parameters, coordinated by the server and realized at the server side. Specifically, to describe the model sharing to topology among the clients, we use a directed graph represented by the adjacency matrix $\mathbf{A} = (a_{ij})_{m\times m}$

with:

$$a_{ij} = \begin{cases} 1 & \text{if } i \neq j \text{ and client } i \text{ imports the model of client } j \\ 0 & \text{else} \end{cases}$$

Each client may participate in federated learning with different purposes (i.e., for a better model for their local tasks or economic rewards). Meanwhile, we assume that the server has no interest in clients' tasks and only acts as a neutral coordinator. With the model sharing topology $\mathbf{A}$, we then use a graphical game model to formulate the market of PFL. The players in this game are the clients, without the server. In the market, the clients can choose the models they will import, so the action of client $i$ is described by $\mathbf{a}_i$ (the $i$th roll of $\mathbf{A}$). The action of model sharing can reflect the fairness of the training. For example, a client may share its own model with many other clients, but import few models from them. In this case, this client may not be satisfied with the arrangement of training, as it cannot obtain proportionate treatment from the federation. Therefore, we additionally introduce a utility function to capture the clients' non-training benefits (i.e., economic gain, satisfaction with the collaboration) in the procedure of training.

**Definition 1** (utility)**.** Consider that the clients are sharing models for multiple rounds (in each round $t$ the clients share models according to $\mathbf{A}^t$). The utility of client $i$ in each round is defined as:

$$U_i^t = G_i(\mathbf{a}_i^t) - \sum_{j \in [m]} a_{ji}^t c_i - p_i^t. \tag{1}$$

We elaborate on the three components as follows:

**Definition 2** (collaboration gain)**.** $G_i : \{0,1\}^m \to \mathbb{R}$ is the gain function for client $i$, with $G_i(\mathbf{a}_i^t)$ describing client $i$'s gain of data resource from chosen collaborators $\mathbf{a}_i^t$.

In our work, we consider a specified $G_i$ to describe the diminishing collaboration gain for the quality-aware clients. We assume that for client $i$ the gain is related to the amount of imported data, which can be represented by a continuous and concave function $g_i : \mathbb{R} \to \mathbb{R}$. Specifically, we consider:

$$G_i(\mathbf{a}_i) = g_i(\sum_{j \in [m]} a_{ij} N_j) = \sqrt{\frac{K_i}{N_i}} - \sqrt{\frac{K_i}{N_i + \sum_{j \in [m]} a_{ij} N_j}}, \tag{2}$$

with hyper-parameter $K_i$ representing client $i$'s level of eagerness for data. $K_i$ can be varied among the clients , reflecting their different data needs. If $K_i = 0$, that means the client cannot benefit from enlarged data access, so he may be a pure model seller and will not buy models from any other client. If $K_i > 0$, getting more data from the collaborators will increase the gain, but the marginal benefit brought by each collaborator will become smaller. If there is no collaborator, the gain is zero, as $G_i(\mathbf{0}) = g_i(0) = 0$.

**Definition 3** (sharing cost)**.** If $\theta_i^t$ is imported by one another client, client $i$ will suffer a loss of $c_i$.

$c_i$ indicates client $i$'s unwillingness to share its model due to fairness or privacy concerns. By definition, unless the payback is larger than $c_i$, the change of utility is negative and client $i$ is reluctant to share its model. From this perspective, $c_i$ can also be taken as a minimum price to share the model. So we introduce money transactions to overcome this barrier to collaboration.

**Definition 4** (overall payment)**.** $p_i^t$ represents the amount of payment client $i$ should pay to the system in round $t$.

If $p_i^t > 0$, that means client $i$ has to pay for the benefits gained from the federation; otherwise, that means client $i$ is rewarded for his contribution to the federation. Since the server is neutral, we only consider the monetary transaction among the clients. Denote $r_{ij}^t$ as the remittance from client $i$ to client $j$. Since the money transaction is symmetric: $p_i^t = \sum_{j \in [m]} r_{ij}^t - \sum_{j \in [m]} r_{ji}^t$, there is $\sum_{i \in [m]} p_i^t = 0$.

Based on our evaluation of clients' utility, we have the social welfare of the whole market system.

**Definition 5** (social welfare)**.** Denote $\mathbf{c} = (c_1, ..., c_m)^\top$, the **social welfare** at round $t$ is defined as:

$$\text{SW}(\mathbf{A}^t) = \sum_{i \in [m]} U_i^t = \sum_{i \in [m]} \left[ G_i(\mathbf{a}_i^t) - \mathbf{c}^\top \mathbf{a}_i^t \right]. \tag{3}$$

In iPFL, our goal is to build up an inclusive PFL system that provides personalized training according to clients' models and economic needs. To achieve this, we propose a novel PFL optimization problem, which pursues smaller training loss and larger between-collaborators model similarity, while maintaining the economic utility of all the clients. Define the similarity between two models $\theta_i$ and $\theta_j$ using a differentiable function $d(\theta_i, \theta_j)$. To enable more collaboration among more similar clients, we can include pair-wise similarity in the optimization problem of PFL. Thus, our training objective is defined as:

$$\min_{\forall i \in [m]: \theta_i; \mathbf{A}} \sum_{i \in [m]} L_i(\theta_i) + \lambda \sum_{i,j \in [m]} a_{ij} \frac{N_j}{N_i} d(\theta_i, \theta_j) - \mathrm{SW}(\mathbf{A}). \tag{4}$$

The first two terms $L_i(\theta_i) + \lambda \sum_{i,j \in [m]} a_{ij} \frac{N_j}{N_i} d(\theta_i, \theta_j)$ are model-similarity-aware training loss. With pairwise collaboration indicated by binary indicator $a_{ij}$, if client $i$ imports $j$'s model, a regularization term $\lambda \frac{N_j}{N_i} d(\theta_i, \theta_j)$ will be added to the training loss. The third term $\mathrm{SW}(\mathbf{A})$ is the social welfare under the graph $\mathbf{A}$, which can also be taken as a regularization term to avoid $\mathbf{A}$ degrade to $\mathbf{0}$. Therefore, the clients can attain personalized models without losing generality by minimizing both the loss on local tasks and the model difference compared to their collaborators. At the same time, our system also optimizes social welfare by refining the clients' selection of references, which ensures the benefits of each client and makes the training more economic-efficient.

## 4 Inclusive and Incentivized Personalized Federated Learning

We overview the system in Algorithm 1, which takes $T$ rounds in total to alternatively optimize personalized model $\theta_i$ and neighbor selection $\boldsymbol{A}_i$ for each client and assign appropriate payment among clients. Specifically, in each round $t$, the clients first update and upload their local models $\theta_i^t$. Then, the server calculates the data sharing graph $\boldsymbol{A}^t$ by optimizing the clients' actions according to the game model. The amount of payment $p_i^t$ for each client is also calculated according to $\boldsymbol{A}^t$ of round $t$. At the end of each round, the clients have two choices: 1) pay $p_i^t$ and receive the aggregated model $\bar{\theta}_i^t$ for next round; 2) quit the federation and take the best model in previous rounds as the final model. So the clients can leave the multi-round training at ant time.

The training procedure involves three key steps: local model training at the client side, where personalized models are trained locally; graph topology learning at the server side, where the model sharing topology is learnt from the uploaded local models; payment calculation at the server side, where a bill for each client is determined by the server and the clients complete money transaction by paying the bill.

**Local model training**. To train a personalized model by collective data, each client updates its model parameters locally by simultaneously minimizing loss on local tasks and model-level distance from the selected collaborators' models. Since it is not feasible to optimize all the clients' models at the same time, we update the local models by block gradient descent. That is, in round $t$, each client updates its model by:

$$\theta_i^{t+1} = \arg\min_{\theta_i} L_i(\theta_i) + \lambda \sum_{j \in [m]} A_{ij}^t \frac{n_j}{n_i} d(\theta_i, \theta_j^t), \tag{5}$$

where $A_{ij}^t$ is the collaboration indicator in $\boldsymbol{A}^t$ determined by the server at round $t$. If the server determines that client $i$ should collaborate with client $j$, $A_{ij}$ will be set as 1, so that client $i$ will be encouraged to learn from client $j$ during the local model training by minimizing the distance between local model $\theta_i$ and collaborators' models $\{\theta_j^t\}_{A_{ij}=1}$. However, directly solving Equation (5) requires times of communication cost because client $i$ need access to all collaborators' local models. To efficiently avoid introducing additional communication costs, we propose to apply the proximal gradient descent method, in which the server computes Equation (6) in advance before transmitting information to clients and each client optimizes Equation (7) during local model training in practical implementation:

$$\bar{\theta}_i^t = \theta_i^t - \frac{\eta}{n_i} \sum_{j \in [m]} A_{ij}^t n_j \nabla_{\theta_i^t} d(\theta_i^t, \theta_j^t) \tag{6}$$

$$\theta_i^{t+1} = \arg\min_{\theta_i} L_i(\theta_i) + \frac{\lambda}{2\eta} \|\theta_i - \bar{\theta}_i^t\|_2^2, \tag{7}$$

where $\eta$ is the step size in the calculation of the proximal center. With such a technique, the server only needs to send client $i$ a proximal center $\bar{\theta}_i^t$ at round $t$ instead of multiple models from collaborators.

