Chapter 16

Incentives in Federated Learning

Rachael Hwee Ling Lim[a], Sebastian Shenghong Tay[a], Xinyi Xu[a], Yehong Zhang[b], Zhaoxuan Wu[a], Xiaoqiang Lin[a], See-Kiong Ng[a], Chuan-Sheng Foo[c], Patrick Jaillet[d], Trong Nghia Hoang[e], and Bryan Kian Hsiang Low[a]

[a]National University of Singapore, Singapore, [b]Peng Cheng Laboratory, Shenzhen, People's Republic of China, [c]Agency for Science, Technology and Research, Singapore, [d]Massachusetts Institute of Technology, MA, USA, [e]Washington State University, WA, USA

ABSTRACT

This chapter explores *incentive* schemes that encourage clients to participate in federated learning (FL) and contribute more valuable data. Such schemes are important to enable collaboration in competitive situations where clients need justifiable incentives to participate and benefit others with information acquired at significant costs and resources, such as collecting and processing data, computing and communicating model updates, risking the privacy of data via shared model updates. Incentivization addresses these concerns through three key components: (1) *fair contribution evaluation* of each client's data, (2) *client selection* to maximize the utility of the global model, and (3) *reward allocation* to clients. Intuitively, clients desire higher valued rewards which should at least outweigh their costs. These and other requirements will be formally described as incentives. The chapter will also discuss some recent solutions and open problems to achieve these incentives in various settings, which includes settings where the contribution evaluation is declared or measured while the rewards can be monetary- or model-based.

KEYWORDS

Incentives, Fairness, Rationality, Rewards, Welfare

1.1 OVERVIEW AND MOTIVATION

Federated learning requires clients to contribute data and resources and seeks to collaboratively train a global model with higher utility, e.g., validation accuracy. In this chapter, we will discuss *incentives* required to encourage more clients to participate, increase their contribution and address the concerns of the global server (or model owner).

To begin, a key concern in adopting federated learning in practice is that clients might be hesitant to participate considering their significant resources and costs incurred to collect data, compute model updates as well as the risk of losing data privacy while sharing information with the others. For example, a bank may be cautious about collaborating with other organizations as it may leak sensitive information about its customers and business. Furthermore, in most cases, the bank will need a guaranteed profit to participate meaningfully in a contribution: the benefit must outweigh the incurring cost and resources. On the other hand, the global server (or model owner) seeks to maximize the global model utility but may be constrained by a limited budget to compensate the clients. These concerns and desires will be formally described as incentives in Section 1.3.

Incentivization addresses these concerns through three main components: (1) *contribution evaluation*, valuing the (potential) contribution of each client; (2) *client selection*, selecting a subset of potential clients; and (3) *reward allocation* to the clients, deciding the target value of the rewards and realizing the target value by giving out different monetary payments, collaboratively trained models or outputs (such as predictions and generated dataset). Importantly, the non-monetary rewards described above are *freely replicable*: Like digital goods, they can be replicated at zero marginal cost for more clients.

These main components will be discussed in Sections 1.4-1.6. Subsequently, we will discuss how incentives are achieved in the monetary reward (Section 1.8) and the freely replicable non-monetary reward settings (Section 1.9).

1.2 PROBLEM SETTING

In this chapter, we consider a global server (model owner) S and n clients. As described in the previous chapter on data valuation, the utility of any coalition (or their collaboratively trained model) is measured with the same utility function v. Each client m may contribute a resource C_m. The resource C_m can be client m's dataset $\mathcal{D}_m$ in the non-FL setting; the corresponding weight/gradient updates or computational resources used in the FL setting; and predictions on query dataset in the *federated prediction* setting. Simultaneously, each client m expects to receive at least a minimum reward or cost $\chi_m(\cdot)$ in return. Client m's minimum reward can be the utility of the model trained using C_m, i.e., $v(\mathbf{w}^m)$, and cost $\chi_m(C_m, \gamma_m)$ may be client m's total cost to collect data or compute weight updates. Each client may have a different cost function χ_m and cost type γ_m (which can be defined as the cost per unit of C_m). The main goal of incentivization is to get each client m to contribute C_m and increase his contribution, for example, by computing the gradients based on a larger local dataset, removing noise and participating in more iterations.

