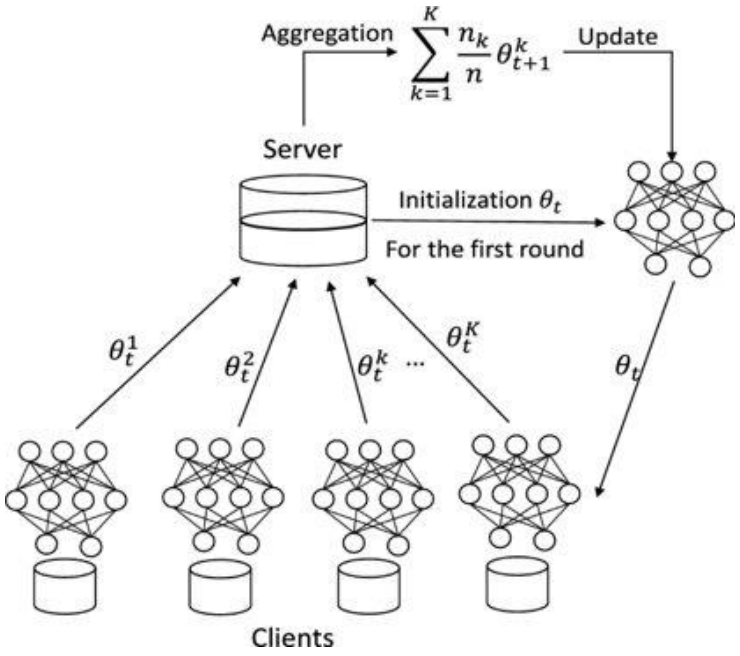
FL @ Client Contribution Evaluation
FL @ Client Selection
FL @ Reward/Incentive

2022_Fall

x

Vanilla Federated Learning



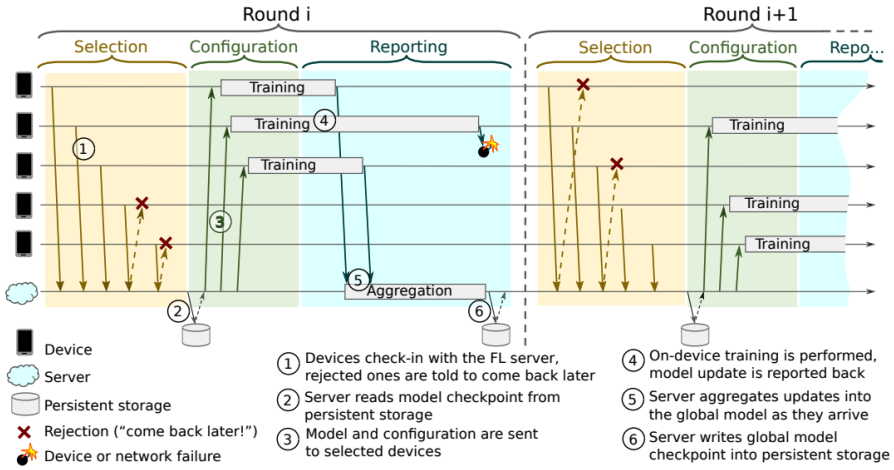


Figure 1: Federated Learning Protocol

Algorithm 1 `FederatedAveraging`. The K clients are indexed by k; B is the local minibatch size, E is the number of local epochs, and η is the learning rate.

Server executes:

initialize w_0

for each round $t = 1, 2, \dots$ **do**

$m \leftarrow \max(C \cdot K, 1)$

$S_t \leftarrow$ (random set of m clients)

for each client $k \in S_t$ **in parallel do**

$w_{t+1}^k \leftarrow \text{ClientUpdate}(k, w_t)$

$w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k$

ClientUpdate(k, w): *// Run on client k*

$\mathcal{B} \leftarrow$ (split $\mathcal{P}_k$ into batches of size B)

for each local epoch i from 1 to E **do**

for batch $b \in \mathcal{B}$ **do**

$w \leftarrow w - \eta \nabla \ell(w; b)$

return w to server

Reference Architecture for FL

FLRA: A Reference Architecture for Federated Learning Systems

FLRA: A Reference Architecture for Federated Learning Systems, https://arxiv.org/abs/2106.11570

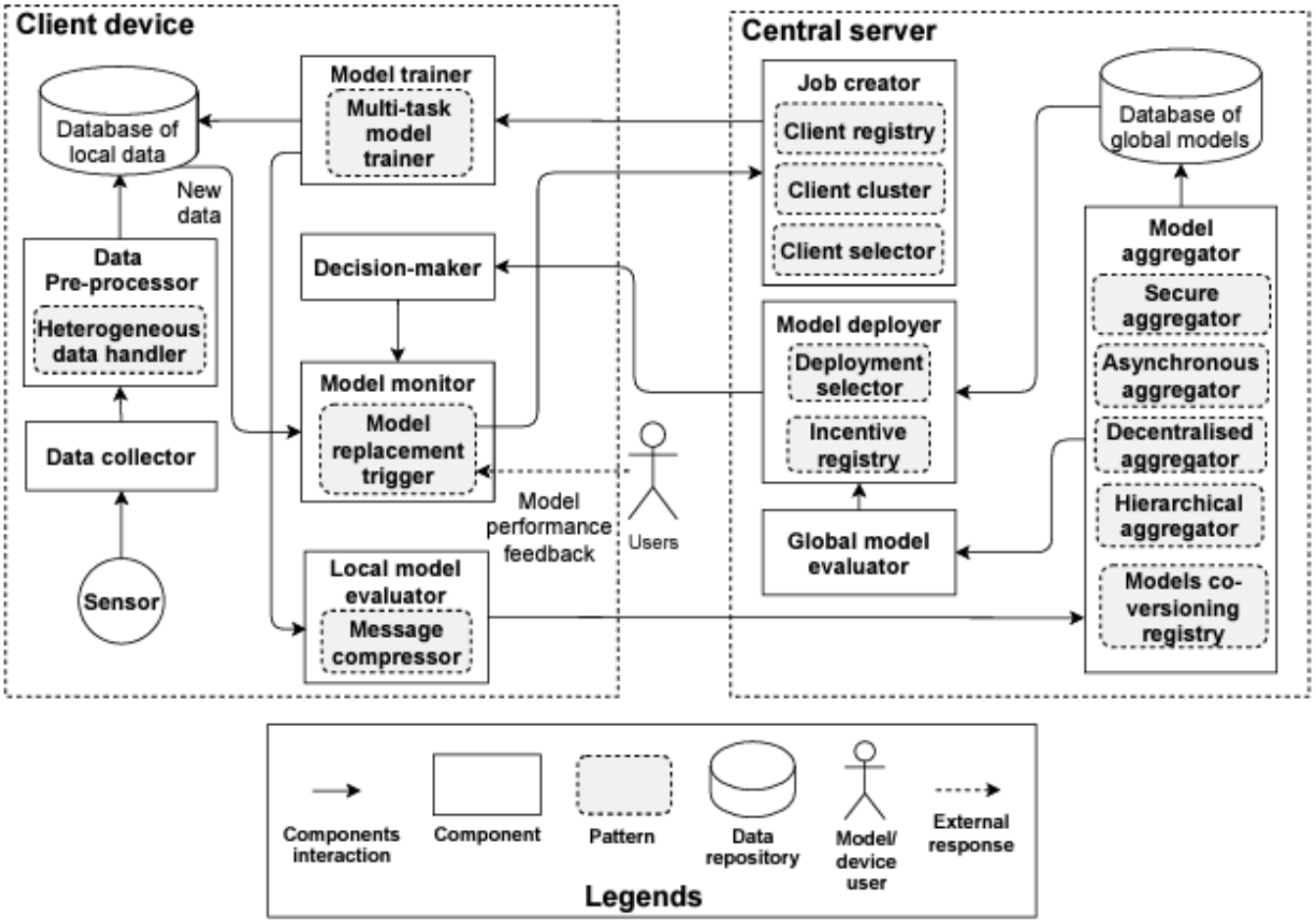


Fig. 3: FLRA: a reference architecture of federated learning systems.

