Contract Theory Approach to Incentive Mechanism Design for Federated Learning

Tyler Rotello
Colorado School of Mines, Golden, CO, USA

Abstract—Federated learning is a distributed machine learning network where a model is trained across multiple decentralized edge devices or servers holding local data samples, and the model is updated collaboratively without exchanging raw data only model parameters. This makes federated learning promising for a few reasons, primary among them being promoting user privacy without deteriorating model performance, but also adding diverse data sources that could boost model performance. It is thus crucial to encourage participants in the global model to both participate and to provide high quality data to the model. In this paper we aim to design incentive-compatible and individualrational contract mechanism to maximize the utility of the server in the incomplete information model.

I. Introduction

The goal of the Federated Learning model is to, essentially, advance machine learning capabilities while addressing privacy concerns, promoting collaboration, and optimizing the use of decentralized resources. By achieving these objectives, federated learning aims to provide a more privacy-preserving, efficient, and inclusive approach to model training across a distributed network of devices. After collecting sufficient local parameters from the users the model, a global model is created from the aggregate of the local parameters. This will continue until some end goal is reached. This is what enables Federated Learning to increase user-privacy protection without sacrificing model performance.

The decentralized approach leads it to be more efficient due to the prevalence of edge computing, the model training is distributed to the devices at the network edge, reducing reliance on a central server, which leads to lower communication cost, decreased latency and improved efficiency. This decentralized approach makes the model resistant to device failure as one device failing doesn't necessarily affect the overall model training. The model can also be better for personalization as the model is trained on local model characteristics, making it excel in areas such as personal recommendations. The data can also be more robust as it is collected from a diverse audience[1]. All of these benefits are very tangible, but one questions hangs above it all: how to incentive users to contribute high quality data to the model?

This is where the field of Incentive Mechanism Design enters. Participants in federated learning contribute their resources, such as processing power and time, for model training. Incentives help ensure that participants are willing to allocate these resources, which may come with associated costs. If a user doesn't have the proper incentive, they will not contribute to the model as their time would be better spent elsewhere. Participating in the model also leads to a privacy cost to the user, which they must be compensated for. Because Federated Learning is decentralized, unlike centralized approaches, there is no central authority to enforce participation. Incentives play a critical role in motivating participants to contribute resources and collaborate in a decentralized manner. Incentives can also be used to address imbalances in the data provided, to encourage users to provide rarer types of data that can be valuable to the model[1].

In summary, effective incentive structures contribute to the success and sustainability of federated learning ecosystems by encouraging broad and active participation.

The field of incentive mechanism design is an area of growing interest and there have been a number of strategies considered to maximize the ability of the federated learning model. I will briefly describe a few different strategies that were considered to solve the problem.

In Auction Theory there are two players, the model/server acting as the auctioneer and the user/participant acting as the bidder. The server controls the process of the auction and the participants who will respond with their bids. Auction's are a great way for the user to reveal their true type[1].

Shapley Value is another strategy that has been used to evaluate a user's true value to the model. The Shapley Value is a concept from cooperative game theory that assigns a unique value to each player in a cooperative game. The Shapley Value aims to fairly distribute the total payoff among the players based on their contributions. It is computed by considering all possible permutations of the players and calculating the average marginal contribution of each player over all possible orders[1]. The main problem with the Shapley Value is it is exceedingly complex to calculate. There are a number of ways to reduce it, such as the Federated Shapley Value, this is mostly beyond the scope of this paper, interested readers should refer to [2]

The Stackelberg Game is another way to formulate the game, it is appropriate for Incentive Mechanism Design because it is a sequential model of game theory and can be used for the sale of common products. In Federated Learning there are two players, the leader(server) and follower(user), the leader optimizes the profit by considering the reactions of the followers. In the incomplete information model, the computation complexity of NE is increased due to an increase in the number of IC constraints[1].

In this paper I am considering a Contract Theory approach. Contract Theory explores the design and analysis of contracts to facilitate cooperation and exchange between self-interested individuals. In Incentive Mechanism Design, the public procurement is widely used, in this model the server publishes a list of contracts to the participants, and the participants pick the contract that was designed for its' type. A big part of Contract Theory is incomplete information, where there is information asymmetry between parties, in this case the participant has information about itself(its' true type for example) that the model doesn't know. This is where Contract Theory shines, the public procurement strategy embodies a self-revealing property where the server can elicit information about users' types to create optimal provisions. A user, given a list of contracts, chooses the one which will maximize its utility, which should be the one associated with its' type. A central concern with Contract Theory are the Incentive Compatibility(IC) and Individual Rationality(IR) constraints which will be examined later[1].