**Graph topology learning**. The server needs to find a suitable model-sharing graph based on the local models uploaded by clients. The graph is optimized by minimizing the model distance between collaborators and maximizing social welfare of the overall system, which corresponds to solving a sub-problem of Equation (4):

$$\forall i \in [m] : \boldsymbol{A}_i^t = \arg\min_{\boldsymbol{A}_i} \phi_i(\boldsymbol{A}_i) = \lambda \sum_{j\in[m]} A_{ij} \frac{n_j}{n_i} d(\theta_i^t, \theta_j^t) + \mathbf{c}^\top \boldsymbol{A}_i - G_i(\boldsymbol{A}_i) \tag{8}$$

$$\text{s.t. } \forall i, j \in [m] : A_{ij} \in \{0, 1\}.$$

where $\phi_i$ denotes the objective function of the sub-problem for each client. The problem is an NP-hard integer programming and finding an optimal solution can be very costly. Therefore, we propose an efficient graph learning algorithm (Algorithm 2) to get an approximate solution for this optimization problem in $O(m)$ time. In this algorithm, we calculate a threshold data amount $n_k^{Th}$ for each potential collaborator of client $i$ by $g_i(n_j^{Th}) - g_i(n_j^{Th} - n_j) = c_j + \lambda \frac{n_j}{n_i} d(\theta_i^t, \theta_j^t)$. Since the marginal collaboration gain brought by each collaborator of client $i$ decreases with client $i$'s total accessible data amount, if $A_{ij} = 0$ and $n_j^{Th} > n_j + \sum_{k\in[m]} A_{ik} n_k$, then setting $A_{ij} = 1$ would make $\phi_i$ smaller. Therefore, Algorithm 2 keeps adding the client $j$ with the largest $n_j^{Th}$ to the collaborators of client $i$ until $\forall j : A_{ij} = 0 \Rightarrow n_j^{Th} \leq n_j + \sum_{k\in[m]} A_{ik} n_k$. Hence, after Algorithm 2 reaches the condition of termination, we have a solution $\boldsymbol{A}_i^*$ that satisfies:

$$\begin{aligned} \forall j \neq i : A_{ij}^* = 1 \Rightarrow \phi_i(\boldsymbol{A}_i^* - \boldsymbol{E}_j) > \phi_i(\boldsymbol{A}_i^*) \\ \forall j \neq i : A_{ij}^* = 0 \Rightarrow \phi_i(\boldsymbol{A}_i^* + \boldsymbol{E}_j) \geq \phi_i(\boldsymbol{A}_i^*), \end{aligned} \tag{9}$$

where $\boldsymbol{E}_j$ is the $j$th row of the identity matrix. Though this algorithm cannot ensure a globally optimal solution to Equation (8), its solution $\boldsymbol{A}_i^*$ is a locally optimal choice for client $i$, as adding or removing any collaborator will not make the objective $\phi_i$ smaller. By introducing such an approximate solution, our graph learning algorithm can attain a feasible $\boldsymbol{A}^t$ efficiently and robustly without sacrificing a large amount of time searching for unnecessary optimality, which is sufficiently effective in practice.

**Payment calculation**. According to the definition of the utility of clients in Definition 1, if client $i$ imports the model from $j$, it will pose a cost of $c_j$ to client $j$. This indicates that the model sharing is not reciprocal, leading to the dilemma that some clients lack the incentive to join the training. Therefore, after confirming the collaboration graph $\boldsymbol{A}^t$ among clients, the server needs to determine the required payment $p_i^t$ for client $i$, which needs to ensure that contributions from clients are aptly rewarded and those benefiting from these contributions are appropriately charged. In our payment design, we consider the reward calculated based on the benefit brought by the imported model. The payment is defined as follows: if client $i$ imports $j$'s model, client $i$ pays to $j$ the marginal benefit minus the model difference:

$$r_{ij}^t = A_{ij}^t [G_i(\boldsymbol{A}_i^t) - G_i(\boldsymbol{A}_i^t - \boldsymbol{E}_j) - \lambda \frac{n_j}{n_i} d(\theta_i^t, \theta_j^t)],$$

where $G_i(\boldsymbol{A}_i^t) - G_i(\boldsymbol{A}_i^t - \boldsymbol{E}_j)$ is the marginal benefit (change of the gain) brought by $j$'s model and $\lambda \frac{n_j}{n_i} d(\theta_i^t, \theta_j^t)$ is the model difference term defined in Equation (4). In this way, $p_i^t$ can be written as:

$$\begin{aligned} p_i^t &= \sum_{j\in[m]} r_{ij}^t - \sum_{j\in[m]} r_{ji}^t \\ &= \sum_{j:A_{ij}^t=1} \left[ G_i(\boldsymbol{A}_i^t) - G_i(\boldsymbol{A}_i^t - \boldsymbol{E}_j) - \lambda \frac{n_j}{n_i} d(\theta_i^t, \theta_j^t) \right] - \sum_{j:A_{ji}^t=1} \left[ G_j(\boldsymbol{A}_j^t) - G_j(\boldsymbol{A}_j^t - \boldsymbol{E}_i) - \lambda \frac{n_i}{n_j} d(\theta_j^t, \theta_i^t) \right]. \end{aligned} \tag{10}$$

We can see that this payment policy is beneficial for both client $i$ and $j$: while client $j$ gets paid more than minimal price, client $i$ does not lose all the benefits brought by client $j$'s model. Therefore, the model transaction in our system is reciprocal and no client is conveying benefits to others for free. Also, different from simply covering client $j$'s cost by setting $r_{ij}^t = c_j$, the clients cannot directly affect the payment by manipulating $c_i$. This can significantly reduce the regret of pricing (i.e., losing money for not setting $c_i$ higher) and greedy behaviors (i.e., reporting higher $c_i$ for more profits).

# 5 ANALYSIS

In this section, we provide some theoretical discussion to show the special properties of our system. First, we show that our system is *individual rational*. Theorem 1 ensures that the clients will be satisfied with the training arrangement (the proofs in this section are in Appendix D).

**Theorem 1** (Individual Rationality)**.** *If* $\mathbf{A}^t$ *is given by Algorithm 2, then* $\forall i \in [m], t \in [T] : U_i^t = G_i(\mathbf{a}_i^t) - \sum_{j\in[m]} a_{ji}^t c_i - p_i^t \geq 0.$
Second, Theorem 2 shows that the claim of $c_i$ is incentive compatible, as increasing $c_i$ will result in less sharing of client $i$'s model. So clients can control the spread of their models.

**Lemma 2** (Incentive Compatibility of $c_i$)**.** *Denote* $\mathbf{A}^t$ *the graph calculated by the server when everyone honestly reports their* $c_i$ *and* $\hat{\mathbf{A}}^t$ *the new graph when client i report* $\hat{c}_i > c_i$*. Then* $\{j|\hat{a}_{ji}^t = 1\} \subseteq \{j|a_{ji}^t = 1\}$.
On the basis of Theorem 2 we can additionally prove Theorem 3, which ensures that the clients cannot obtain additional income by overstating $c_i$.

**Theorem 3** (Truthfulness)**.** *Denote* $U_i$ *the one-round utility of client i when everyone honestly reports their* $c_i$ *and* $\hat{U}_i^t$ *as its utility when client i reports* $\hat{c}_i > c_i$*. Then* $\forall t : \hat{U}_i^t \leq U_i^t$.
At the same time, if client $i$ reports $\hat{c}_i < c_i$, it risks selling his model at a low price and the mechanism cannot ensure $U_i^t > 0$. Thus, the clients are encouraged to reveal their true cost $c_i$ to the server. This contributes to harmonious collaboration because clients do not need to be secretive about their unwillingness to share.

We defer analysis on our system's robustness against abnormally reported data amount $N_i$ and model parameters $\theta_i^t$ in Section C.

# 6 EXPERIMENTS

## 6.1 PERFORMANCE EVALUATION

We use five image and text classification datasets commonly used in FL literature; and four instruction-tuning datasets for training instruction-following large language models. Classification datasets includes CIFAR-10 Krizhevsky et al. (2009), Fashion-MNIST Xiao et al. (2017), PACS Li et al. (2017), FEMNIST Caldas et al. (2018), and Shakespeare Caldas et al. (2018); while instruction-tuning datasets includes three financial datasets (FIQA Maia et al. (2018), TFNS Magic (2022), and NWGI Yang (2023)) and a coding dataset Chaudhary (2023). We compare our algorithm iPFL with other 7 baselines, including two general FL algorithms–FedAvg McMahan et al. (2017) and FedProx Li et al. (2020), and 5 classical PFL algorithms–Ditto Li et al. (2021a), FedAMP Huang et al. (2021), CFL Sattler et al. (2020), FedFomo Zhang et al. (2020) and pFedGraph Ye et al. (2023b).

To evaluate the economic performance of iPFL, we introduce the utility function, as defined in Definition 1. It consists of three components: collaboration gain (Equation (2)) with preference $K$, the sharing cost with individual unwillingness $c$ and accumulated payment (Equation (10)) in all rounds. To evaluate model performance, we utilize the classification accuracy metric in classification tasks. For evaluation in instruction-tuning tasks, we utilize the corresponding test dataset for financial clients to evaluate accuracy and Humaneval Chen et al. (2021) for coding clients to evaluate passing rate. The baselines, specific settings for $K$ and $c$, and implementation details are provided in the experimental section in supplementary information.