FLRA: A Reference Architecture for Federated Learning Systems

FLRA: A Reference Architecture for Federated Learning Systems, https://arxiv.org/abs/2106.11570

The central servers interacts with a massive number of client devices that are both system heterogeneous and statistically heterogeneous. The magnitude of client devices number is also several times larger than that of the distributed machine learning systems [18,24]. To increase the model and system performance, client devices can be selected every round with predefined criteria (e.g., resource, data, or performance) via **client selector** component.

Table 1: Components of the federated learning reference architecture

Stages	Types	Components	Responsibility
Job creation	Mandatory	Job creator	Initialises training job and global model
	Optional	Client registry	Improves system's **maintainability** and **reliability** by maintaining client's information
		Client cluster	Tackles **statistical heterogeneity** & **system heterogeneity** by grouping clients with similar data distribution or resources before aggregation
		Client selector	Improves **model & system's performance** by selecting high performance client devices
Data collection & preprocessing	Mandatory	Data collector	Collects raw data through sensors or smart devices deployed
		Data preprocessor	Preprocesses raw data
	Optional	Heterogeneous Data Handler	Tackles **statistical heterogeneity** through data augmentation methods

FLRA: A Reference Architecture for Federated Learning Systems

FLRA: A Reference Architecture for Federated Learning Systems, https://arxiv.org/abs/2106.11570

Local model training.
Once the client receives the job from the central server, the model trainer component performs model training based on configured hyperparameters (number of epochs, learning rate, etc.). In the standard federated learning training process proposed by McMahan in [28], only model parameters (i.e., weight/gradient) are mentioned to be sent from the central server, whereas in this reference architecture, the models include not only the model parameters but also the hyperparameters.

Model evaluation.
The **local model evaluator** component measures the performance of the local model and uploads the model to the model aggregator on the central server if the performance requirement is met. In distributed machine learning systems, the performance evaluation on client devices is not conducted locally, and only the aggregated server model is evaluated. However, for federated learning systems, local model performance evaluation is required for system operations such as client selection, model co-versioning, contributions calculation, incentive provision, client clustering, etc.

Model training	Mandatory	Model trainer	Trains local model
		Local model evaluator	Evaluates local model performance after each local training round
		Model aggregator	Aggregates local models to produce new global model

FLRA: A Reference Architecture for Federated Learning Systems

FLRA: A Reference Architecture for Federated Learning Systems, https://arxiv.org/abs/2106.11570

this technique is particularly relevant when faced with nonIIDdata which can produce personalisedmodel that may outperform the best possible shared global model [18]

The conventional design of a federated learning system that relies on a central server to orchestrate the learning process might lead to a single point of failure. **A decentralize aggregator** performs model exchanges and aggregation in decentralized manner to improve system reliability. The known uses of decentralized aggregator include BrainTorrent[31] and FedPGA[15]. Blockchain can be employed as a decentralized solution for federated learning systems.

Optional	Multi-task model trainer	Improves **model performance** (personalisation) by adopting multi-task training methods
	Message compressor	Improves **communication efficiency** through message size reduction to reduce bandwidth consumption
	Secure aggregator	Improves **data privacy** & **system security** through different secure multiparty computation protocols
	Asynchronous aggregator	Improves **system performance** by reducing aggregation pending time of late client updates
	Decentralised aggregator	Improves system **reliability** through the removal of single-point-of-failure
	Hierarchical aggregator	Improves **system performance** & tackle **statistical heterogeneity** & **system heterogeneity** by aggregating models from similar clients before global aggregation
	Model co-versioning registry	Improves system's **accountability** by recording the local models associated to each global models to track clients' performances

FLRA: A Reference Architecture for Federated Learning Systems

FLRA: A Reference Architecture for Federated Learning Systems, https://arxiv.org/abs/2106.11570

The **incentive registry** component maintains all the client devices' incentives based on their contributions and agreed rates to motivate clients to contribute to the training. Blockchain has been leveraged in FLChain[3] and DeepChain[36] to build aincentive registry.

Model deployment	Mandatory	Model deployer	Deploys completely-trained-models
		Decision maker	Decides model deployment
	Optional	Deployment selector	Improves **model performance** (personalisation) through suitable model users selection according to data or applications
		Incentive registry	Increases clients' **motivatability**
Model monitoring	Mandatory	Model monitor	Monitors model's data inference performance
	Optional	Model replacement trigger	Maintains **system & model performance** by replacing outdated models due to performance degrades

Federated Learning Design and Functional Models

Federated Learning Design and Functional Models : Survey,
https://www.researchsquare.com/article/rs 2101865/latest.pdf

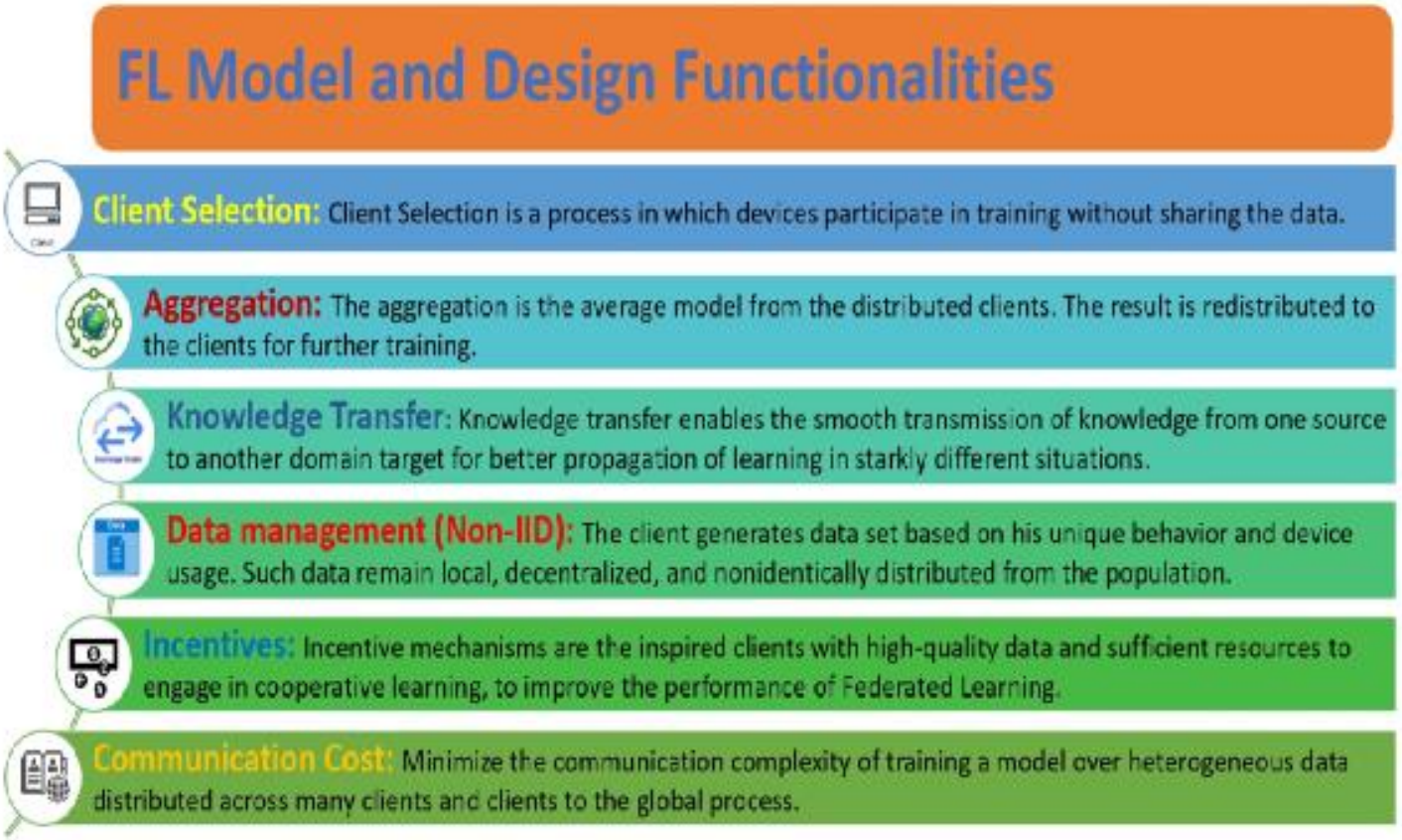


Fig. 1 Functionalities and Design Requirements of Federated Learning

Federated Learning Design and Functional Models

Federated Learning Design and Functional Models : Survey,
https://www.researchsquare.com/article/rs 2101865/latest.pdf