In this paper I examined a Contract Theory approach to Incentive Mechanism Design that aims to maximize the Server's utility while guaranteeing individual rationality and incentive compatibility constraints. There are some significant simplifications, but it is a solid foundation for future work.

II. Related Work

There has been a lot of interest in the field of Incentive Mechanism Design for Federated Learning and other distributed network paradigms. Federated Learning is itself a very large field, and there are many areas where it is applicable, and so a lot research has been done into it. Many of them, in the simplest form, center around trying to maximize the Server's utility(and thus the model's performance) through some of the methods described above.

For a brief overview, readers interested in Incentive Design using Auction Theory should refer to [3] readers interested in Shapley Value should refer to [2][4][5] those who are interested in Stackelberg Games should view [6][7][8] and finally those curious in Contract Theory should view [9-14] Keep in mind many of these intersect with one another.

As stated above, there is a lot of research being done in this area, and due to the nature of the our timeline, this paper is a simplified version of Incentive Mechanism Design so not much work is directly comparable to what I have done, but my paper is instead a simplified version of other papers and most closely follows how to design a contract from what is laid out in [13].

III. System Model and Game Formulation

This paper considers the scenario where there is one central server and *n* users in the incomplete information model, in this model the server only knows the distribution of information regarding the users.

The server, for every job, will publish the payment p_i and the required data quantity q_i, which will be referred to as a contract (p_i,q_i). It is not reasonable to assume that data quantity is a reliable metric for data quality and thus positive impact on the global mode, and an area for future work will be to evaluate the users data quality using the Federated Shapley Value mentioned above to determine how valuable a user is to a model and to disqualify malicious users. ***The users who select a contract (p_i,q_i) will be referred to as type-*i* users.*** As mentioned earlier in this paper, two requirements for incentive mechanisms that are of vital importance in contracts are *Individual Rationality(IR)* which states: An incentive mechanism is said to be individually rational if it ensures that each participant is motivated to voluntarily participate in the system, given their own preferences and objectives. In other words, an individually rational incentive mechanism provides participants with incentives that make it better for them to participate in the model than to not, meaning the utility from the model is higher than their reserve utility u_i^*. And *Incentive Compatibility(IC)* which states that an incentive mechanism is said to be incentive compatible if it encourages participants to reveal their private information and act in a way that aligns with the goals of the server. In other words, users have an incentive to honestly disclose their private information and engage in a way that supports the models goals. This one is harder to guarantee, but is necessary to ensure that the server is in control.

A. Model

In this model, the aim is to maximize the utility of the sever given incomplete information using contracts that satisfy IC and IR constraints. There are *m* user types, each type is defined only by the data quantity q_i to simplify the problem. The server only knows the distribution of user types, and the probability that a user belongs to type-*i* is λ_i The expected utility of a user of type-i doing task (q_i,p_i) is defined as the difference between the gain from the expected payment p_j and cost of doing task of data quantity q_j

$$U_{i,j} = p_j - c(q_j) \tag{1}$$

I am assuming that the cost function c() is the cost of acquiring data quantity q_j and will just be data quantity q_j so will be removed in the rest of the paper for convenience. This is a simplification, and in future work a cost function could included to be more robust. Based off of this the IR constraint can be defined as:

$$p_i - q_i \geq u^* \tag{2}$$

for a user of type-*i* I assume that a user will elect to participate in the model if $p_i - q_i = u^*$. The IC constraint can be defined as:

$$p_i - q_i \geq p_j - q_j \tag{3}$$

for a user of type-*i* where $1 \leq i\ j \leq m, i /= j$. Essentially saying it is most profitable for a user to be honest(the model doesn't say a user has to be honest, but they have to fulfill their contract). Note the IR constraint is simplified based off of the IC constraint, the contract that provides the maximum value to the user is the one of their type, so only that contract has to be compared to their reserve utility u^*.

The Server utility can now be defined as

$$U_S = \sum_{i=1}^{m} n_i * (v(q_i) - p_i) \tag{4}$$

n_i is the number of users who are of type-*i* and thus select those contracts. v() is a valuation function of the data quantity, which can be any increasing function, I will assume it is $v(q_i) = 3 * q_i$ so that it is more profitable to the model to encourage users to provide more data, other than that it is totally arbitrary. The U_S is what I am aiming to maximize and is subject to the constraints in (2) and (3).