To achieve incentives outlined in Section 1.3, we require the 3 main components. During *contribution evaluation*, we will assign each client m's contribution C_m a value $\phi(C_m)$ which may depend on the utility function v. During the *reward* *allocation* phase, each selected client's value $\phi(C_m)$ is used to determine his reward value r_m. Sometimes, we need an intermediate step of *client selection* to select a subset of n clients, $\mathcal{S}$, to maximize the utility of the global model, $\mathbf{w}^{\mathcal{S}}$, $v(\mathbf{w}^{\mathcal{S}})$ and ensure that the global server's total budget $B(\cdot)$ can cover the selected clients total rewards, $\sum_{m \in \mathcal{S}} r_m$. The budget $B(\cdot)$ may be fixed or dependent on the aggregated contribution C_m across every client $m \in \mathcal{S}$.

1.3 INCENTIVES

In this section, we will describe what the global server (model owner) and clients intuitively desire as formal properties. In particular, the global server requires:

Feasibility (F). The reward r_m allocated to each client m is limited by the maximum reward available, such as the server's fixed budget $B(\cdot)$.

Truthfulness (T). Each client m should truthfully report information about his cost, $\chi_m(C_m, \gamma_m)$ and his contributions, C_m. For example, it is undesirable if any client m over-reports their data quantity or submits false data to increase their reward r_m and decrease those of others.

Each client m requires:

Privacy (P). Each client m might be concerned about the global server or other clients accessing and inferring its sensitive data through his contribution, C_m. Furthermore, if the global server is untrustworthy or may exclude clients due to its limited budget, each client m risks benefitting others without getting a reward for his contribution and costly effort, e.g., to collect data and compute model weight updates. Under this risk, any client m, such as a bank, may be unwilling to submit its customers' data or other C_m to improve the credit rating or loan predictions for other banks before receiving a reward.

Collaborative Fairness (CF). A client m should *fairly* receive a higher $\phi(C_m)$ than another client j if his contribution C_m is more valuable than client j's C_j such as when m share gradient updates from a larger and more informative dataset. In particular, a free-rider with a zero-valued contribution should get no reward.

Individual Rationality (IR). Each client m must have non-negative profits: The client's benefit must at least balance its costs, i.e. $r_m \geq \chi_m(C_m, \gamma_m)$.

Moreover, to maximally incentivize both clients and server, we should consider:

Group Welfare (GW). The reward scheme should maximize the total welfare, i.e., profits, of the server and all clients. For example, if a client is selected and submits his contribution C_m, his profit will be the reward received less his cost $r_m - \chi_m(C_m, \gamma_m)$. As another example, the server's benefit from the collaboration and profit increases as the utility of the collaboratively trained global model, $\nu(\mathbf{w}^{\mathcal{S}})$, increases.

As an overview, *privacy* and *truthfulness* are addressed during *contribution evaluation*. *Fairness* is addressed during *contribution evaluation* and special care is needed to maintain it during *reward allocation*. To ensure *feasibility*, *individual rationality* and maximize *group welfare* simultaneously, we control the reward value r_m for each client m and consider selecting a subset of clients $\mathcal{S}$. The incentives will be further elaborated as part of each component. Moreover, additional but less common incentives are discussed in Section 1.7.

1.4 CONTRIBUTION EVALUATION

In this section, we will discuss how to assign each client m's contribution, C_m, a value, $\phi(C_m)$, to address the privacy, fairness, and truthfulness incentives.