Table 2 Summary of Client Selection methods

S.No	Client Selection Models	Goal	Enviornment and Dataset	Evaluation Parameters
1	Fed CS-Client Selection [34]	Client Selection based on client resource conditions.	MEC, ML Tasks, MNIST, CIFAR-10	Accuracy CIFAR-10 (0.54%), MNIST (74%).
2	MAB-based Client Selection [35]	Client Selection based on the rich throughput and computation cost.	MEC, ML Tasks, CIFAR-10	Learning Rate and Communication round accuracy.
3	Power-of-Choice based Client Selection [41]	Guarantee of FL training with a biased client selection method.	DNN, FMNIST	Communication Round Accuracy (71.2% to 76.5%), Convergence speed, Global Loss.
4	Arbitrary client sampling probabilities [45]	Reduced convergence time.	MNIST and EMNIST	Convergence time, Communication cost and Accuracy.
5	FedAECS [46]	Energy consumption management and balancing the trade-off between accuracy and cost in Edge client learning.	MATLAB-Simulation	Energy Consumption and Accuracy.
6	FedGP [47]	Correlation-based client selection strategy.	FMNIST and CIFAR-10	Communication Overhead.
7	FedCor [37]	Correlation-based client selection strategy.	FMNIST and CIFAR-10	Communication Cost.
8	Multi-Arm Bandit (MAB) [42]	Dynamic selection of clients for improved overall accuracy based on the base learning.	MNIST	Communication Cost.
9	PyramidFL [36]	Achieving a higher final model performance (i.e., time-to-accuracy).	Multiple dataset is used.	Accuracy, Clock Time.
10	FedCorr [39]	Identification of noisy clients and separate measurement of all clients to find incorrect labels based on sample losses.	CIFAR-10/100	Communication Test Accuracy.
11	DRFL [40]	Biased Client Selection to encourage fairness and adjust the wait dynamically.	FMNIST	Fairness.
12	Distributed Client Selection [43]	Client Devices for minimizing overall cost.	CIFAR-10, FMNIST, and MNIST	Communication cost, Trade-Off.
13	Fuzzy logic –Client selection [48]	Client selection based on the quantity of samples, throughput, computational capabilities, and sample freshness.	CIFAR-10	Communication Round Accuracy.
14	FedPNS [49]	Selection of nodes that propel faster model convergence.	CIFAR-10, CIFAR-100	Accuracy over Communication Round.
15	Oort [51]	Use of guided participant for selection and performance based on the time to accuracy and for quick training of clients.	Google Speech, OpenImage and etc.	Communication Rounds and Accuracy.
16	E3CS [52]	Boosting of convergence speed.	EMNIST-Letter and CIFAR-10	Communication Rounds and Accuracy.
17	FedMCCS [53]	Multi-Criteria Approach for client selection based on memory, timing, energy, and client resources.	NSL-KDD	Communication Rounds and Accuracy.

FL @ Client Contribution Evaluation

Client Contribution

[NIPA] 인공지능중심 산업융합 집적단지 조성사업 연구개발 - 헬스케어 AI 융합 연구개발, https://appliedai.skku.edu/appliedailab/ongoing_prj.do?mode=view&articleNo=17788&article.offset=0&articleLimit=10

노이즈가 가미된 연합학습 환경에 대한 클라이언트 기여도 측정 방법의 적합성 평가, https://appliedai.skku.edu/appliedailab/domestic_pub.do?mode=view&articleNo=25863&article.offset=0&articleLimit=10

연합학습은 분산된 환경에서 직접 데이터를 접근하지 않고 각 클라이언트에서 학습한 모델 파라미터를 통합하여 연합 모델을 생성시키는 분산 머신러닝 기술이다. 연합 모델 성능 향상을 위해 연합학습 통합 알고리즘에 대한 연구가 활발하게 진행되고 있는 반면, 클라이언트 기여도 측정 방법 및 클라이언트 제거 기술에 대한 연구도 하나의 연합학습 연구 분야로 급부상하고 있다. 특히, 노이즈가 투입되어 데이터가 훼손될 수 있는 환경에서 '훼손된 클라이언트'(corrupted clients)를 클라이언트 기여도로 선별하는 기술이 필요하다. 본 논문에서는 연합학습에서 클라이언트 기여도 측정으로 기존에 연구된 대표적인 두 가지 방법, Federated LOO와 Federated SV를 소개한다. 이후 이 두 방법이 노이즈가 가미된 연합학습 환경에서 적절하게 작동되는지 노이즈가 투입된 환경에서 실험을 통해 적합성을 평가한다.

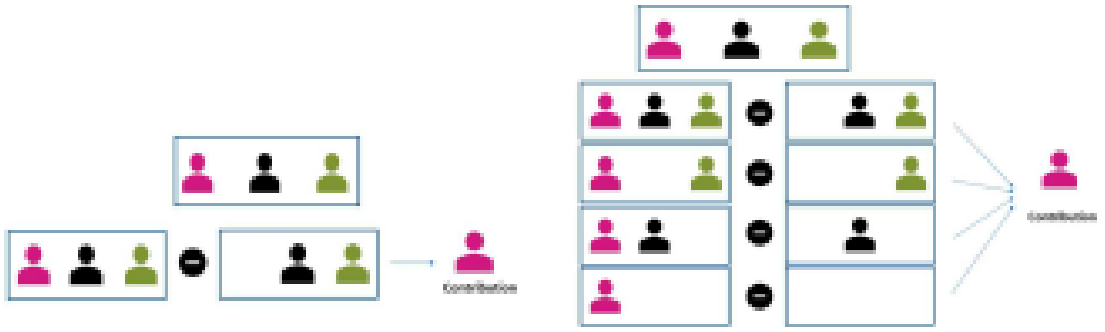


그림 1. 클라이언트 기여도 측정 방법 설명

Client Contribution

Empirical Measurement of Client Contribution for Federated Learning with Data Size Diversification, https://ieeexplore.ieee.org/document/9906094

Client contribution evaluation is crucial in federated learning(FL) to effectively select influential clients. Contrary to data valuation in centralized settings, client contribution evaluation in FL faces a lack of data accessibility and consequently challenges stable quantification of the impact of data heterogeneity. To address this instability of client contribution evaluation, we introduce an empirical method, Federated Client Contribution Evaluation through Accuracy Approximation(FedCCEA), which exploits data size as a tool for client contribution evaluation.

클라이언트 기여 평가는 영향력 있는 클라이언트를 효과적으로 선택하기 위해 연합 학습 (FL) 에서 중요하다. 중앙 집중식 설정의 데이터 평가와 달리 , FL 의 클라이언트 기여 평가는 데이터 접근성의 부족에 직면하여 결과적으로 데이터 이질성의 영향을 안정적으로 정량화하는 데 어려움을 겪는다. 이러한 클라이언트 기여 평가의 불안정성을 해결하기 위해 데이터 크기를 클라이언트 기여 평가 도구로 활용하는 경험적 방법인 FedCCEA (FedCCEA) 를 소개한다.