For classification tasks, we consider 9 settings with 5 datasets. For CIFAR-10 and Fashion-MNIST, we design three types of data partition among clients: termed as *NIID*, *Cluster*, and *Skew*. (i) NIID is a common setting Wang et al. (2020); Yurochkin et al. (2019); Acar et al. (2020); Ye et al. (2023a), where local data among clients follows the Dirichlet distribution (default $\beta = 0.1$). (ii) The Cluster involves random client clustering, distinguishing between high (smaller $\beta$) and low heterogeneous levels within and between groups. (iii) For Skew, total classes are divided into clusters so that in each cluster, each client possesses 5 classes. FEMNIST and Shakespeare exhibit natural heterogeneity. In the case of PACS with four domains, each cluster represents one domain, namely the Cluster partition. The evaluation results of our iPFL against eight baselines across nine settings are shown in Figure 2, emphasizing the comparisons on the trade-off between model performance and economic utility. Our iPFL achieves a comparable or even better performance with performance-oriented

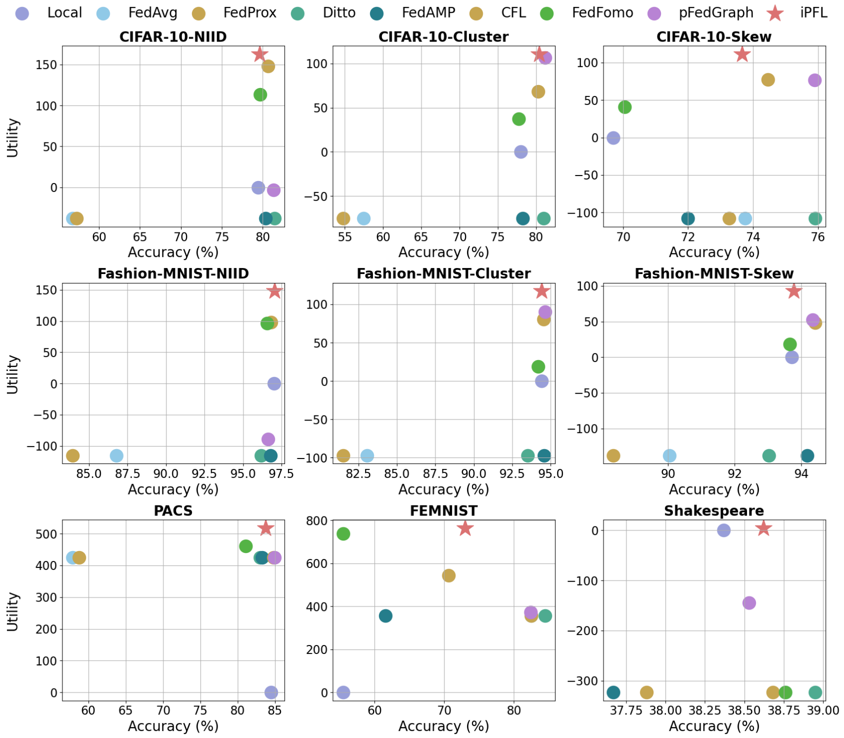


Figure 2: Comparison of average utility and accuracy in scatter under different settings. Our iPFL achieves comparable or even better model performance and the highest utility across 9 settings.

baselines. Meanwhile, iPFL consistently excels in economic utility, as evidenced by its highest plot scatter across diverse settings. Specifically, iPFL outperforms FedAMP by 1.75% in accuracy and 217.4 in utility, respectively. Overall, these results show that iPFL effectively strikes a balance between model performance and economic utility, demonstrating its capacity to harmonize model performance and economic benefits within the personalized federated learning framework.

For the instruction-tuning tasks, we consider two scenarios. (i) We configure a scenario for financial sentiment analysis with six clients, where every two clients share one of the following datasets: FIQA Maia et al. (2018), TFNS Magic (2022), or NWGI Yang (2023). (ii) We consider a more complex scenario to represent a higher heterogeneity level, where five clients possess the code data from CodeAlpaca Chaudhary (2023) and three clients own the financial data from NWGI. The results presented in Section 6.1 demonstrate the superiority of iPFL: it excels in both accuracy and utility metrics across scenarios. For example, iPFL demonstrates a remarkable 5.55% improvement in accuracy and a 58.3 gain in utility on the financial scenario compared to other baselines. Besides, in the second setting, other baselines are inferior to local training in utility for some clients, failing to guarantee the IR property. This dual achievement highlights the effectiveness of our approach in enhancing model performance and economic utility.

### 6.2 Inclusive market

To verify that our iPFL is inclusive that can include clients with diverse preferences and economic conditions, we simulate a market that consists clients with diverse roles. In the market, some traders buy and sell models, buyers who only buy models, sellers who only sell models, and attackers who try to sell poisoned models (we use a randomly parameterized model). These are achieved by setting the profiles of clients: we set the level of data eagerness as a random positive value for traders and buyers, while zero for sellers and attackers; we set the cost as a random positive number for traders, $+\infty$ for buyers, zero for sellers and attackers; see details in supplementary material. Finally, we build a market based on CIFAR-10-Cluster scenario with 12 clients and conduct model training for 20 rounds. We record the accumulated money transaction and the accuracy difference between

Table 1: Comparisons of model performance (accuracy or passing rate, %) and utility of different algorithms on two instruction-tuning scenarios. Our iPFL consistently outperforms other baselines.

| Scenarios | Finance | | | | Finance & Code | | |
|---|---|---|---|---|---|---|---|
| Evaluation | FIQA | TFNS | NWGI | Avg-Utility | NWGI | Code | Avg-Utility |
| Local | 84.02±6.09 | 80.58±0.83 | 43.17±4.48 | 0.0±0.0 | 50.61±2.63 | 13.54±2.30 | 0.0±0.0 |
| FedAvg | 78.19±1.94 | 81.25±5.30 | 52.25±4.77 | 45.9±0.0 | 49.94±2.46 | 15.00±0.70 | 58.2±150.4 |
| FedProx | 78.55±0.40 | 80.56±6.28 | 52.44±1.68 | 45.9±0.0 | 49.61±2.67 | 15.00±1.26 | 58.2±150.4 |
| FedAMP | 84.01±4.03 | 76.63±5.48 | 42.56±6.98 | 45.9±0.0 | 51.58±1.38 | 14.02±0.86 | 58.2±150.4 |
| CFL | 85.11±6.61 | 77.06±6.98 | 45.94±4.68 | 45.9±0.0 | 52.28±4.97 | 14.15±0.80 | 58.2±150.4 |
| pFedGraph | 83.65±5.57 | 76.75±5.66 | 47.94±3.09 | 45.9±0.0 | 50.00±5.65 | 14.27±1.76 | 137.7±291.6 |
| **iPFL** | **85.47±6.10** | **83.38±2.83** | **56.25±1.06** | **96.5±0.0** | **53.11±0.51** | **15.85±1.14** | **208.1±131.1** |

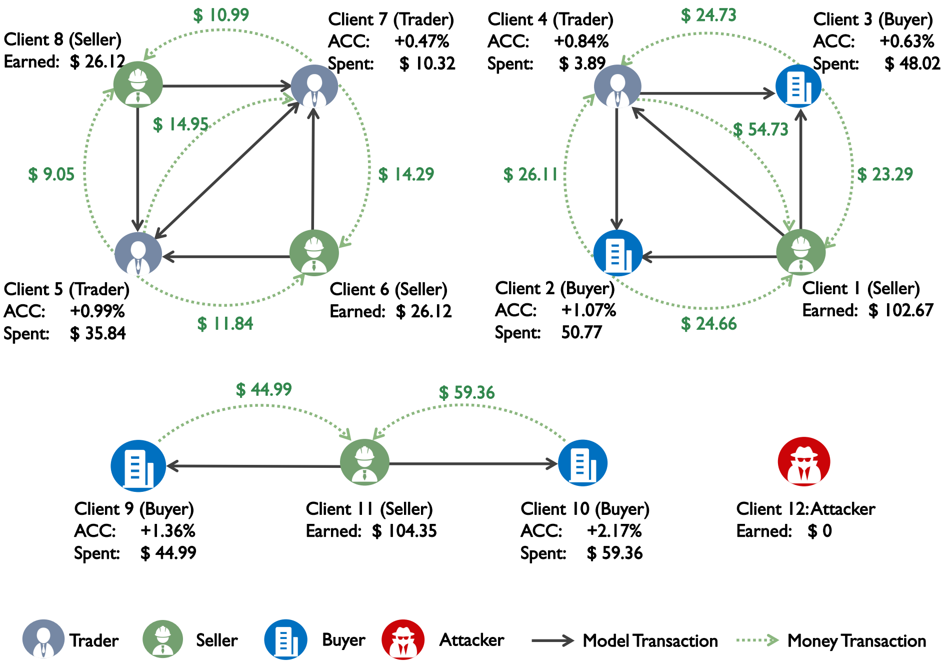


Figure 3: The illustration of the transaction graph of an inclusive market.

model trained by iPFL and local training, and demonstrate them in Figure 3. From the figure, we can clearly see that the transactions among the clients are well aligned with their roles (i.e., purposes). The traders buy models from others to obtain models with higher accuracy, and sell models to others to make a profit at the same time. The buyers pay others to buy models to improve their models while the sellers earn money by selling models. The attacker is successfully isolated by others, doing no harm to the market. Overall, the experiments verify that iPFL is an incentivized and inclusive PFL system since every unique individual gains benefits from joining the system.