For **collaborative fairness (CF)**, a client m should *fairly* receive a higher value $\phi(C_m)$ than another client j if his contribution C_m is more valuable than client j's C_j. Similarly, client m's new value $\phi(C_{m'})$ should increase when his new contribution $C_{m'}$ is more valuable than C_m (*strict monotonicity*). See the previous chapter on data valuation for federated learning for a detailed discussion on how existing works define and evaluate more "valuable" formally through a utility function ν. The definition can be as simple as the data quantity or dependent on the FL model (e.g., supervised vs. generative) and validation set(s).

For **privacy (P)**, each client m can limit the extent that the global server and other clients can infer about his dataset $\mathcal{D}_m$ by using *differential privacy* to protect his contribution. [5] consider the setting where each client m decides his own privacy budget and perturbs and protects his contribution C_m by adding random noise before sending C_m to the server.

Moreover, to address the problem that clients might be unwilling to share their contributions before receiving their rewards, the server can require each client to declare the value of $\phi(C_m)$ instead, such as during auctions [2], or estimate $\phi(C_m)$ in round t based on contributions from earlier rounds [3]. However, the lack of access to client m's contribution C_m limits the choice of $\phi(C_m)$. For example, if the model utility ν (e.g., validation accuracy) of coalitions with multiple clients cannot be computed, we cannot use the *Shapley value* to define $\phi(C_m)$.

The **truthfulness (T)** incentive is related to **incentive compatibility (IC)**. A mechanism, such as an auction to decide the reward values, is IC when it is optimal (individually profit-maximizing) for all clients to truthfully declare their cost, $\chi_m(C_m, \gamma_m)$ (or cost type γ_m) and contribution value, $\phi(C_m)$. Separately, to incentivize clients to submit true and high-quality information instead of false and adversarial ones, the server can assign and consider each client's reputation and values in earlier rounds [10] or define $\phi(C_m)$ using the correlation in clients' predictions [8] or model updates [9].

1.5 CLIENT SELECTION

In this section, we will discuss how selecting a subset of clients, $\mathcal{S}$, out of all n clients can address the rationality, group welfare and truthfulness incentive. Client selection is especially important when the server has a limited budget $B(\cdot)$ and cannot afford to pay the total costs across all clients, $\sum_m \chi_m(\cdot)$, due to **feasibility (F)** constraint.

For **individual rationality (IR)**, any client with negative profits, i.e., $r_o < \chi_o(\cdot)$, should be excluded and not selected for the collaboration. For any unselected client o, his cost and reward (hence profits) is zeroed, i.e., $r_o = \chi_o(\cdot) = 0$.

To maximize **group welfare (GW)**, the global server should select clients that increase the global model's utility $v(\mathbf{w}^{\mathcal{S}})$ at a lower total cost to selected clients, $\sum_{m\in\mathcal{S}} \chi_m(\cdot)$. If costs are ignored, the global server can rank each client m based on their contribution value $\phi(C_m)$, defined using utility functions from the previous chapter on data valuation in federated learning. There are more specific strategies to increase group welfare. For example, the work of [15] uses deep reinforcement learning to intelligently choose clients to participate in each round of FL to counterbalance the bias introduced by non-IID data and improve the utility of the global model with fewer communication rounds.

To simultaneously encourage **truthfulness (T) / incentive compatibility**, [10] propose that the global server assigns a *reputation score* for each client based on past validation set performance and selects clients with the highest reputation in each round.

1.6 REWARD ALLOCATION

This section further discusses how to set the reward values $(r_m)_{m\in\mathcal{S}}$ using the contribution value $(\phi(C_m))_{m\in\mathcal{S}}$ to maintain fairness and maximize group welfare. In addition, we also discuss how the value r_m can correspond to monetary rewards or non-monetary rewards. The incentive conditions may differ slightly in the two settings.