Client Contribution

Empirical Measurement of Client Contribution for Federated Learning with Data Size Diversification, https://ieeexplore.ieee.org/document/9906094

데이터 중심 접근 방식에서 FL 은 클라이언트 수준에서 기여도를 측정하여 데이터 품질을 고려합니다. 클라이언트 기여는 일반적으로 연합 모델 성능에 대한 각 클라이언트에 대한 데이터 세트의 영향으로 정의됩니다. 고객 기여도 측정은 연합 모델 개선의 두 가지 특정 측면에 적용된다

(i) 클라이언트 선택
딥 러닝 모델의 맥락과 관련하여 , 모든 데이터가 동일한 가치를 가지는 것은 아니다 [5]. 따라서 고품질 및 저품질 데이터를 보존하고 폐기하는 것은 고성능 딥 러닝 모델을 훈련하기 위한 전제 조건이다 [6]. 마찬가 지로 , 모든 클라이언트가 연합 설정에 동일하게 기여하는 것은 아닙니다 [7]—[10]. 영향력 있는 고객을 선택 고 불필요한 고객을 제거하기 위해 이러한 고객을 면밀히 모니터링하고 각 고객의 기여도를 측정해야 합니다.

(ii) 인센티브 할당
경제적으로 , 고객 기여는 이익을 극대화하면서 인센티브를 공정하게 배분하는 데 적합한 표준입니다 [[15]. 고객 기여와 함께 적절한 인센티브 배분은 각 클라이언트의 고품질 데이터 양이 모델 정확도에 영향을 미치는 FL 에 높은 기여자가 적극적으로 참여하도록 동기를 부여할 수 있다 . 이러한 인센티브 메커니즘은 중앙 서버 또는 조정자 에 의한 고성능 연합 모델을 통해 비즈니스 시스템에서 수익과 비용을 효율적으로 관리하는 데 도움이 될 수 있습니다

Then the question is, “how do we evaluate the client contribution in the FL setting?" Unfortunately, a different view from data valuation of centralized learning is required.
그렇다면 질문은 "FL 설정에서 고객 기여도를 어떻게 평가하느냐 는 것이다 . 불행하게도 , 중앙집중식 학습의 데이터 평가와는 다른 관점이 필요하다

Client Contribution

Empirical Measurement of Client Contribution for Federated Learning with Data Size Diversification, https://ieeexplore.ieee.org/document/9906094

Moreover, the impact of data distribution, noise, and data quantity is not as clear as in centralized settings because they strongly rely on combinations with other clients. As shown in Fig. 1 ....
또한 데이터배포, 노이즈 및 데이터 양의 영향은 중앙집중식 설정처럼 명확하지 않습니다. 다른 클라이언트와의 조합에 크게 의존하기 때문입니다. 그림1에 나타난 바와 같이...

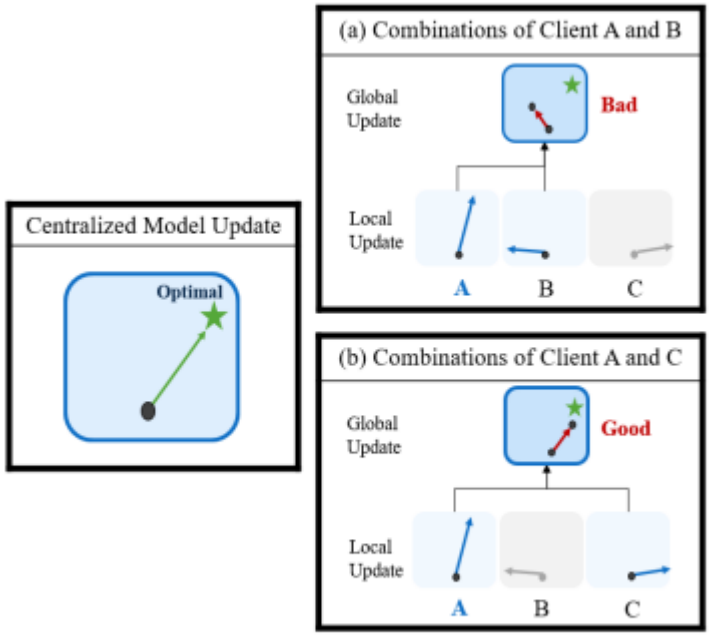


FIGURE 1. Examples of the combinatorial impact of client A in a single round with different combinations.

Client Contribution

Empirical Measurement of Client Contribution for Federated Learning with Data Size Diversification, https://ieeexplore.ieee.org/document/9906094

From early explorations, Shapley Value [8], [21], a game-theoretic evaluation method, predicts the overall combinatorial impact of clients on performance by averaging the marginal test accuracy with all the possible client subsets including and excluding a client as shown in Fig. 2.
Although it is a theoretically well-structured evaluation method, the client contribution measurement by Shapley Value faces challenges with extreme accuracy fluctuations of some combinations in heterogeneous data environments. These drastic combinatorial effects result in unstable client contribution

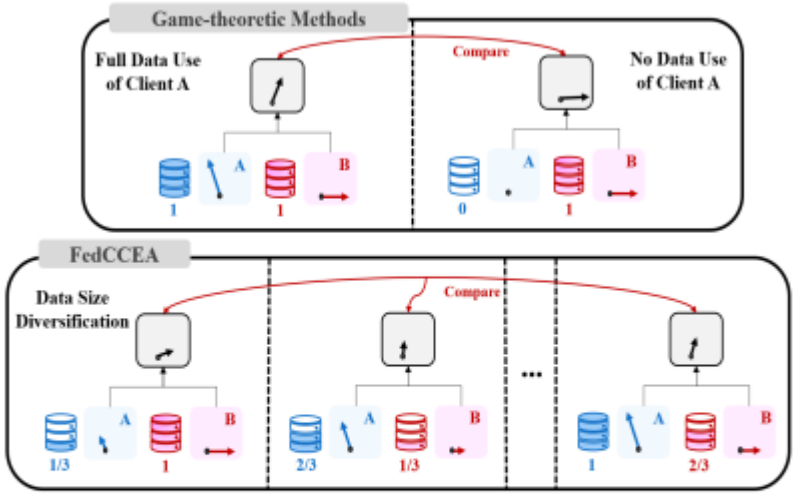


FIGURE 2. Data use cases of each contribution evaluation method while measuring contribution of client A. Regarding the game-theoretic methods, *full or no* data of client A are used to compare the global updates, including and excluding client A. On the contrary, FedCCEA enables several simulations with *data size diversification* to analyze the impact of client A on model performance, while considering various data size sets.

Client Contribution

Empirical Measurement of Client Contribution for Federated Learning with Data Size Diversification, https://ieeexplore.ieee.org/document/9906094

II. RELATED WORKS

A.DATA VALUATION

Data Valuation, a phrase similar toClient Contribution Evaluation, has been widely studied recently to improve centralized machine learning models and to explain black-box predictions.

B. CLIENT CONTRIBUTION EVALUATION FOR FL

In addition to a model-centric approach that focuses on FL optimization [36]–[39], the client contribution evaluation in our study is a data-centric solution to the client-drift problem of FedAvg[2]. The server provides more credit to major clients and fewer credit to minor clients.

TABLE 1. Summary of Client Contribution Evaluation Methods for Federated Learning

Information Used	Keyword	Literature	Description
Local Weights/ Local Gradients	Leave-one-out	[14], [25]	Measure the marginal performance difference of a specific client's participation.
	Shapley Value	[8], [21], [26], [27]	Measure the weighted mean of the marginal performance difference of all possible subsets with a specific client's participation.
	Weight Difference	[28]	Use client contribution based on the directional difference of local weights/gradients for incentive allocation.
	DRL Models	[29], [30]	Empirically predict each client contribution using REINFORCE or DQN models with local weights/gradients.
Local Data Size	Data Quantity	[12]	Simply define a local data size as a total value of each local dataset for incentive allocation.
	FedCCEA	Ours	Empirically predict an averaged impact of each local dataset using deep learning models with diverse cases of data size.