B. Optimization Function

I cannot directly use the the equation in (4) as that assumes we know the number of users n_i of a specific type, which is not true in the incomplete information case. We only posses the distribution of the number of users and the probability that a user belongs to type-*i* as λ_i so the utility of the server should be expressed as an expectation value and use it as the objective function:

$$\mathbb{E}[U_S] = n * \sum_{i=1}^{m} \lambda_i * (v(q_i) - p_i) \tag{5}$$

n is the total number of users as λ_i is a probability and so the summation is scaled for the number of users. Based off this the objective function is

$$\max_{(Q,P)} n * \sum_{i=1}^{m} \lambda_i * (v(q_i) - p_i) \tag{6}$$

which is constrained by the IC and IR constraints in (2) and (3). (Q,P) represents the contracts for the server. We can also simplify the IC constraint by establishing an order of user type based off the data quantity they provide $q_1 \leq q_2 \leq \ldots \leq q_m$. This ordering coupled with an increasing value function means the users receiving the least utility should be providing quantity q_1 which would simplify the constraint to be:

$$p_1 - q_1 \geq u^* \tag{7}$$

This is assuming all users have the same reserve utility which I will assume is u^*=1.

IV. Algorithms

The equation in (6) can now be solved together with constraints laid out in (3) and (7). I will solve it heuristically rather than formatting it as a Stackelberg Game to find the Nash Equilibrium. It is a linear programming problem and will be solved that way.

These factors have been stated elsewhere in the paper, but for convenience I will restate them here. The value of the data to the server is $v(q_i)=3 * q_i$, this function is arbitrarily picked as I am not considering incentive mechanism design for specific use-cases. There are obvious problems with this function, but I will not expand upon them here. The cost to a user for providing data of q_i is considered to be the value q_i, in other terms, it costs them 1 unit of value to provide 1 unit of data quantity. The reserve utility for each user will just be a constant $u^* = 1$. I am also assuming the server perfectly evaluates how much a users data is worth to further simplify, in future work this would not be the case.

The game is then just solved as a linear programming problem in scipy using constraints with various numbers of users to find the optimal contracts. In this scenario it is easy to see that the problem will solve to be such that the profit to the user of selecting any contract will be the same, in other words:

$$p_1 - q_1 = p_2 - q_2 = \ldots = p_m - q_m \tag{8}$$

This is due to the simplifications I have made and the IC and IR constraints. A quick conjecture is given:

$U_{i,i} = p_i - c(q_i) \geq U_{i,j} = p_j - c(q_j)$ (9) where *i* is the users type and *j* is any other type and c() is the function that determines how much it costs for a user to provide q_i. In this case $c(q_i)$ will return a value equal to q_i and so is represented that way. It is important to note that the only thing that determines user type is q_i which is an objective measure and the same for all users, that means coupled with the IC constraint it is easy to see why the equation in (8) is true.

V. Performance Evaluation

From the IC constraint essentially simplifying to (8) and the IR constraint set out, it is easy to see that the cost to the server for every contract, what they will offer to the user, will be $p_i = q_i + u^*$. Essentially the user will make exactly u^* even in the case of incomplete information and dishonest users due to the constraints.

The server is, of course, incentivized to encourage more users of type-*m* as the server's payment function is $v(q_i) = 3 * q_i$ but they will only pay $q_i + u^*$ meaning their profit for a type-*i* user will be $2 * q_i - u^*$. So the server will only provide contracts with $q_i \geq u^*/2$. Of course the server doesn't know the user types, but users also have no incentive to switch from whatever contract they pick as no contract will ever give a better payoff than what they already have(due to the IC constraint). And this is what my results agreed with, every contract gives the user the same payment and the server is encouraged to only provide type-*m* contracts because they are as profitable for every type-*i*

user as a type-*i* contract would be.

In reality the user incurs a cost for revealing their true type, so in this scenario users would actually be encouraged to not pick the contract of their type, but that variable is not included here. An area for future work would be to provide a parameter of that nature and to make the cost function for a

user of type-*i* incur a cost for not providing data of their type as it should be more difficult to acquire, and so would include another descriptor for user type.