For monetary rewards, each selected client m receives monetary payment r_m from the global server's budget $B(\cdot)$ and can optionally get the same global model $\mathbf{w}^{\mathcal{S}}$. A negative r_m implies that client m should pay the global server instead. Using monetary rewards for incentives is convenient to implement: after the payment to each client is decided, it is easy to realize and pay each client the exact amount. However, the global server and clients must agree on the monetary value per unit change in the contribution value $\phi(C_m)$ or the utility function v. The global server's profit is the monetary value of the model less the total monetary payments, $v(\mathbf{w}^{\mathcal{S}}) - \sum_{m\in\mathcal{S}} r_m$. For each selected client m, the profit is the payment received less cost $r_m - \chi_m(C_m, \gamma_m)$. The incentive conditions for monetary rewards that must be simultaneously satisfied are:

Feasibility (F). The reward scheme must ensure **budget balance**: the total monetary rewards should not exceed the budget, i.e., $\sum_{m\in\mathcal{S}} r_m \leq B(\cdot)$.

Individual Rationality (IR). Each client m must be paid a reward r_m which is at least his cost $\chi_m(\cdot)$, i.e., $r_m \geq \chi_m(\cdot)$.

Group Welfare (GW). The total welfare/profit of the server and clients to maximize is $v(\mathbf{w}^{\mathcal{S}}) - \sum_{m\in\mathcal{S}} \chi_m(C_m, \gamma_m)$.

Additionally, to maximize the welfare of clients only and incentivize their participation, the server should fully use the budget, i.e., $B(\cdot) = \sum_{m\in\mathcal{S}} r_m$ (**efficiency**).

For non-monetary rewards, each selected client m will not receive any monetary payment but may receive a different model, $\mathbf{w}^{m,r}$ or additional data $\mathcal{D}^r_m$. Non-monetary rewards may be preferable when there is no available budget to compensate clients (e.g., due to legal restrictions or financial constraints). Moreover, we focus on non-monetary rewards that are *freely replicable*: Like digital goods, we can replicate a model, its outputs or data at zero marginal cost for more clients. This increases the profits for all clients and is hence preferable to monetary rewards setting where increasing the reward r_m (and profit) for client m requires a decrease in reward r_j for another client j when the budget is fully utilized. However, non-monetary rewards might be less convenient to implement:

For each client m, how do we efficiently generate a reward model or data worth some arbitrary value r_m and ensure that a higher reward value r_m would correspond to a higher utility measured by v?

The global server's profit is simply the value of the global model $v(\mathbf{w}^{\mathcal{S}})$. For each selected client m, the profit is the reward value less the minimum expected reward (i.e., cost) $r_m - \chi_m(\cdot)$. If the reward value r_m is defined as the utility of the rewarded model, $v(\mathbf{w}^{m,r})$, it would be more appropriate if the "cost" is similarly defined using the utility. The incentive conditions for non-monetary rewards that must be simultaneously satisfied are as follows:

Feasibility (F). No client m can get more than the most valuable model or dataset derived using the aggregated contribution of clients in $\mathcal{S}$. Formally, if the utility measured with v does not decrease as clients are added, we require $r_m \leq v(\mathbf{w}^{\mathcal{S}})$.

Individual Rationality (IR). Each client m may expect the utility of his rewarded model, r_m, to be at the least the utility the model he can build without participating in FL, i.e., $\chi_m(\cdot) = \nu(\mathbf{w}^m)$.

Group Welfare (GW). The total welfare/profit of the server and clients to maximize is $\nu(\mathbf{w}^{\mathcal{S}}) + \sum_{m\in\mathcal{S}} r_m - \sum_{m\in\mathcal{S}} \chi_m(C_m, \gamma_m)$. As the rewards are *freely replicable*, increasing r_m does not cost the global server and will improve group welfare. Group welfare is maximized when all clients get the most valuable model, e.g., $\mathbf{w}^{\mathcal{S}}$. However, there will be no fairness. A weakened desirable condition is **weak efficiency** — the most valuable client k should fully utilize the contributions, i.e., $r_k = \nu(\mathbf{w}^{\mathcal{S}})$.