## 7 Conclusion

Our work addresses the challenges of depleted publicly available data and the need for collaboration among private institutions through an inclusive sharing market. This market incentivizes diverse participants with unique model preferences and economic conditions to contribute effectively. Our proposed iPFL, as demonstrated by comprehensive experiments, excels in balancing model performance and economic utility across diverse tasks and scales. It promotes individual rationality, robustness to various model attacks and preventing dishonest practices, which contributes to a stable and trustworthy market environment. We anticipate that our iPFL can serve as a valuable technique for boosting future AI models on decentralized private data while making everyone satisfied.

## REFERENCES

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2020.

Debra Box and Dalenca Pottas. Improving information security behaviour in the healthcare context. *Procedia Technology*, 9:1093–1103, 2013.

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

Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

Sahil Chaudhary. Code alpaca: An instruction-following llama model for code generation. `https://github.com/sahil280114/codealpaca`, 2023.

Guangyan Chen, Meiling Wang, Yi Yang, Kai Yu, Li Yuan, and Yufeng Yue. Pointgpt: Auto-regressively generative pre-training from point clouds. *arXiv preprint arXiv:2305.11487*, 2023.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021.

Mingshu Cong, Han Yu, Xi Weng, Jiabao Qu, Yang Liu, and Siu Ming Yiu. A vcg-based fair incentive mechanism for federated learning. *arXiv preprint arXiv:2008.06680*, 2020.

Yongheng Deng, Feng Lyu, Ju Ren, Yi-Chao Chen, Peng Yang, Yuezhi Zhou, and Yaoxue Zhang. Fair: Quality-aware federated learning with precise user incentive and model aggregation. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pp. 1–10. IEEE, 2021.

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in Neural Information Processing Systems*, 33:3557–3568, 2020.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Jingoo Han, Ahmad Faraz Khan, Syed Zawad, Ali Anwar, Nathalie Baracaldo Angel, Yi Zhou, Feng Yan, and Ali R Butt. Tiff: Tokenized incentive for federated learning. In *2022 IEEE 15th International Conference on Cloud Computing (CLOUD)*, pp. 407–416. IEEE, 2022.

Jigna J Hathaliya and Sudeep Tanwar. An exhaustive survey on security and privacy issues in healthcare 4.0. *Computer Communications*, 153:311–335, 2020.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2021.

Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. Personalized cross-silo federated learning on non-iid data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 7865–7873, 2021.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

Georgios Kaissis, Alexander Ziller, Jonathan Passerat-Palmbach, Théo Ryffel, Dmitrii Usynin, Andrew Trask, Ionésio Lima Jr, Jason Mancuso, Friederike Jungmann, Marc-Matthias Steinborn, et al. End-to-end privacy preserving deep learning on multi-institutional medical imaging. *Nature Machine Intelligence*, 3(6):473–484, 2021.

Jiawen Kang, Zehui Xiong, Dusit Niyato, Shengli Xie, and Junshan Zhang. Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory. *IEEE Internet of Things Journal*, 6(6):10700–10714, 2019.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Sai Praneeth Karimireddy, Wenshuo Guo, and Michael I Jordan. Mechanisms that incentivize data sharing in federated learning. *arXiv preprint arXiv:2207.04557*, 2022.

Ahmad Faraz Khan, Xinran Wang, Qi Le, Azal Ahmad Khan, Haider Ali, Jie Ding, Ali Butt, and Ali Anwar. Pi-fl: Personalized and incentivized federated learning. *arXiv preprint arXiv:2304.07514*, 2023.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 5542–5550, 2017.

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

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning*, pp. 6357–6368. PMLR, 2021a.

Xin Li, Guodong Cheng, Liangxu Wang, Juanle Wang, Youhua Ran, Tao Che, Guoqing Li, Honglin He, Qiang Zhang, Xiaoyi Jiang, et al. Boosting geoscience data sharing in china. *Nature Geoscience*, 14(8):541–542, 2021b.

Yuan Liu, Mengmeng Tian, Yuxin Chen, Zehui Xiong, Cyril Leung, and Chunyan Miao. A contract theory based incentive mechanism for federated learning. In *Federated and Transfer Learning*, pp. 117–137. Springer, 2022.

Hongtao Lv, Zhenzhe Zheng, Tie Luo, Fan Wu, Shaojie Tang, Lifeng Hua, Rongfei Jia, and Chengfei Lv. Data-free evaluation of user contributions in federated learning. In *2021 19th International Symposium on Modeling and Optimization in Mobile, Ad hoc, and Wireless Networks (WiOpt)*, pp. 1–8. IEEE, 2021.

Neural Magic. Twitter financial news sentiment. *https://huggingface.co/datasets/zeroshot/twitter-financial-news-sentiment*, 2022.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. Www'18 open challenge: Financial opinion mining and question answering. *Companion Proceedings of the The Web Conference 2018*, 2018. URL `https://api.semanticscholar.org/CorpusID:13866508`.

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

Niklas Muennighoff, Alexander M Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling data-constrained language models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=j5BuTrEj35.

OpenAI. Gpt-4 technical repor. *arXiv preprint arXiv:2303.08774*, 2023.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *NIPS*, 35:27730–27744, 2022a.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022b.

W Nicholson Price and I Glenn Cohen. Privacy in the age of medical big data. *Nature medicine*, 25 (1):37–43, 2019.

Tao Qi, Fangzhao Wu, Chuhan Wu, Liang He, Yongfeng Huang, and Xing Xie. Differentially private knowledge transfer for federated learning. *Nature Communications*, 14(1):3785, 2023.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems*, 32(8):3710–3722, 2020.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

Zhuan Shi, Lan Zhang, Zhenyu Yao, Lingjuan Lyu, Cen Chen, Li Wang, Junhao Wang, and Xiang-Yang Li. Fedfaim: A model performance-based fair incentive mechanism for federated learning. *IEEE Transactions on Big Data*, 2022.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*, 2023.

Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405, 2020.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Pablo Villalobos, Jaime Sevilla, Lennart Heim, Tamay Besiroglu, Marius Hobbhahn, and Anson Ho. Will we run out of data? an analysis of the limits of scaling datasets in machine learning. *arXiv preprint arXiv:2211.04325*, 2022.

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

Guan Wang, Charlie Xiaoqian Dang, and Ziye Zhou. Measure contribution of participants in federated learning. In *2019 IEEE international conference on big data (Big Data)*, pp. 2597–2604. IEEE, 2019.

Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=BkluqlSFDS`.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, et al. How far can camels go? exploring the state of instruction tuning on open resources. *arXiv preprint arXiv:2306.04751*, 2023.

Chuhan Wu, Fangzhao Wu, Lingjuan Lyu, Tao Qi, Yongfeng Huang, and Xing Xie. A federated graph neural network framework for privacy-preserving personalization. *Nature Communications*, 13(1):3091, 2022.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Hongyang Yang. Data-centric fingpt. open-source for open finance. `https://github.com/AI4Finance-Foundation/FinGPT`, 2023.

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

Rui Ye, Yaxin Du, Zhenyang Ni, Siheng Chen, and Yanfeng Wang. Fake it till make it: Federated learning with consensus-oriented generation. *arXiv preprint arXiv:2312.05966*, 2023a.

Rui Ye, Zhenyang Ni, Fangzhao Wu, Siheng Chen, and Yanfeng Wang. Personalized federated learning with inferred collaboration graphs. In *International Conference on Machine Learning*, pp. 39801–39817. PMLR, 2023b.

Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, Nghia Hoang, and Yasaman Khazaeni. Bayesian nonparametric federated learning of neural networks. In *International Conference on Machine Learning*, pp. 7252–7261. PMLR, 2019.

Rongfei Zeng, Chao Zeng, Xingwei Wang, Bo Li, and Xiaowen Chu. A comprehensive survey of incentive mechanism for federated learning. *arXiv preprint arXiv:2106.15406*, 2021.

Yufeng Zhan, Peng Li, Zhihao Qu, Deze Zeng, and Song Guo. A learning-based incentive mechanism for federated learning. *IEEE Internet of Things Journal*, 7(7):6360–6368, 2020.

Jie Zhang, Song Guo, Xiaosong Ma, Haozhao Wang, Wenchao Xu, and Feijie Wu. Parameterized knowledge transfer for personalized federated learning. *Advances in Neural Information Processing Systems*, 34:10092–10104, 2021.

Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M Alvarez. Personalized federated learning with first order model optimization. *arXiv preprint arXiv:2012.08565*, 2020. FedFomo, weighted combination method, heterogeneity, exchange all.