VI. Conclusion

This work was very beneficial to provide an introduction to me to the field of Incentive Mechanism Design and was am incredible look into Game Theory being used to solve real problems. The problem I solved was very simple in nature and thus didn't have any substantive findings. A lot of my work was based off [9-14] and specifically [13] so want to give credit there.

In the future I would improve upon this work in a number of ways, some of which I briefly talked about earlier in the paper. I would like to explore using the Federated Shapley Value so the model could accurately evaluate users' data value, this could be used to assign ”standings” to users where some users are considered more truthful and deserve higher incentives, this could also be done to eliminate malicious users. The FSV is complex to calculate so it wouldn't been done to every user every time, but instead randomly and more to ”suspicious” users or those in lower standing somewhat similar to [14]. I also want to increase the robustness of the model by adding more parameters and increasing what the model tracks, I originally had more but it was too complex for the time given. I would also like to more thoroughly examine the problem from the side of auction theory as it appears to be less studied.

References

[1] R. Zeng, C. Zeng, X. Wang, B. Li, and X. Chu, "A Comprehensive Survey of Incentive Mechanism for Federated Learning," *arXiv (Cornell University)*, Jun. 2021.

[2] T. Wang, J. Rausch, C. Zhang, R. Jia, and D. Song, "A Principled Approach to Data Valuation for Federated Learning," *arXiv (Cornell University)*, Sep. 2020, doi: https://doi.org/10.48550/arxiv.2009.06192.

[3] Y. Deng et al., "FAIR: Quality-Aware Federated Learning with Precise User Incentive and Model Aggregation," *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, May 2021, doi: https://doi.org/10.1109/infocom42981.2021.9488743.

[4] A. Ghorbani and J. Zou, "What is your data worth? Equitable Valuation of Data," Jun. 2019.

[5] R. Jia et al., "Towards Efficient Data Valuation Based on the Shapley Value," *International Conference on Artificial Intelligence and Statistics*, vol. 89, pp. 1167–1176, Apr. 2019.

[6] Y. Zhan, P. Li, Z. Qu, D. Zeng, and S. Guo, "A LearningBased Incentive Mechanism for Federated Learning," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6360–6368, Jul. 2020, doi: https://doi.org/10.1109/jiot.2020.2967772.

[7] Y. Sarikaya and O. Ercetin, "Motivating Workers in Federated Learning: A Stackelberg Game Perspective," *IEEE Networking Letters*, pp. 1–1, 2019, doi: https://doi.org/10.1109/lnet.2019.2947144.

[8] S. R. Pandey, N. H. Tran, M. Bennis, Y. K. Tun, A. Manzoor, and C. S. Hong, "A Crowdsourcing Framework for On-Device Federated Learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3241–3256, May 2020, doi: https://doi.org/10.1109/TWC.2020.2971981.

[9] W. Y. B. Lim et al., "Towards Federated Learning in UAV-Enabled Internet of Vehicles: A Multi-Dimensional Contract-Matching Approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 5140–5154, Aug. 2021, doi: https://doi.org/10.1109/tits.2021.3056341.

[10] N. Ding, Z. Fang, and J. Huang, "Incentive Mechanism Design for Federated Learning with Multi-Dimensional Private Information," *Modeling and Optimization in Mobile, Ad-Hoc and Wireless Networks*, pp. 1–8, Jun. 2020.

[11] W. Y. B. Lim et al., "Hierarchical Incentive Mechanism Design for Federated Machine Learning in Mobile Networks," *IEEE Internet of Things Journal*, pp. 1–1, 2020, doi: https://doi.org/10.1109/jiot.2020.2985694.

[12] N. Ding, Z. Sun, E. Wei, and R. A. Berry, "Incentive Mechanism Design for Federated Learning and Unlearning," Oct. 2023, doi: https://doi.org/10.1145/3565287.3610269.

[13] M. Li, J. Lin, D. Yang, G. Xue, and J. Tang, "QUAC: Quality-Aware Contract-Based Incentive Mechanisms for Crowdsensing." Accessed: Dec. 11, 2023. [Online]. Available: http://wiser.arizona.edu/papers/YANG-CNF-MASS2017-1.pdf

[14] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive Mechanism for Reliable Federated Learning: A Joint Optimization Approach to Combining Reputation and Contract Theory," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10700–10714, Dec. 2019, doi: https://doi.org/10.1109/JIOT.2019.2940820.