Client Contribution

Empirical Measurement of Client Contribution for Federated Learning with Data Size Diversification, https://ieeexplore.ieee.org/document/9906094

[8] A principled approach to data valuation for federated learning, https://arxiv.org/abs/2009.06192
[21] Data shapley: Equitable valuation of data for machine learning, https://arxiv.org/abs/1904.02868
[25] Toward understanding the influence of individual clients in federated learning, https://ojs.aaai.org/index.php/AAAI/article/view/17263
[28] Incentive mechanism for horizontal federated learning based on reputation and reverse auction, https://dl.acm.org/doi/abs/10.1145/3442381.3449888
[29] Efficient client contribution evaluation for horizontal federated learning, https://ieeexplore.ieee.org/abstract/document/9413377
[30] Optimizing federated learning on non-iid data with reinforcement learning, https://ieeexplore.ieee.org/abstract/document/9155494

TABLE 1. Summary of Client Contribution Evaluation Methods for Federated Learning

Information Used	Keyword	Literature	Description
Local Weights/ Local Gradients	Leave-one-out	[14], [25]	Measure the marginal performance difference of a specific client's participation.
	Shapley Value	[8], [21], [26], [27]	Measure the weighted mean of the marginal performance difference of all possible subsets with a specific client's participation.
	Weight Difference	[28]	Use client contribution based on the directional difference of local weights/gradients for incentive allocation.
	DRL Models	[29], [30]	Empirically predict each client contribution using REINFORCE or DQN models with local weights/gradients.
Local Data Size	Data Quantity	[12]	Simply define a local data size as a total value of each local dataset for incentive allocation.
	FedCCEA	Ours	Empirically predict an averaged impact of each local dataset using deep learning models with diverse cases of data size.

Client Contribution

Empirical Measurement of Client Contribution for Federated Learning with Data Size Diversification, https://ieeexplore.ieee.org/document/9906094

IV. EXPERIMENTS
In this section, we want to answer the following questions:
1) How does accuracy variation occur in the Shapley Value evaluation and how does FedCCEA address this problem?
2) Is FedCCEA evaluation accurate even in the strong nonIIDand noisy environments?
3) Is FedCCEA evaluation accurate even with partial participation?

To answer each question, we design
(i) an accuracy variation comparison,
(ii) a client removal test, and
(iii) experiments for complexity analysis with different numbers of clients.

1) BASELINE EVALUATION METHODS
We answer the above questions and prove the strengths by comparing FedCCEAto the three baseline evaluation methods in recent studies.
• RoundSV[8], [40] is an approximation ofShapley Value in FL. We use the permutation-based RoundSV, utilizing Monte-Carlo sampling for SV approximation.
• Fed-Influence in Accuracy(FIA)[25] is a type of FedInfluencemeasurement metric that simply measures the influence by investigating the effect of removing a client only. The actual FIA value can be obtained from the results of the leave-one-out test.
• RRAFL[28] measures the contribution of cosine similarity between the final global weight vector and current local weight vectors.

Client Contribution

Empirical Measurement of Client Contribution for Federated Learning with Data Size Diversification, https://ieeexplore.ieee.org/document/9906094

E. FURTHER EXPERIMENTS
1) Convergence Analysis for Client Selection
2) Client Removal Test with Different Number of Clients
3) Complexity Analysis

Data shapley

Data shapley: Equitable valuation of data for machine learning, https://arxiv.org/abs/1904.02868

Federated Shapley Value

A principled approach to data valuation for federated learning, https://arxiv.org/abs/2009.06192

Abstract.

Federated learning (FL) is a popular technique to train machine learning (ML) models on decentralized data sources. In order to sustain long-term participation of data owners, it is important to fairly appraise each data source and compensate data owners for their contribution to the training process.

The Shapley value (SV) defines a unique payoff scheme that satisfies many desiderata for a data value notion. It has been increasingly used for valuing training data in centralized learning.

However, computing the SV requires exhaustively evaluating the model performance on every subset of data sources, which incurs prohibitive communication cost in the federated setting. Besides, the canonical SV ignores the order of data sources during training, which conflicts with the sequential nature of FL.

This paper proposes a variant of the SV amenable to FL, which we call the federated Shapley value. The federated SV preserves the desirable properties of the canonical SV while it can be calculated without incurring extra communication cost and is also able to capture the effect of participation order on data value. We conduct a thorough empirical study of the federated SV on a range of tasks, including noisy label detection, adversarial participant detection, and data summarization on different benchmark datasets, and demonstrate that it can reflect the real utility of data sources for FL and has the potential to enhance system robustness, security, and efficiency.

We also report and analyze "failure cases" and hope to stimulate future research.

Federated Shapley Value

A principled approach to data valuation for federated learning, https://arxiv.org/abs/2009.06192

Abstract.

연합 학습(FL)은 분산된 데이터 소스에서 기계 학습(ML) 모델을 훈련하는 인기 있는 기술입니다. 데이터 소유자의 장기적인 참여를 유지하기 위해서는 각 데이터 소스를 공정하게 평가하고 데이터 소유자가 교육 과정에 기여한 것에 대해 보상하는 것이 중요합니다.

Shapley 값(SV)은 데이터 값 개념에 대한 많은 요구 사항을 충족하는 고유한 보수 체계를 정의합니다. 중앙 집중식 학습에서 훈련 데이터를 평가하는 데 점점 더 많이 사용되고 있습니다.

그러나 SV를 계산하려면 데이터 소스의 모든 하위 집합에 대한 모델 성능을 철저하게 평가해야 하므로 연합 설정에서 엄청난 통신 비용이 발생합니다. 게다가, 표준 SV는 훈련 중 데이터 소스의 순서를 무시하는데, 이는 FL의 순차적 특성과 충돌합니다.

이 논문은 연합 Shapley 값이라고 부르는 FL에 순응하는 SV의 변형을 제안합니다. 연합 SV는 표준 SV의 바람직한 속성을 유지하면서 추가 통신 비용을 들이지 않고 계산할 수 있고 데이터 값에 대한 참여 순서의 영향을 포착할 수 있습니다. 우리는 노이즈 레이블 탐지, 적대적 참가자 탐지 및 다양한 벤치마크 데이터 세트에 대한 데이터 요약을 포함한 다양한 작업에 대해 연합 SV에 대한 철저한 경험적 연구를 수행하고 FL에 대한 데이터 소스의 실제 유용성을 반영할 수 있음을 보여줍니다. 시스템 견고성, 보안 및 효율성을 향상시킬 수 있습니다.

또한 "실패 사례"를 보고 및 분석하여 향후 연구를 활성화할 수 있기를 바랍니다.

Federated Shapley Value

A principled approach to data valuation for federated learning, https://arxiv.org/abs/2009.06192

A fundamental question in FL is how to value each data source. FL makes use of data from different entities. In order to incentivize their participation, it is crucial to fairly appraise the data from different entities according to their contribution to the learning process.

For example, FL has been applied to financial risk prediction for reinsurance [1], where a number of insurance companies who may also be business competitors would train a model based on all of their data and and the resulting model will create certain profit. In order to prompt such collaboration, the companies need to concur with a scheme that can fairly divide the earnings generated by the federated model among them.

The SV has been proposed to value data in recent works [6, 10, 11]. The SV is a classic way in coopereative game theory to distribute total gains generated by the coalition of a set of players. One can formulate ML as a cooperative game between different data sources and then use the SV to value data.

Despite the appealing properties of the SV, it cannot be directly applied to FL. By definition, the SV calculates the average contribution of a data source to every possible subset of other data sources.

- Thus, evaluating the SV incurs prohibitive communication cost when the data is decentralized.
- Moreover, the SV neglects the order of data sources, yet in FL the importance of a data source could depend on when it is used for training. For instance, in order to ensure convergence, the model updates are enforced to diminish over time (e.g., by using a decaying learning rate); therefore, intuitively, the data sources used toward the end of learning process could be less influential than those used earlier.

Hence, a new, principled approach to valuing data for FL is needed.