To preserve the **collaborative fairness (CF)** incentive, a higher contribution value $\phi(C_m)$ should translate to a higher reward value r_m. This is achieved by the naive solution of rewarding each client m with their contribution value, i.e., $r_m = \phi(C_m)$. However, the **IR** and **efficiency/weak efficiency** conditions may not be satisfied. They can be satisfied when the reward for each client is determined by some monotonically increasing function $r : \phi(C_m) \mapsto r_m \in \mathbb{R}$. If client m has a higher contribution evaluation than client k, client m should have a higher reward. Formally, if $\phi(C_m) > \phi(C_k)$ then $r_m > r_k$. To disincentivize free-riders, a client m with zero-valued contribution, $\phi(C_m) = 0$, should get no reward, $r_i = 0$, thus, we require $r(0) = 0$. A valid function for r is $r(x) = (ax)^\rho$ with $a, \rho > 0, x \geq 0$. Later, we will discuss how these incentives are satisfied for monetary (Section 1.8) and non-monetary (Section 1.9) rewards.

1.7 OTHER INCENTIVES

From existing surveys [14, 17], there may be other requirements for the client selection and reward allocation:

Nash equilibrium [11]. Any client m and the server S cannot *unilaterally* increase their profits, $r_m - \chi_m(C_m, \gamma_m)$ and $\nu(\mathbf{w}^{\mathcal{S}}) - \sum_{m\in\mathcal{S}} r_m$, respectively by changing their decisions, e.g., on C_m and $B(\cdot)$ respectively, when others' decisions are fixed. A Nash equilibrium affords stability and predictability of outcomes as neither any client nor the server has an incentive to change their decision.

Computational efficiency. For resource-constrained scenarios, e.g., FL on mobile devices, it may not be suitable to use computationally costly approaches for contribution evaluation (e.g., Shapley values without approximation), client selection and reward decision and realization.

Robustness to adversarial contributors. The model owner should filter out or discount adversarial updates.

Robustness to replication. A client m cannot increase his total profits or rewards by duplicating himself and participating as more clients such as m'. [4] analyzes existing cooperative game theory solution concepts and proposes a replication-robust reward distribution.

Stability of $\mathcal{S}$. If clients are free to form alternative coalitions and the rewards available depend on the aggregated contribution C_m for $m \in \mathcal{S}$, an additional incentive is that the coalition $\mathcal{S}$ is stable. There must exist some client $m \in \mathcal{S}$ who has no incentive to form another coalition, i.e., all coalitions would not increase client m's profits. [12] discusses the stability of the grand coalition for freely replicable model rewards.

1.8 MONETARY REWARDS

In the basic setting where any client m does not incur costs or expect a minimum reward to participate, i.e., $\chi_m(\cdot) = 0$, we can reward each selected client a share of the global server budget $B(\cdot)$ proportional to its contribution value $\phi(C_m)$. For **efficiency** and **collaborative fairness**, each client m's reward should be $\frac{\phi(C_m)}{\sum_{j \in \mathcal{S}} \phi(C_j)} \times B(\cdot)$. In the more complex setting where each client expects some minimum reward to participate, it is apt to use economic theory solutions to decide the monetary reward r_m for every potential client m and achieve incentives.

When the server and clients have perfect information but binding agreements are not possible (e.g., client m can always alter C_m), *non-cooperative game theory* is appropriate. Each participant optimizes their individual profit while anticipating the actions of other profit-maximizing participants. In the non-cooperative setting, such as the Prisoner Dilemma's game, the resulting solution will be a **Nash equilibrium** that may have lower group welfare than a cooperative outcome. We can consider a simultaneous game or Stackelberg game where the leader, e.g., the model owner, moves first and declares its decision Φ after using backward induction to predict the actions of the followers, e.g., clients as in [5, 18].