# A Algorithm Pseudo Codes

**Algorithm 1** Overview of iPFL

```
input: $m$ clients, each with a local private dataset for training.
Server sends an initial model $\theta^0$ to every client.
for $i = 1$ to $m$ do
    Client $i$ performs local training and obtains $\theta_i^1 = \arg\min_{\theta_i} L_i(\theta_i) + \frac{\lambda}{2\eta}\|\theta_i^t - \bar{\theta}_i^0\|_2^2$.
    Client $i$ reports $n_i, c_i, G_i(\cdot)$ back to Server.
end for
for $t = 1$ to $T - 1$ do
    for $i = 1$ to $m$ do
        Client $i$ reports $\theta_i^t$ back to Server.
    end for
    for $i = 1$ to $m$ do
        Server calculates $\boldsymbol{A}_i^t$ by Algorithm 2.                    \\ Graph Topology Learning
    end for
    for $i = 1$ to $m$ do
        Server calculates $p_i^t$ according to $\boldsymbol{A}^t$.                   \\ Payment Calculation
        if Client $i$ pays $p_i^t$ to Server then
            Server calculates the prox-center $\bar{\theta}_i^t = \theta_i^t - \frac{\eta}{n_i}\sum_{j\in[m]} a_{ij} n_j \nabla_{\theta_i} d(\theta_i, \theta_j^t)$.
            Server sends the prox-center model $\bar{\theta}_i^t$ to Client $i$.
            Client $i$ updates $\theta_i^{t+1} = \arg\min_{\theta_i} L_i(\theta_i) + \frac{\lambda}{2\eta}\|\theta_i^t - \bar{\theta}_i^t\|_2^2$. \\ Local Model Training
        else
            Client $i$ quits and takes the best $\theta_i \in \{\theta_i^{t'} | t' \leq t\}$.
        end if
    end for
end for
output: $\theta_i, i \in [m]$ for each client.
```

**Algorithm 2** Graph Topology Learning

```
input: $g_i, \{\theta_1^t, ..., \theta_m^t\}, \{c_1, ..., c_m\}, \{n_1, ..., n_m\}$.    \\ Here $g_i(x) = \sqrt{\frac{K_i}{n_i}} - \sqrt{\frac{K_i}{n_i+x}}$
initialization: $\boldsymbol{A}_i = \boldsymbol{0}, n = 0$
for $j = 1$ to $m$ do
    Calculate threshold $n_j^{Th}$ by solving $g_i(n_j^{Th}) - g_i(n_j^{Th} - n_j) = c_j + \lambda\frac{n_j}{n_i} d(\theta_i^t, \theta_j^t)$.
    if no solution then
        Set $n_j^{Th} = 0$.
    end if
end for
for $j = 1$ to $m$ do
    $k = \arg\max_j n_j^{Th} \quad s.t. A_{ij} = 0$
    if $n + n_k < n_k^{Th}$ then
        $A_{ik} = 1$                        \\ Add the remaining client with the largest threshold
        $n = n + n_k$
    else
        break                               \\ Stop adding if total data amount reaches the threshold
    end if
end for
output: $\boldsymbol{A}_i$.
```

# B Related Work on Incentive federated learning

There have been extensive works on incentive FL mechanisms by introducing economic rewards. To appropriately assign economic rewards, one direct approach is to allocate rewards by precise and fair contribution assessment(Wang et al., 2019; Lv et al., 2021; Shi et al., 2022). For example, in FedFAIM by Shi et al. (2022), the amount of reward is decided by model quality assessment and Shapley value-based contribution measure. Another approach to assign economic rewards by combining FL with game theory models like reverse auction Deng et al. (2021); Cong et al. (2020) or contract theory(Liu et al., 2022; Kang et al., 2019). For example, in FAIR by Deng et al. (2021), a reverse auction is conducted to recruit clients by their bids and model quality, which ensures clients' incentive and server's model quality at the same time.

Nonetheless, present standalone solutions are usually designed for traditional FL framework (all clients contribute to one central model) and fixed market structure (server buys models from clients), which cannot be applied to PFL scenarios and complicated inclusive markets. To overcome these limitations, there have been some discoveries about incentive FL mechanisms under personalized model needs or heterogeneous economic conditions. In addressing personalized model needs, Khan et al. (2023) proposes an incentivized PFL framework, PIFL, which can provide personalized models by client clustering and incentivize the clients by setting prices for participating in a cluster. In the face of heterogeneous economic conditions, Han et al. (2022) proposes TIFF, where the clients are considered providers and consumers of the central model. However, neither of them considers both personalized model needs and heterogeneous economic conditions at the same time, so there is still a gap between present mechanisms and an inclusive and incentivized PFL system.

Therefore, in our work, we first consider a graph-based PFL framework, which can reserve a personalized model for each client and clearly describe the collaboration topology of the clients. Then we design a graphical game in such graph-based PFL, where each client has its unique economic profile: it can be a seller, buyer or both. With such a design, we can evaluate the personalized needs of each client and set up an inclusive market system in PFL.

## C Theory

Here, we discuss our system's robustness against abnormally reported data amount $N_i$ and model parameters $\theta_i^t$. For benign and quality-aware clients, they have no reason to be dishonest about $N_i$ and $\theta_i^t$ as lying about $N_i$ and $\theta_i^t$ is harmful to their models: reporting wrong $N_i$ will result in inaccurate model aggregation; uploading fake $\theta_i^t$ may result in less personalization. However, malicious attackers can upload noisy models to attack the system or exaggerate their data amount to defraud extra payment and increase their weight in others' models. To address this issue, client selection procedure considers both data amount and model similarity in Algorithm 2. Theorem 4 shows that malicious clients who upload abnormal $N_i$ and $\theta_i$ are likely to be isolated without introducing extra efforts of model verification (e.g., testing models on a validation set).

**Theorem 4** (Robustness against abnormal data amount)**.** *If $\mathbf{A}^t$ is given by Algorithm 2 and $N_i \to +\infty$, then $\forall j \in [m] : a_{ji}^t = 0$.*

As a result, the malicious clients whose priority is attacking the federation can only be trusted by other clients when they report a relatively smaller data volume. This means that their impact is limited: if client $j$ reports a smaller $N_j$, it will receive a smaller reference weight ($N_j/N_i$) in the second term of Equation (5) and other clients will not strongly emphasize its parameters in aggregation. Besides, the consequence of uploading a fake model is similar. If client $i$ is malicious and uploads a fake model $\theta_i'$ (e.g., perturbing the real model parameters), $\theta_i'$ is very likely to be different from other clients' normal models that are trained on real datasets, indicating that the model difference term $\|\theta_i - \theta_j\|_2^2$ is large, which will increase the value of $N_i^{Thresh}$ calculated by other clients and make client $i$ less possible to be chosen. Therefore, the influence of such malicious clients is also limited.

## D Proofs for the theories

Here we present the proofs of all the theories in our paper.

**Theorem 1** (Individual Rationality)**.** *If $\mathbf{A}^t$ is given by Algorithm 2, then $\forall i \in [m], t \in [T] : U_i^t = G_i(\mathbf{a}_i^t) - \sum_{j\in[m]} a_{ji}^t c_i - p_i^t \geq 0$.*

*Proof.* According to the definition of economic utility $U_i$ in Definition 1 and payment $p_i^t$ in Equation (10), the utility of client $i$ in round $t$ is:

$$\begin{aligned}U_i^t &= G_i(\mathbf{a}_i^t) - \sum_{j\in[m]} a_{ji}^t c_i - p_i^t \\ &= G_i(\mathbf{a}_i^t) - \sum_{j:a_{ij}^t=1}\left[G_i(\mathbf{a}_i^t) - G_i(\mathbf{a}_i^t - \mathbf{e}_j) - \lambda\frac{N_j}{N_i}d(\theta_i^t,\theta_j^t)\right] \\ &\quad + \sum_{j:a_{ji}^t=1}\left[G_j(\mathbf{a}_j^t) - G_j(\mathbf{a}_j^t - \mathbf{e}_i) - \lambda\frac{N_i}{N_j}d(\theta_j^t,\theta_i^t) - c_i\right]. \end{aligned} \tag{11}$$

If $\mathbf{a}_i^t$ is a solution given by Algorithm 2, then $\forall i \in [m], \forall j \in [m]$:

$$\begin{aligned}& a_{ij}^t = 1 \\ \Rightarrow & \phi_i(\mathbf{a}_i^t - \mathbf{e}_j) > \phi_i(\mathbf{a}_i^t) \\ \Rightarrow & G_i(\mathbf{a}_i^t) - G_i(\mathbf{a}_i^t - \mathbf{e}_j) > c_j + \lambda\frac{N_j}{N_i}d(\theta_i^t,\theta_j^t). \end{aligned} \tag{12}$$

Since $g_i(x) = \sqrt{\frac{K_i}{N_i}} - \sqrt{\frac{K_i}{N_i+x}}$ is a concave and increasing function with $g_i(0) = 0$, there is:

$$\begin{aligned}G_i(\mathbf{a}_i^t) =& g_i(\sum_{k:a_{ik}^t=1} N_k) - g_i(0) \\ =& \sum_{j:a_{ij}^t=1}\left[g_i(N_j + \sum_{\substack{k:k<j\\ a_{ik}^t=1}} N_k) - g_i(\sum_{\substack{k:k<j\\ a_{ik}^t=1}} N_k)\right] \\ \geq& \sum_{j:a_{ij}^t=1}\left[g_i(N_j + \sum_{\substack{k:k\neq j\\ a_{ik}^t=1}} N_k) - g_i(\sum_{\substack{k:k\neq j\\ a_{ik}^t=1}} N_k)\right] \\ =& \sum_{j:a_{ji}^t=1}\left[G_i(\mathbf{a}_i^t) - G_i(\mathbf{a}_i^t - \mathbf{e}_j)\right]. \end{aligned} \tag{13}$$