Federated Shapley Value

A principled approach to data valuation for federated learning, https://arxiv.org/abs/2009.06192

FL의 근본적인 질문은 각 데이터 소스를 어떻게 평가하느냐이다. FL은 서로 다른 엔티티의 데이터를 사용합니다. 이들의 참여를 장려하기 위해서는 학습 과정에 대한 기여도에 따라 서로 다른 실체의 데이터를 공정하게 평가하는 것이 중요하다.

예를 들어, FL은 재보험[1]에 대한 재무위험 예측에 적용되었으며, 여기서 사업 경쟁자가 될 수 있는 다수의 보험회사는 모든 데이터를 기반으로 모델을 교육하고 그 결과 모델이 특정 수익을 창출할 것이다. 그러한 협력을 촉진하기 위해서는, 그 회사들은 연합 모델에 의해 창출된 수익을 그들 사이에 공정하게 나눌 수 있는 계획에 동의할 필요가 있다.

SV는 최근 연구에서 데이터의 가치를 평가하기 위해 제안되었습니다 [6, 10, 11]. SV는 협동 게임 이론에서 플레이어들의 연합에 의해 발생하는 총 이득을 분배하는 고전적인 방법이다. ML을 서로 다른 데이터 소스 간의 협력 게임으로 공식화한 다음 SV를 사용하여 데이터를 평가할 수 있습니다.

SV의 매력적인 특성에도 불구하고 FL에 직접 적용할 수는 없다. 정의에 따라 SV는 다른 데이터 소스의 가능한 모든 하위 집합에 대한 데이터 소스의 평균 기여도를 계산합니다.

- 따라서 데이터가 분산될 때 SV를 평가하면 엄청난 통신 비용이 발생한다.
- 더욱이 SV는 데이터 소스의 순서를 무시하지만, FL에서 데이터 소스의 중요성은 훈련에 사용되는 시기에 따라 달라질 수 있다. 예를 들어, 수렴을 보장하기 위해 모델 업데이트는 시간이 지남에 따라 감소하도록 강제됩니다(예: 쇠퇴하는 학습 속도를 사용). 따라서 직관적으로 학습 과정이 끝날 때 사용되는 데이터 소스는 이전에 사용된 데이터 소스보다 영향력이 적을 수 있습니다.

따라서 FL에 대한 데이터 평가에 대한 새롭고 원칙적인 접근법이 필요하다.

FL @ Client Selection

Client Selection

Oort: Efficient Federated Learning via Guided Participant Selection,https://www.usenix.org/conference/osdi21/presentation/lai
https://github.com/Kwangkee/FL/blob/main/FL%40ClientSelection.md#oort

As a result, data characteristics and device capabilities vary widely across clients. Yet, **existing efforts randomly select FL participants, which leads to poor model and system efficiency.** In this paper, we propose Oort to improve the performance of federated training and testing with guided participant selection.

With an aim to improve time-to-accuracy performance in model training, **Oort prioritizes the use of those clients who have both data that offers the greatest utility in improving model accuracy and the capability to run training quickly.**

Unfortunately, clients may not all be simultaneously available for FL training or testing [44]; they may have heterogeneous data distributions and system capabilities [19,38]; and including too many may lead to wasted work and suboptimal performance [19] (§2). **Consequently, a fundamental problem in practical FL is the selection of a "good" subset of clients as participants,** where each participant locally processes its own data, and only their results are collected and aggregated at a (logically) centralized coordinator.

Although random participant selection is easy to deploy, unfortunately,
- it results in poor performance of federated training because of large heterogeneity in device speed and/or data characteristics.
- **Worse, random participant selection can lead to biased testing sets and loss of confidence in results.**

Client Selection

Oort: Efficient Federated Learning via Guided Participant Selection,https://www.usenix.org/conference/osdi21/presentation/lai
https://github.com/Kwangkee/FL/blob/main/FL%40ClientSelection.md#oort

1. Job submission
2. Participant selection:
the coordinator enquires the clients meeting eligibility properties (e.g., battery level), and forwards their characteristics (e.g., liveness) to Oort. Given the developer requirements (and execution feedbacks in case of training 2a),
Oort selects participants based on the given criteria and notifies the coordinator of this participant selection(2b).

3. Execution
4. Aggregation

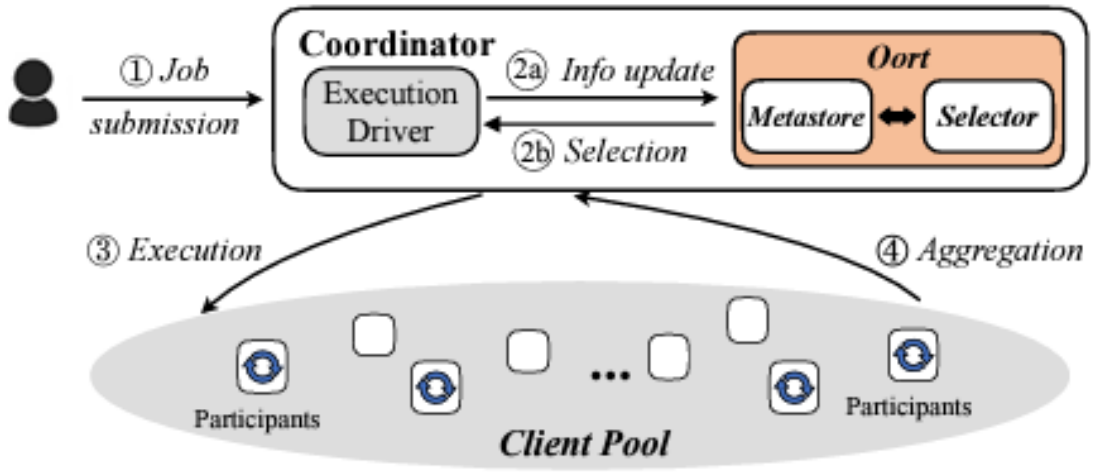


Figure 5: ***Oort architecture. The driver of the FL framework interacts with Oort using a client library.***

Sample Selection

Oort: Efficient Federated Learning via Guided Participant Selection,https://www.usenix.org/conference/osdi21/presentation/lai
https://github.com/Kwangkee/FL/blob/main/FL%40ClientSelection.md#oort

주요아이디어: loss-based statistical utility design
주요아이디어: MAB (Multi-Armed Bandit) problem, exploration-exploitation

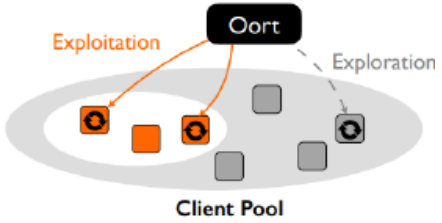
Challenge 1: Identify Heterogeneous Client Utility

- Statistical utility
  - Capture how the client data can help to improve the model
  - Metric: *aggregate training loss* of client data
    - Higher loss ⟶ higher stats utility (proof in paper)
- Utility of a client = $\frac{stats\_util\ (i)}{round\_duration\ (i)}$
  - i.e., *speed* of accumulating stats utility in ***round i***

Heterogeneity Scalability Dynamics Robustness 18

Challenge 2: Select High-Utility Clients at Scale

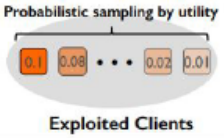
- How to identify high-utility clients from millions of clients?
  - *Spatiotemporal* variation: heterogeneous utility across clients over rounds
- Exploration + Exploitation
  - Explore not-tried clients
  - Exploit known *high-utility* clients



Challenge 3: Select High-Utility Clients Adaptively

- How to account for stale utility since last participation?
  - Utility changes due to dynamics

1. Aging: add uncertainty to utility ⟶ *Re-discover missed good clients*
   - current_utility = last_observed_utility + *observation_age*
2. Probabilistic selection by utility values
   - Prioritize high-utility clients
   - Robust to outliers and uncertainties



Heterogeneity Scalability Dynamics Robustness 21

More in Our Paper

- How to respect privacy
- How to be robust to corrupted clients
- How to enforce diverse selection criteria
  - Fairness, data distribution for FL testing

Heterogeneity Scalability Dynamics Robustness 22

Sample Selection

FedBalancer: Data and Pace Control for Efficient Federated Learning on Heterogeneous Clients (ACM MobiSys2022), https://nmsl.kaist.ac.kr/projects/fedbalancer/
https://github.com/Kwangkee/FL/blob/main/FL%40ClientSelection.md#fedbalancer

Unlike centralized training that is usually based on carefully-organized data, FL deals with on-device data that are often unfiltered and imbalanced. As a result, conventional FL training protocol that treats all data equally leads to a waste of local computational resources and slows down the global learning process.