When binding agreements are possible (e.g. a legal contract states that enforce a reward function), each client m may be willing to submit their contribution C_m before receiving their reward. We can apply *cooperative game theory* (CGT) [1] to achieve higher group welfare. The most basic and studied form of cooperative games is a *characteristic function game* (CFG) defined by the set of clients and a characteristic function v that maps coalitions (or their contributions) to a value. Note that CFG sets the minimum expected reward $\chi_m(\cdot) = v(\{m\})$ and does not support arbitrarily defining the cost $\chi_m(C_m, \gamma_m)$. The solution to a CFG is a partition of clients into a coalition structure and a reward vector, which distributes the value of each coalition among its members. As CFG implicitly assumes the participation of all clients and that the total reward available $B(\cdot)$ is proportional to $v(\cdot)$, further **client selection** step is needed to ensure **budget balance** when the budget is a limited constant. Solution concepts, such as the Shapley value and the core, can ensure **fairness** and **stability**. However, the Shapley value does not always ensure **individual rationality** for all games/clients and the core may be empty (i.e., no viable solution)s. Moreover, there is no consideration of **truthfulness**. These incentives have to be addressed in additional steps such as through the reward mapping function and future solutions.

Auction theory and *contract theory* are useful tools to incentivize clients to share private information about their data quality and cost types truthfully with the global server (model owner). They explicitly consider **incentive compatibility**, **individual rationality** as constraints, support arbitrarily defined costs and **client selection**. However, both tools may not enforce *fairness* — clients with the same contribution but different declared costs may get different rewards. In a reverse auction, multiple bidders (clients) want to sell their contribution C_m and declare its value $\phi(C_m)$ and cost type γ_m to the buyer (server) with a budget $B(\cdot)$. Auction design involves explicitly setting rules for selecting winning bidders (clients) $\mathcal{S}$ and deciding the monetary payment, $(r_m)_{m\in\mathcal{S}}$ to optimize the *group welfare* while satisfying IC/IR and **budget balance** constraints. For example, *Vickrey auction/mechanism* incentivizes each client m to truthfully share his private and true cost/data quality with the server by setting m's reward r_m only based on the valuation/total profits of *others* and independent of client m's declared values. [2, 3, 6] use auctions to incentivize clients. Alternatively, the server can design specific contracts (which specify the payment for the contribution, i.e., mapping from $\phi(C_m) \mapsto r_m$) for clients with different cost types to maximize the server's profits. Each client m will choose and sign none/one of the provided contracts which ensures its profit is non-negative and maximized. If the client does not contribute C_m, the server can withhold payment. See [7] for a contract theory in FL example.

1.9 NON-MONETARY REWARDS

For freely-replicable non-monetary rewards, such as model and data derivatives, existing CGT literature is largely inapplicable as CGT assumes the constraint of limited rewards. Without this constraint, one can naively decide to allocate the maximum possible reward to all clients to maximize **group welfare**. However, this violates **collaborative fairness** which demands that a client m with a higher contribution value $\phi(C_m)$ get a strictly more valuable reward. This raises the following two questions:

First, how should we decide the reward value to increase **group welfare (GW)** further while still maintaining other desired incentives such as **collaborative fairness (CF)**? Is there a parameter that can control *altruism* and tradeoff between **GW** and **CF** while ensuring **weak efficiency**?

Second, how can we flexibly and efficiently control the reward value? Before considering the FL setting, we will first study some non-FL setting examples to cover some general strategies.