By Equation (12) and Equation (13), we have:

$$\begin{aligned}U_i^t =& G_i(\mathbf{a}_i^t) - \sum_{j:a_{ij}^t=1}\left[G_i(\mathbf{a}_i^t) - G_i(\mathbf{a}_i^t - \mathbf{e}_j)\right] \\ &+ \sum_{j:a_{ji}^t>0}\left[G_j(\mathbf{a}_j^t) - G_j(\mathbf{a}_j^t - \mathbf{e}_i) - \lambda\frac{N_i}{N_j}d(\theta_j^t,\theta_i^t) - c_i\right] \\ &+ \sum_{j:a_{ij}^t>0}\lambda\frac{N_j}{N_i}d(\theta_i^t,\theta_j^t) \\ \geq& \sum_{j:a_{ij}^t>0}\lambda\frac{N_j}{N_i}d(\theta_i^t,\theta_j^t) \geq 0. \end{aligned}$$

□

**Lemma 2** (Incentive Compatibility of $c_i$)**.** *Denote* $\mathbf{A}^t$ *the graph calculated by the server when everyone honestly reports their* $c_i$ *and* $\hat{\mathbf{A}}^t$ *the new graph when client i report* $\hat{c}_i > c_i$*. Then* $\{j|\hat{a}_{ji}^t = 1\} \subseteq \{j|a_{ji}^t = 1\}$*.*

*Proof.* Denote the original and new threshold for client $i$ calculated by client $j$ in Algorithm 2 as $N_i^{Th}$ and $\hat{N}_i^{Th}$. There is $\hat{c}_i > c_i \Rightarrow \hat{N}_i^{Th} < N_i^{Th}$. So client $i$ will not be added to client $j$'s collaborators earlier. By the time client $i$ is considered, $\hat{N}_i^{Th} < N_i^{Th} \leq \sum_{k\in m} a_{jk}N_k \leq \sum_{k\in m} \hat{a}_{jk}N_k$, so $a_{ji}^t = 0 \Rightarrow \hat{a}_{ji} = 0$. Therefore, $\{j|\hat{a}_{ji}^t = 1\} \subseteq \{j|a_{ji}^t = 1\}$. □

**Theorem 3** (Truthfulness). *Denote $U_i$ the one-round utility of client $i$ when everyone honestly reports their $c_i$ and $\hat{U}_i^t$ as its utility when client $i$ report $\hat{c}_i > c_i$. Then $\forall t : \hat{U}_i^t \leq U_i^t$.*

*Proof.* First, the claim of cost $c_i$ will not change the procedure of choosing collaborators of client $i$ in Algorithm 2, so there is:

$$\forall j \in [m] : a_{ij}^t = \hat{a}_{ij}^t. \tag{14}$$

Second, the change of $c_i$ does not influence the marginal benefit of other clients. If client $j$ keeps the same choice for client $i$, then its order of adding other clients should also be unchanged, so there is:

$$\forall j \in [m] : a_{ji}^t = \hat{a}_{ji}^t \Rightarrow \mathbf{a}_j^t = \hat{\mathbf{a}}_j^t, \tag{15}$$

Finally, by Equation (11), Equation (12), Equation (14), Equation (15), and Theorem 2, there is:

$$\begin{aligned} U_i^t - \hat{U}_i^t = & \sum_{j:a_{ji}^t=1} \left[ G_j(\mathbf{a}_j^t) - G_j(\mathbf{a}_j^t - \mathbf{e}_i) - \lambda \frac{N_i}{N_j} d(\theta_j^t, \theta_i^t) - c_i \right] \\ & - \sum_{j:\hat{a}_{ji}^t=1} \left[ G_j(\hat{\mathbf{a}}_j^t) - G_j(\hat{\mathbf{a}}_j^t - \mathbf{e}_i) - \lambda \frac{N_i}{N_j} d(\theta_j^t, \theta_i^t) - c_i \right] \\ & = \sum_{j:a_{ji}t - \hat{a}_{ji}^t=1} \left[ G_j(\mathbf{a}_j^t) - G_j(\mathbf{a}_j^t - \mathbf{e}_i) - \lambda \frac{N_i}{N_j} d(\theta_j^t, \theta_i^t) - c_i \right] \geq 0 \end{aligned}$$

which means $\hat{U}_i^t \leq U_i^t$. □

**Theorem 4** (Robustness against abnormal data amount). *If $\mathbf{A}^t$ is given by Algorithm 2 and $N_i \to +\infty$, then $\forall j \in [m] : a_{ji}^t = 0$.*

*Proof.* For any $\mathbf{A}^t$ attained by Algorithm 2 there is:

$$a_{ji}^t = 1 \Rightarrow G_j(\mathbf{a}_j^t) - G_j(\mathbf{a}_j^t - \mathbf{e}_i) - c_i - \lambda \frac{N_i}{N_j} d(\theta_i, \theta_j) > 0. \tag{16}$$

Consider $i$ reports an enormous $N_i \to +\infty$. Assume $a_{ji}^t = 1$. Since the change of $G_i$ is bounded:

$$G_j(\mathbf{a}_j^t) - G_j(\mathbf{a}_j^t - \mathbf{e}_i) \leq \sqrt{K_j/N_j},$$

we have:

$$G_j(\mathbf{a}_j^t) - G_j(\mathbf{a}_j^t - \mathbf{e}_i) - c_i - \lambda \frac{N_i}{N_j} d(\theta_i, \theta_j) \to -\infty.$$

This contradicts to Equation (16). So $a_{ji}^t = 0$. □

## E Experimental Details

### E.1 Baselines

We compare our method with the other 5 personalized baselines. 1) Ditto Li et al. (2021a) applies an alternating optimization approach to jointly solve for the global model and personalized models. 2) FedAMP Huang et al. (2021) encourages collaborations between clients with similar model parameters via empirical exponential weight calculation. 3) CFL Sattler et al. (2020) divides clients into two clusters, minimizing the maximum similarity between clients from different clusters when the stopping criterion is violated. 4)FedFomo Zhang et al. (2020) introduces a mechanism where clients evaluate the performance of received models on their target task, using these evaluations to weight each model's parameters in a personalized update. 5) pFedGraph Ye et al. (2023b) inferring the collaboration graph based on model similarity via solving the corresponding optimization problem.

### E.2 Hyperparameter Settings

**Hyperparameters in the utility function.** In performance evaluation, we uniformly set $c_i = 1$, and $K$ depends on the data size in specific dataset. Simply speaking, since the model differences via cosine similarity are within a certain range, the larger the data volume, the corresponding $K$ should be larger. Specifically, we set $K = 5E5$ for 5 classical datasets: CIFAR-10, Fashion-MNIST, PACS, FEMNIST and Shakespeare; and choose $K = 2E4$ and $K = 5E4$ separately for mixed-Finance and Code+Finance dataset.

**Hyperparameters in algorithms.** The hyperparameters for various federated learning baselines are detailed below:

- FedProx Li et al. (2020): $\mu = 0.1$, controlling the impact of local regularization on training loss.
- Ditto Li et al. (2021a): $\lambda = 1$, governing the interpolation between local and global models.
- CFL Sattler et al. (2020): $\epsilon_1 = 2.0, \epsilon_2 = 2.5$, parameters in the process of splitting clients.
- FedFomo Zhang et al. (2020): $M = 6$, indicating the maximum number of models sent from the server to clients.
- FedAMP Huang et al. (2021): $\lambda = 0.01$, where $\lambda$ serves as a regularization parameter.
- pFedGraph Ye et al. (2023b): $\alpha = 0.8, \lambda = 0.01$, where $\alpha$ influences the collaboration graph optimization, and $\lambda$ balances individual utilities with collaboration necessity.

The hyperparameters $\lambda$ and $\eta$ in iPFL are tuned according to the demand of participants (reflected by $K_i$) in different datasets. The tuning of $\lambda$ and $\eta$ used in our experiments is shown in Table 2.

Table 2: The tuned optimal hyperparameters of our algorithm for each dataset.

| Dataset | CIFAR-10 | F-MNIST | PACS | FEMNIST | Shakes. | mixed-Fin. | Code+Fin. |
|---|---|---|---|---|---|---|---|
| $\lambda$ | 2 | 0.001 | 0.0001 | 1 | 2 | 5 | 1 |
| $\eta$ | 5 | 5 | 5 | 5 | 5000 | 1 | 5 |

**Hyperparameters in FL setting.** For the number of clients and corresponding average data size in each setting, we list the information in Table 3. Especially for FEMNIST and Shakespeare, we select the 20 and 10 clients with the largest amount of data respectively.

### E.3 Implementing details

**Classification tasks.** We use the same setup for all the baselines and our method. Specifically, we run FL for 50 communication rounds and train local models for $\tau = 200$ iterations (except $\tau = 50$ iterations for FEMNIST) with a batch size 64. We use SGD optimizer with a learning rate of 0.01. We utilize CNN-based networks for image classification tasks, and LSTM for text-related tasks.

**Instruction-tuning tasks** The QA tasks involve training the smallest Llama2 model, boasting 7 billion parameters. We employ the conventional supervised fine-tuning (SFT) method Ouyang et al. (2022a) and integrate quantization and parameter-efficient fine-tuning techniques Hu et al. (2021)

Table 3: Information of dataset across different settings in our experiments. We use F-MNIST to short for Fashion-MNIST, Shakes. for Shakespeare and similarly for mixed-Finance and Code+Finance.