To this end, we propose FedBalancer, a systematic FL framework that **actively selects clients' training samples.**Our sample selection strategy prioritizes more "informative" data while respecting privacy and computational capabilities of clients. To better utilize the sample selection to speed up global training, we further introduce an adaptive deadline control scheme that predicts the optimal deadline for each round with varying client training data.

For model developers who prototype a mobile AI with**FL without a proxy dataset,** achieving faster convergence on thousands to millions of devices is desired to efficiently test multiple model architectures and hyperparameters [33]. Service providers who frequently update a model with continual learning with FL require to minimize the user overhead with better time-to-accuracy performance [39].

A key objective in FL is to optimize time-to-accuracy performance. FL tasks typically require hundreds to thousands of rounds to converge [13, 38], and clients participating at a round undergo substantial computational and network overhead [21]. Deploying FL across thousands to millions of devices should be done efficiently, quickly reaching the model convergence while not sacrificing the model accuracy. This becomes more important when FL has tobe done multiple times, as often the case when model developers prototype a new model with FL without a proxy dataset or periodically update a deployed model to new domain via continual learning or online learning with FL.

Sample Selection

FedBalancer: Data and Pace Control for Efficient Federated Learning on Heterogeneous Clients (ACM MobiSys2022), https://nmsl.kaist.ac.kr/projects/fedbalancer/

The sample selection of FedBalancerprioritizes more "informative" samples of clients to efficiently utilize their computational effort. **This allows low-end devices to contribute to the global training within the round deadline by focusing on smaller but more important training samples.**To achieve high time-to-accuracy performance, the sample selection is designed to operate without additional forward or backward pass for sample utility measurement at FL rounds. Lastly, FedBalancercan coexist and collaborate with orthogonal FL approaches to further improve performance.

The loss threshold ratio (ltr) enables FedBalancerto start training with all samples and gradually remove already-learned samples. FedBalancerinitialize ltras 0.0 and gradually increasesthe value by loss threshold step size (lss) as shown in Algorithm 3. Note that the deadline ratio ($ddlr$), which controls the deadline of each round (described in Section 3.3), is also controlled with ltr.

FedBalancergradually increase loss threshold to remove already-learned samples
- Round 가 진행될수록, Loss threshold 는gradually increase

The intuition of sampling a portion of data from $UTi$is to avoid catastrophic forgetting [35, 78] of the model on already-learned sample

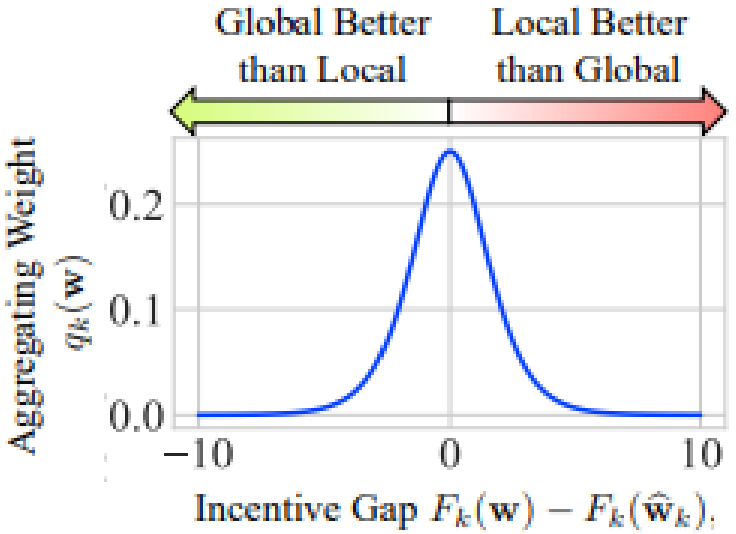
- We sample $L\cdot p$samples from $OTi$and $L\cdot (1-p)$ samples from $UTi$where Lindicates the number of selected samples and $p$is a parameter in an interval of [0.5, 1.0]
- L, the length of selected samples, is determined based on the hardware speed of a client
- p is a FedBalancerparameter between 0.5 ≤ p ≤ 1.0.
- -> p 가 클수록, catastrophic forgetting 을 좀 더 걱정한다는 의미.

Client Selection – Loss based

Yae Jee Cho, https://github.com/Kwangkee/FL/blob/main/FL@CarnegieMellon.md#yae-jee-cho

Towards Understanding Biased Client Selection in Federated Learning,
https://proceedings.mlr.press/v151/jee-cho22a.html
In our work, we present the convergence analysis of federated learning with biased client selection and quantify how the bias affects convergence speed. **We show that biasing client selection towards clients with higher local loss yields faster error convergence.**

To Federate or Not To Federate: Incentivizing Client Participation in Federated Learning,
https://arxiv.org/abs/2205.14840
Figure 2: Aggregating weight qk(w) for any clientkversus the empricalincentive gap Fk(w) −Fk(wbk). The weight qk(w) is small for clients that already have a very large incentive (global much better than local) or no incentive at all (local much better than global), andis highest for clients that are moderately incentivized (global similar tolocal).



FL @ Reward/Incentive

GTG-Shapley

GTG-Shapley: Efficient and Accurate Participant Contribution Evaluation in Federated Learning, https://arxiv.org/abs/2109.02053

Federated Learning (FL) bridges the gap between collaborative machine learning and preserving data privacy. To sustain the long-term operation of an FL ecosystem, it is important to attract high quality data owners with appropriate incentive schemes. As an important building block of such incentive schemes, it is essential to fairly evaluate participants' contribution to the performance of the final FL model without exposing their private data.

Shapley Value (SV)-based techniques have been widely adopted to provide fair evaluation of FL participant contributions.

However, existing approaches incur significant computation costs, making them difficult to apply in practice. In this paper, we propose the Guided Truncation Gradient Shapley (GTG-Shapley) approach to address this challenge. It reconstructs FL models from gradient updates for SV calculation instead of repeatedly training with different combinations of FL participants.

In addition, we design a guided Monte Carlo sampling approach combined with within-round and between-round truncation to further reduce the number of model reconstructions and evaluations required, through extensive experiments under diverse realistic data distribution settings.

The results demonstrate that GTG-Shapley can closely approximate actual Shapley values, while significantly increasing computational efficiency compared to the state-of-the-art, especially under non-i.i.d. settings.

GTG-Shapley

GTG-Shapley: Efficient and Accurate Participant Contribution Evaluation in Federated Learning, https://arxiv.org/abs/2109.02053

연합 학습(FL)은 협업 기계 학습과 데이터 개인 정보 보호 사이의 격차를 메운다. FL 생태계의 장기적 운영을 지속하기 위해서는 적절한 인센티브 계획을 통해 고품질 데이터 소유자를 유치하는 것이 중요하다. 그러한 인센티브 계획의 중요한 구성 요소로서, 개인 데이터를 노출하지 않고 최종 FL 모델의 성능에 대한 참가자의 기여를 공정하게 평가하는 것이 필수적이다.

섀플리 밸류(SV) 기반 기법은 FL 참가자 기여도에 대한 공정한 평가를 제공하기 위해 널리 채택되었다.