1.9.1 NON-FL SETTING

Existing works have considered rewarding clients with different informativeness of data or model for fairness. Hence, it is natural that the reward value of client m is set to exactly the utility of the non-monetary reward measured using v, i.e., $r_m = v(\cdot)$. However, instead of rewarding client m with $r_m = \phi(C_m)$, using alternatives such as the ρ**-Shapley value** [12] which sets $r_m^* \triangleq (\phi(C_m)/\max_j \phi(C_j))^\rho \times v(\mathbf{w}^{\mathcal{S}}), \rho \in (0, 1]$ and where ϕ is the Shapley value can increase the reward further and is **fairness-preserving** and *weakly efficient*. By selecting a smaller ρ, we increase altruism, **group welfare** and give higher reward values to weaker contributors. Setting $\rho = 0$ assigns the best possible reward $v(\mathbf{w}^{\mathcal{S}})$ to all clients. While this maximizes **group welfare**, **fairness** is lost.

The next challenge is to efficiently achieve the target reward value $(r_m^*)_{m \in \mathcal{S}}$ and some solutions are discussed below:

Model rewards, adding noise to data. In [12], each client m will get a different rewarded model $\mathbf{w}_m^r$ trained on data with additional noise injected to the training labels $\mathbf{y}$. For example, in Bayesian regression models, Gaussian noise of different variance σ_m^2 can be added and optimized by root finding. Higher variance σ^2 reduces the information gained on the model parameters and log-likelihood of the validation set. A more general approach can add noise to other quantities, such as FL gradients.

Synthetic data rewards, controlling number of data samples. In [13], each client m participates in the collaborative training of a generative model and is rewarded with a synthetic dataset drawn from the generative model that augments their original dataset. Stronger contributors will have an augmented dataset with a lower maximum-mean discrepancy (MMD) to a reference data distribution. This reference distribution is approximated with all clients' data together with a large pool of synthetic data. [13] use a modified version of the ρ-Shapley value from [12] to compute rewards that achieve similar incentives such as fairness. The structure of the problem allows a group welfare-maximizing set of parameters to be found with linear optimization. The synthetic reward dataset is generated by greedily sampling synthetic data points from the generative model's data distribution G until the target reward value is reached. The sampling probability of a synthetic data x point is set using the softmax function and proportional to $\exp[\beta \bar{\Delta}_x^m]$ where $\bar{\Delta}_x^m$ is the scaled marginal improvement in v of client m due to x. A larger β will sample points with higher marginal improvement, resulting in higher similarity to G but a smaller synthetic dataset. This sampling algorithm stops when the target reward value is reached.

1.9.2 FL SETTING

During conventional FL, the global server will share the current model weights with each selected client and request the clients to compute weight updates. Thus, if we view trained models as rewards to achieve incentives, conventional FL unfairly gives each selected client the same reward, the latest $v(\mathbf{w}^{S})$ in each iteration, to sustain federated learning. Hence, [16] reward clients with different training-time gradients/ weight updates. A weaker contributor will be rewarded with a more sparsified gradient vector with more components zeroed out. However, this results in clients subsequently reporting weight updates from different locations in the weight space for the server aggregation.

Additionally, if the reward value r_m is exactly the utility of the non-monetary reward measured using v, it may be challenging or inefficient to solve for the weight updates to achieve the desired v, e.g., validation accuracy at every iteration. A simpler approach lets r_m represent a quantity that correlates to higher utility, e.g., the number of unsparsified gradient components. Formally, a client m with aggregated gradients contribution valued at $\phi(C_m)$ will be rewarded with a gradient component that retains the top $r_m = \tanh(\beta\phi(C_m))/\max_j \tanh(\beta\phi(C_j))$ fraction of the components with the largest magnitude. A larger β sparsifies fewer components, leading to less fairness but higher group welfare. β corresponds to an altruism factor. This causes the client's converged model parameters and predictive performance to diverge more from the global server's.

1.10 CONCLUSION AND FUTURE WORK

This chapter gives a preview of incentives in FL, its main components (contribution evaluation, client selection and reward allocation) and some existing works that strive to achieve these incentives using monetary and non-monetary rewards. Next, we briefly describe open problems to be addressed in future work:

First, how can we better achieve the **truthfulness** incentive during contribution evaluation and identify if clients are giving real, high-quality data or contribution C_m? Second, during client selection, instead of maintaining a single coalition of selected clients, $\mathcal{S}$, can clients be partitioned into multiple coalitions to improve group welfare? Last, during reward allocation, what are other ways we can control the non-monetary, e.g. model, reward value in FL? Possible considerations include non-iterative and non-gradient rewards such as the number of rounds participated or updates received.