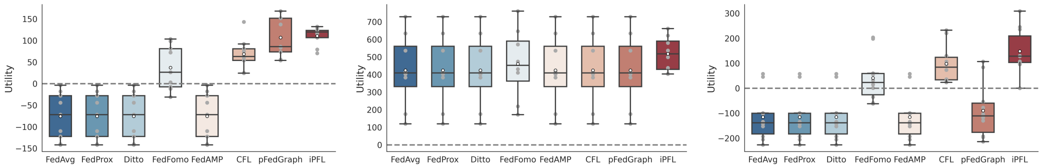
| Dataset | CIFAR-10 & F-MNIST | | | PACS | FEMNIST | Shakes. | mixed-Fin. | Code+Fin. |
|---|---|---|---|---|---|---|---|---|
| Partition | NIID | Cluster | Skew | Cluster | Natural | | - | - |
| # Client | 10 | 9 | 10 | 8 | 20 | 10 | 6 | 8 |
| # Data size | 5000&6000 | | | 999 | 443 | 38682 | 200 | 387 |
| Data type | image | | | image | image | text | text | text |

to reduce trainable parameters. We conduct 50 communication rounds of Federated Learning (FL). The initial learning rate starts at $5 \times 10^{-5}$ in the first round and diminishes to $1 \times 10^{-6}$ by the final round. The batch size is set to 8 and the maximum sequence length is set to 512. The rank of LoRA Hu et al. (2021) is set at 32, with a scalar $\alpha$ value of 64. We adhere to the Alpaca Taori et al. (2023) template for formatting instructions.

### E.4 Experiments of incentive properties

### E.5 Incentive properties

**Individual rationality.** Here, we show the individual client utility distribution on CIFAR-10-Cluster, PACS and Fashion-MNIST-NIID scenarios in Figure 4. We compare iPFL with 7 representative baselines. Remarkably, in these scenarios, our proposed iPFL ensures that the utility of each client remains positive, outperforming all the other algorithms. These experiments convincingly verify that our proposed iPFL ensures the property of individual rationality (i.e., every participant benefits from the system), a critical property to incentivize institutions to join the market willingly Kang et al. (2019); Zeng et al. (2021). Note that we accordingly provide the theoretical guarantee in Theorem 1.



(a) CIFAR-10-Cluster (b) PACS (c) Fashion-MNIST-NIID

Figure 4: The utility distribution of clients with different algorithms under three settings. Specifically, the circle denotes the mean utility of all clients, and the gray scatter represents the individual client utility values. Our iPFL guarantees positive utility for each client and achieves the highest average utility.

**Robustness.** In this part, we investigate the robustness of FL algorithms against four distinct types of model poisoning attackers Ye et al. (2023b). Attack strategies include (a) shuffling model updates, (b) flipping the numerical sign of model updates, (c) manipulating model updates with the same value at each element, and (d) manipulating model updates based on random Gaussian noises. Based on the CIFAR-10-Cluster scenario, we conduct one experiment for each attack type, where one attacker is introduced. We compare our iPFL with four representative state-of-the-art PFL algorithms: Ditto Li et al. (2021a), pFedGraph Ye et al. (2023b), CFL Sattler et al. (2020), and FedAMP Huang et al. (2021). In Figure 5, we illustrate the changes in the averaged performance of benign clients and the utility of malicious clients after being exposed to attack. Notably, only our iPFL succeeds in reducing the utility of the malicious attacker while simultaneously maintaining accuracy for the benign clients. This unique characteristic positions our iPFL as a robust technical foundation for a healthy model-sharing market.

**Truthfulness.** Here, we explore the effects of clients being dishonest by considering a scenario where one client lies about the dataset size or training cost. In Table 4, we show the liar's accuracy and utility over different lying ratios (compared with true value). The table shows that in our iPFL, the liar always achieves lower or the same accuracy, and significantly lower utility. These results verify that one cannot benefit by lying, which demonstrates the effectiveness of our iPFL in discouraging dishonest behaviors, contributing to promote the healthy development of the market. Note that we accordingly provide the theoretical interpretation in Theorem 3.

### E.6 Additional details for inclusive market

The data distribution of each client in the simulation of our proposed inclusive market is shown in Figure 6. As illustrated, we use a smaller value of $\beta = 10$ to create a less heterogeneous intra-cluster, while maintaining a high level of heterogeneity inter-cluster with $\beta = 0.1$.

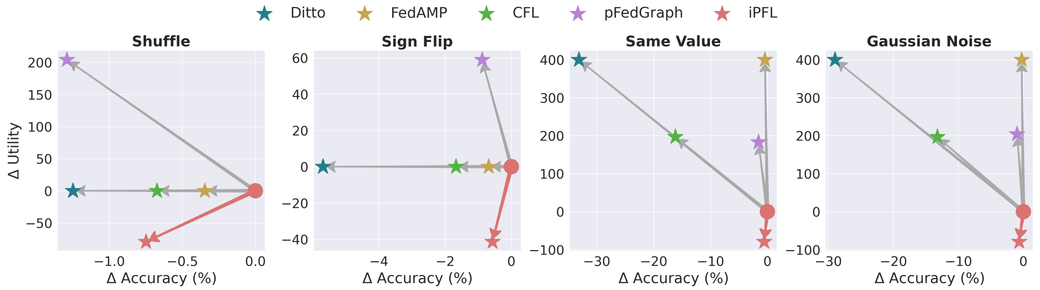


Figure 5: The change of average benign clients' performance (%) and malicious client utility after 4 different attack types of an attacker, under the Cluster setting on CIFAR-10. For each algorithm, we utilize the circle ● and star ★ to separately represent the benign clients' states with their mean accuracy and utility.

Table 4: Liar' performance(%) and utility comparison under different lying ratios (1 denotes honest) of her true data size or cost, under the NIID setting of CIFAR-10. Lying on reported private information causes performance degradation and loss of earnings.

| Cases | Honest | Lying on data size | | | Lying on cost | | |
|---|---|---|---|---|---|---|---|
| Lying Ratio | 1 | 0.1 | 0.5 | 10 | 2 | 5 | 10 |
| Liar's Accuracy | **75.430** | 66.933 | 75.331 | 75.430 | 75.430 | 75.430 | 75.430 |
| Liar's Utility | **617.295** | -513.410 | -9.951 | 0.00 | 0.00 | 0.00 | 0.00 |

Our inclusive market simulation, for each client $i$, denotes $K_i$ to represent the level of data eagerness and denotes $c_i$ as the cost of sharing a model concerning privacy and communication consumption. Therefore, we utilize both $K_i$ and $c_i$ to portray 4 different client types within the personalized inclusive market in Figure 1. For instance, a client with $c_i$ setting to $\infty$ is not willing to share its model, namely a model buyer. In our experiments, the profile ($K_i$ and $c_i$) of each kind of client is randomly generated according to Table 6. Since only the traders and the buyers desire to attain others' models, so for these two types $K_i > 0$, with larger $K_i$ indicating the stronger tendency of buying models. We set $c_i = +\infty$ for the buyers to show their ban on sharing their own models. For traders, we randomly generate $c_i$ to show their different reluctance to model sharing. Naturally, sellers and attackers with $c_i = 0$ urge to sell (or spread) their models. We assume the attacker uploads a random model in each round.

Table 5 shows not only the model improvement and utility of each client but also the model transaction and total money transaction.

### E.7 Validation on gain function

Our gain function comes from the theoretical improvement of model performance brought by extra data resource. By McDiarmid's inequality, if $\sup_{\theta,\mathbf{z}} l(\theta; \mathbf{z}) \leq B$, with probability $1 - \delta$:

$$L_i^*(\theta_i) \leq L_i(\theta_i) + B\sqrt{\frac{\ln(1/\delta)}{2N_i}}.$$

Therefore, assuming client $i$ has correctly imported models under similar data distribution, in this paper we define its collaboration gain as the change of the last term after importing others' models: $G_i(\mathbf{a}_i) = \sqrt{\frac{K_i}{N_i}} - \sqrt{\frac{K_i}{N_i + \sum_{j\in[m]} a_{ij} N_j}}$. In our assumption for modeling the expected performance gain, we rewrite the accuracy gain for the 10-classification task on CIFAR-10 Krizhevsky et al. (2009) as

$$ACC \approx ACC_0 - \sqrt{\frac{K}{N}} = \frac{p_1}{\sqrt{N}} + p_2, \tag{17}$$

where $p_1$ and $p_2$ are undetermined fitting coefficients. We conduct experiments of local training on CIFAR-10 with different data sizes under the IID (independent and identical distribution) to find the

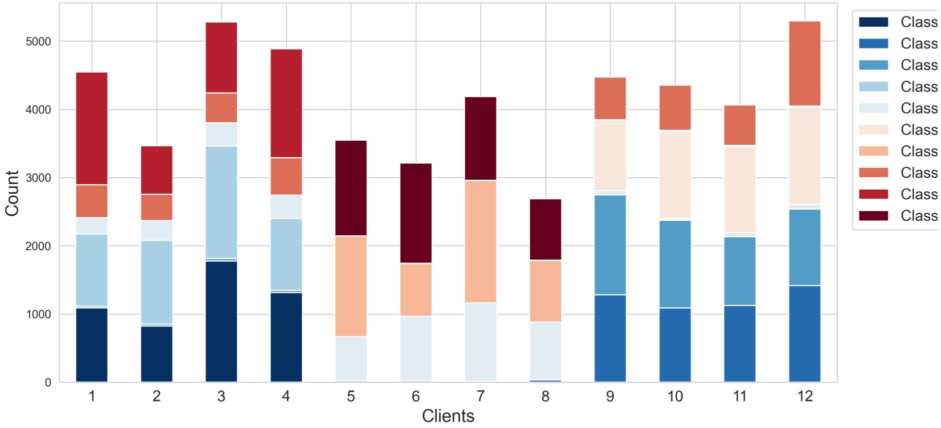


Figure 6: The data distribution of the clients (without attacker).