그러나 기존 접근 방식은 상당한 계산 비용이 발생하므로 실제로 적용하기 어렵다. 본 논문에서는 이 과제를 해결하기 위해 유도 절단 그레이디언트 섀플리(GTG-Shapley) 접근법을 제안한다. 다양한 FL 참가자 조합으로 반복적으로 훈련하는 대신 SV 계산을 위한 그레이디언트 업데이트에서 FL 모델을 재구성한다.

또한 다양한 현실적인 데이터 분포 설정에서 광범위한 실험을 통해 필요한 모델 재구성 및 평가 수를 더욱 줄이기 위해 라운드 내 및 라운드 간 절단과 결합된 유도 몬테카를로 샘플링 접근 방식을 설계한다.

결과는 GTG-Shapley가 특히 i.i.d.가 아닌 설정에서 최첨단보다 계산 효율성을 크게 높이는 동시에 실제 Shapley 값을 근접하게 근사할 수 있음을 보여준다.

FL @ Medical/Healthcare

[AAAI 2022] CAreFL: Contribution-Aware Federated Learning for Smart Healthcare, https://ojs.aaai.org/index.php/AAAI/article/view/21505

- Hence, the canonical SV cannot be directly used for contribution evaluation in the context of FL.
- The key idea of GTG-Shapley is to opportunistically reduce the need for sub-model retraining with model reconstruction and strategic sampling of combinations of participants. It truncates unnecessary sub-model evaluations to reduce computational costs, while maintaining high accuracy of estimated SVs.

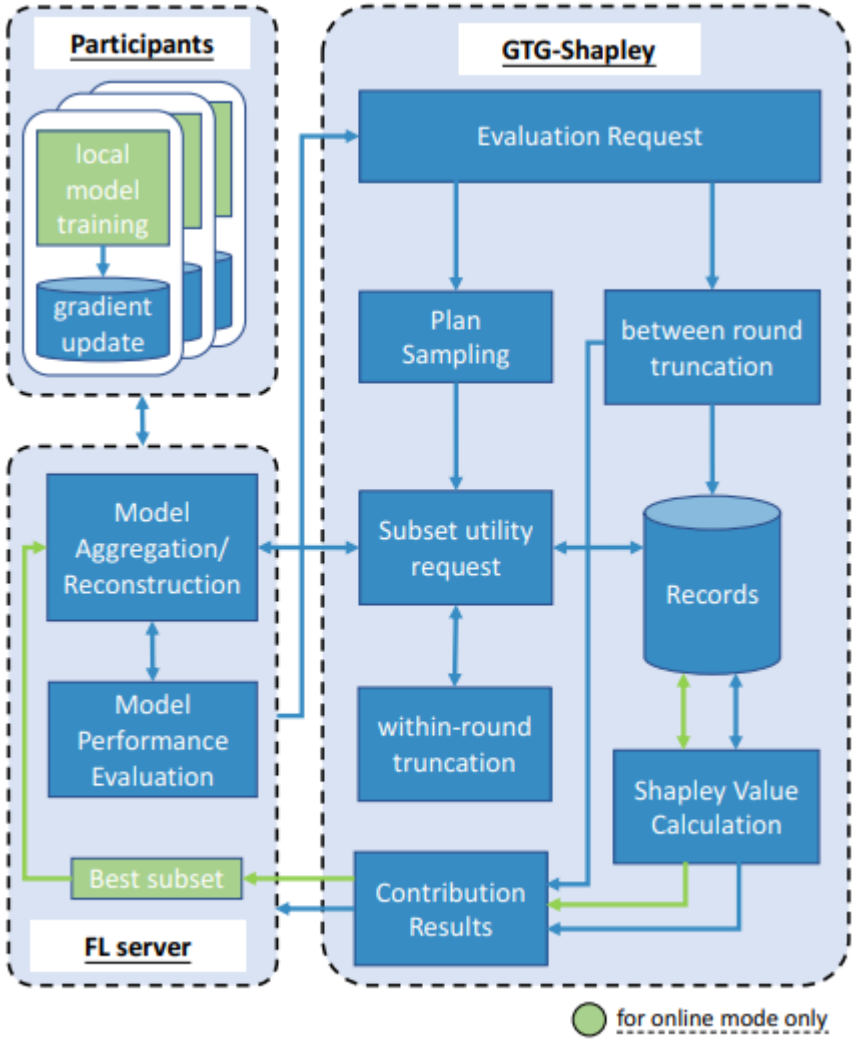


Figure 3: The CAreFL AI Engine

Shapley Value for Data Marketplaces

Private Data Valuation and Fair Payment in Data Marketplaces, https://arxiv.org/pdf/2210.08723

데이터 평가는 데이터 시장에서 필수적인 작업입니다. 데이터 소유자의 기여에 대해 공정하게 보상하는 것을 목표로 합니다. 기계 학습 커뮤니티에서는 협동 게임 이론의 기본 이익 공유 방식인 Shapley 가치가 공정한 신용 할당을 위한 기본 속성을 고유하게 충족하고 모델 성능에 유용하거나 해로운 데이터 소스를 식별합니다.

그러나 Shapley 값을 계산하려면 원본 데이터 소스에 액세스해야 합니다. 개인 정보를 보호하고 공정한 지불을 허용하면서 Shapley 가치 기반 데이터 가격 책정을 활용하는 실제 데이터 시장을 설계하는 방법은 여전히 미해결 문제로 남아 있습니다.

본 논문에서는 Shapley 값을 기반으로 데이터 소스를 개인 정보 보호 방식으로 평가함과 동시에 공정한 지불을 보장하는 데이터 마켓플레이스의 첫 번째 프로토타입을 제안합니다.

우리의 접근 방식은 알고리즘과 시스템 설계 모두에 대한 일련의 혁신을 통해 가능합니다. 먼저 MPC(Multiparty Computation) 회로를 통해 효율적으로 구현할 수 있는 Shapley 값 계산 알고리즘을 제안합니다. 핵심 아이디어는 실제 훈련을 수행하지 않고 입력 데이터 세트에 해당하는 모델 성능을 직접 예측할 수 있는 성능 예측기를 학습하는 것입니다. 성능 예측기의 구조를 기반으로 MPC 회로 설계를 더욱 최적화합니다. 구매자가 지불하는 데이터가 평가된 데이터와 정확히 동일함을 보장하기 위해 공정한 지불을 MPC 회로에 통합합니다. 우리의 실험 결과는 제안된 새로운 데이터 평가 알고리즘이 원래의 값비싼 알고리즘만큼 효과적임을 보여줍니다. 또한 맞춤형 MPC 프로토콜은 효율적이고 확장 가능합니다.

BCFL to transparently evaluate each participant's contribution

Transparent Contribution Evaluation for Secure Federated Learning on Blockchain,
https://ieeexplore.ieee.org/abstract/document/9438754

연합 학습은 여러 당사자가 협력하여 고품질 기계 학습 모델을 구축할 때 유망한 기계 학습 패러다임이다. 그럼에도 불구하고 이들 정당은 기여도에 따른 공정한 포상금 등 충분한 인센티브가 주어질 때만 참여할 의사가 있다.

많은 연구는 학습된 모델에 대한 각 당사자의 기여를 평가하기 위해 Shapley 가치 기반 방법을 탐구했다. 그러나 그들은 일반적으로 모델을 훈련시키고 데이터 소유자의 모델 기여를 평가하기 위해 반신뢰 서버를 가정하는데, 이는 투명성이 부족하고 실제로 연합 학습의 성공을 방해할 수 있다.

본 연구에서는 블록체인 기반 연합 학습 프레임워크와 각 참여자의 기여를 투명하게 평가하기 위한 프로토콜을 제안한다. 우리의 프레임워크는 모델 구축 단계에서 모든 당사자의 개인 정보를 보호하고 모델 업데이트를 기반으로 기여도를 투명하게 평가한다. 손으로 쓴 숫자 데이터 세트에 대한 실험은 제안된 방법이 기여도를 효과적으로 평가할 수 있음을 보여준다.