1.11 ACKNOWLEDGEMENT

This research/project is supported by the National Research Foundation Singapore and DSO National Laboratories under the AI Singapore Programme (AISG

Award No: AISG2-RP-2020-018).

BIBLIOGRAPHY

[1] G. Chalkiadakis, E. Elkind, and M. Wooldridge. Computational aspects of cooperative game theory. In *Synthesis Lectures on Artificial Intelligence and Machine Learning*. 2011.

[2] 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.

[3] 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 *Proc. IEEE INFOCOM*, pages 1–10. IEEE, 2021.

[4] Dongge Han, Michael Wooldridge, Alex Rogers, Olga Ohrimenko, and Sebastian Tschiatschek. Replication robust payoff allocation in submodular cooperative games. *IEEE Transactions on Artificial Intelligence*, 2022.

[5] Rui Hu and Yanmin Gong. Trading data for learning: Incentive mechanism for on-device federated learning. In *Pric. IEEE GLOBECOM*, pages 1–6. IEEE, 2020.

[6] Yutao Jiao, Ping Wang, Dusit Niyato, Bin Lin, and Dong In Kim. Toward an automated auction framework for wireless federated learning services market. *IEEE Transactions on Mobile Computing*, 20(10):3034–3048, 2020.

[7] Jiawen Kang, Zehui Xiong, Dusit Niyato, Han Yu, Ying-Chang Liang, and Dong In Kim. Incentive design for efficient federated learning in mobile networks: A contract theory approach. In *Proc. APWCS*, pages 1–5, 2019.

[8] Yang Liu and Jiaheng Wei. Incentives for federated learning: a hypothesis elicitation approach. *arXiv preprint arXiv:2007.10596*, 2020.

[9] 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 *Proc. WiOpt*, pages 1–8. IEEE, 2021.

[10] Lingjuan Lyu, Yitong Li, Karthik Nandakumar, Jiangshan Yu, and Xingjun Ma. How to democratise and protect ai: Fair and differentially private decentralised deep learning. *IEEE Transactions on Dependable and Secure Computing*, 2020.

[11] John Nash. Non-cooperative games. *Annals of Mathematics*, 54(2):286–295, 1951.

[12] Rachael Hwee Ling Sim, Yehong Zhang, Mun Choon Chan, and Bryan Kian Hsiang Low. Collaborative machine learning with incentive-aware model rewards. In *Proc. ICML*, pages 8927–8936, 2020.

[13] Sebastian Shenghong Tay, Xinyi Xu, Chuan Sheng Foo, and Bryan Kian Hsiang Low. Incentivizing collaboration in machine learning via synthetic data rewards. In *Proc. AAAI*, 2022.

[14] Xuezhen Tu, Kun Zhu, Nguyen Cong Luong, Dusit Niyato, Yang Zhang, and Juan Li. Incentive mechanisms for federated learning: From economic and game theoretic perspective. *IEEE Transactions on Cognitive Communications and Networking*, 2022.

[15] Hao Wang, Zakhary Kaplan, Di Niu, and Baochun Li. Optimizing federated learning on non-iid data with reinforcement learning. In *Proc. IEEE INFOCOM*, pages 1698–1707. IEEE, 2020.

[16] Xinyi Xu, Lingjuan Lyu, Xingjun Ma, Chenglin Miao, Chuan Sheng Foo, and Bryan Kian Hsiang Low. Gradient driven rewards to guarantee fairness in collaborative machine learning. In *Proc. NeurIPS*, volume 34, 2021.

[17] 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.

[18] 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.