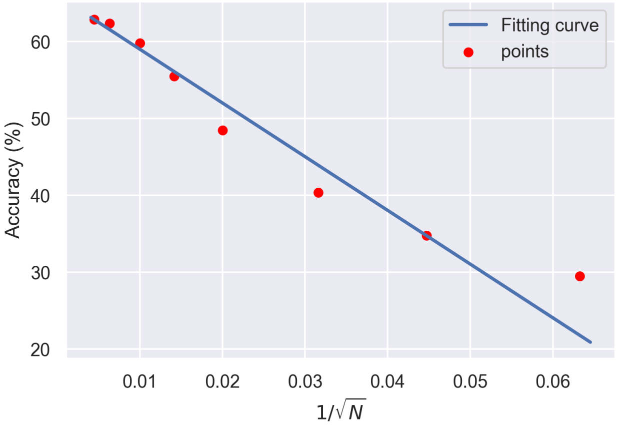


Figure 7: The fitting curve of expected performance gain and real scatter data.

Table 5: The details of inclusive market simulation.

| Client | Type | $N_i$ | $K_i$ | $c_i$ | Local ACC (%) | Fed ACC (%) | ACC Increase (%) | Balance ($) | Utility |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Seller | 4546 | 0 | 0.00 | 70.10 | 70.10 | 0.00 | 102.67 | 102.67 |
| 2 | Buyer | 3468 | 414917 | $+\infty$ | 69.17 | 70.23 | 1.07 | -50.77 | 53.04 |
| 3 | Buyer | 5278 | 468962 | $+\infty$ | 71.91 | 72.54 | 0.63 | -48.02 | 34.41 |
| 4 | Trader | 4885 | 550503 | 0.85 | 70.24 | 71.08 | 0.84 | -3.89 | 23.46 |
| 5 | Trader | 3550 | 563596 | 0.82 | 85.65 | 86.63 | 0.99 | -35.84 | 78.59 |
| 6 | Seller | 3212 | 0 | 0.00 | 83.55 | 83.55 | 0.00 | 26.12 | 26.12 |
| 7 | Trader | 4185 | 382752 | 0.78 | 84.42 | 84.88 | 0.47 | -10.32 | 51.57 |
| 8 | Seller | 2688 | 0 | 0.00 | 82.48 | 82.48 | 0.00 | 20.04 | 20.04 |
| 9 | Buyer | 4473 | 351546 | $+\infty$ | 74.07 | 75.42 | 1.36 | -44.99 | 3.98 |
| 10 | Buyer | 4353 | 590118 | $+\infty$ | 71.50 | 73.68 | 2.17 | -59.36 | 6.05 |
| 11 | Seller | 4065 | 0 | 0.00 | 73.05 | 73.05 | 0.00 | 104.35 | 104.35 |
| 12 | Attacker | 5297 | 0 | 0.00 | - | - | - | 0.00 | 0.00 |

Table 6: The profile of the four types of clients. $\sim U(,)$ denotes the uniform distribution.

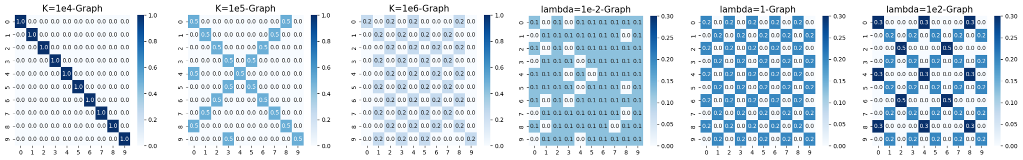
| Type | Trader | Buyer | Seller | Attacker |
|---|---|---|---|---|
| $K_i$ | $\sim U(3e5, 6e5)$ | $\sim U(3e5, 6e5)$ | 0 | 0 |
| $c_i$ | $\sim U(0.5, 1)$ | $+\infty$ | 0 | 0 |

corresponding coefficients, shown in Table 7, and relevant fitting results are shown in Figure 7 and Table 8. The relationship between accuracy and data volume under IID fits well with the equation we modeled in Equation (17). Thus, for the sake of simplicity, we fixed the hyper-parameter $K = p_1^2 \approx 5e5$ in subsequent experiments (K affects the calculation of utility and graph learning).

### E.8 Ablation Study

The ablation experiments are conducted under the Cluster setting on CIFAR-10.

**Effects of $K$ and $\lambda$ on collaboration graph.** In this experiment, we delve into the impact of hyperparameters $K$ and $\lambda$ on the collaboration graph within our federated learning framework. We separately tune the hyper-parameter $K$ with $\lambda = 2, \eta = 5$, $\lambda$ with $K = 1e7, \eta = 5$ under the Skew setting on CIFAR-10 in Figure 8. The heatmaps vividly portray that larger $K$ or smaller $\lambda$ leads to the inclusion of a greater number of clients in the collaboration process.



(a) $K$

(b) $\lambda$

Figure 8: Effects of $K$ and $\lambda$ when selecting collaborators. Larger $K$ or smaller $\lambda$ indicates more clients can be included.

**Effects of local iterations.** Table 9 shows the performance and utility in different local iterations (10, 50, 100, 200, 400, 800). In our experiments for most datasets, we choose iterations equal to 200 regarding the performance and utility.

## F Discussion

In response to the challenges posed by the depletion of publicly available data and the need for collaboration among private institutions, we establish an inclusive sharing market that incentivizes

Table 7: Accuracy (%) V.S. Data size for local training under the uniform data distribution.

| Data size | 50000 | 25000 | 10000 | 5000 | 2500 | 1000 | 500 | 250 |
|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | 62.880 | 62.353 | 59.834 | 55.490 | 48.454 | 40.384 | 34.751 | 29.483 |

Table 8: Fiiting results

| Args | $p_1$ | $p_2$ | $R^2$ | RMSE |
|---|---|---|---|---|
| Value | -698.87 | 66.00 | 0.98275 | 1.8486 |

the contributions of diverse participants with unique model preferences and economic conditions. Rooted in the personalized federated learning paradigm, iPFL integrates a graphical game within the framework based on the directed collaboration graph. iPFL introduces a novel and multifaceted objective, aiming to minimize loss on relevant tasks, maximize pairwise model similarity, and enhance overall social welfare within the system. Our iPFL framework modifies local training methods to achieve improved personalization, flexibly adjusts collaboration regarding models and economic conditions, and implements a sophisticated payment mechanism. The proposed system iPFL facilitates training, collaboration, and transactions to meet each participant's demands and achieves incentive properties theoretically and experimentally. Regarding privacy preservation, our iPFL avoids direct data sharing, ensuring effective data isolation. Regarding the communication overhead, participants in iPFL only need to additionally report their model preference $K_i$, cost $c_i$ and data amount $N_i$ at the start of training. These one-time uploads are negligible for communication but significantly improve the ability to balance model performance and economic utility.

Comprehensive experiments reveal several significant findings about our iPFL. First, iPFL demonstrates exceptional versatility in balancing model performance and economic utility of the AI landscape. Extensive experiments, spanning various machine learning tasks and model scales in Figure 2 and Section 6.1, highlight its capability to achieve comparable or superior model performance and consistently highest social welfare. Second, iPFL ensures individual rationality, as every institution involved in the system achieves non-negative benefits (see Figure 4). This inherent motivation acts as a catalyst, encouraging a growing number of institutions to join the ecosystem. This, in turn, leads to an expansion of the market size, fostering a resilient and extensive database that can further catalyze advancements in AI research. Third, iPFL exhibits a remarkable capability to prevent dishonest practices. Exaggerating data size and cost by participants results in reduced utility (shown in Table 4), acting as a deterrent against dishonest behavior and market fraud. This feature underscores iPFL's commitment to fostering an environment of honesty and integrity. Fourth, iPFL showcases a robust defense against potential attackers. Achieved by effectively isolating malicious participants in Figure 5, iPFL contributes to a stable and trustworthy market environment. In addition, our inclusive simulation experiment in Figure 3 further supports these findings. It demonstrates that honest institutions with distinct needs can acquire what they require, showcasing iPFL as an epitome of an actual healthy market.

Through these advancements, iPFL paves the way for a new era in collaborative AI. With iPFL, institutions can not only benefit from personalized models but also actively contribute to and gain from a flourishing inclusive market while preserving privacy. However, our work also has limitations. Our work assumes the static nature of data, economic needs, and participants' willingness to join during the entire training process. However, in practical scenarios, institutions may choose to exit the training process. For instance, buyers may not require the model for specific tasks or seek more attractive markets. Our framework, designed under the assumption of a static federation, may not fully accommodate such dynamic transformations, especially autonomous exits or joins. Future research could explore more flexible frameworks that adapt to the dynamic states, by adjusting model-sharing strategies, or pricing mechanisms.

Table 9: The average accuracy (%) and utility results in different local iterations.

| Iterations | 10 | 50 | 100 | **200** | 400 | 800 |
|---|---|---|---|---|---|---|
| Acc. | 60.434 | 73.156 | 72.948 | **73.268** | 72.672 | 71.584 |
| Utility | 106.9 | 108.1 | 111.1 | **111.3** | 108.0 | 102.7 |