# Contribution-Aware Federated Learning for Smart Healthcare

**Zelei Liu**[1*], **Yuanyuan Chen**[1*], **Yansong Zhao**[1], **Han Yu**[1†], **Yang Liu**[2†],
**Renyi Bao**[3], **Jinpeng Jiang**[3†], **Zaiqing Nie**[2], **Qian Xu**[4], **Qiang Yang**[4,5†]
[1] School of Computer Science and Engineering, Nanyang Technological University, Singapore
[2] Institute for AI Industry Research, Tsinghua University, Beijing, China
[3] Yidu Cloud Technology Inc., Beijing, China
[4] WeBank, Shenzhen, China
[5] Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong
†han.yu@ntu.edu.sg, liuy03@air.tsinghua.edu.cn, jinpeng.jiang@yiducloud.cn, qyang@cse.ust.hk

## Abstract

Artificial intelligence (AI) is a promising technology to transform the healthcare industry. Due to the highly sensitive nature of patient data, federated learning (FL) is often leveraged to build models for smart healthcare applications. Existing deployed FL frameworks cannot address the key issues of varying data quality and heterogeneous data distributions across multiple institutions in this sector. In this paper, we report our experience developing and deploying the Contribution-Aware Federated Learning (CAreFL) framework for smart healthcare. It provides fair and explainable FL participant contribution evaluation in an efficient and privacy-preserving manner, and optimizes the FL model aggregation approach based on the evaluation results. Since its deployment in Yidu Cloud Technology Inc. in March 2021, CAreFL has served 8 well-established medical institutions in China to build healthcare decision support models. It can perform contribution evaluations 2.84 times faster than the best existing approach, and has improved the average accuracy of the resulting models by 2.62% compared to the previous system (which is significant in industrial settings). To our knowledge, it is the first contribution-aware federated learning successfully deployed in the healthcare industry.

## Introduction

Artificial intelligence (AI) technologies are increasingly intertwined with many aspects of our daily life. For most machine learning approaches, data is at the core of powering their performance. This is especially true for healthcare applications involving AI. In such applications, more high quality data are usually required in order to achieve an acceptable level of performance. However, medical data collected by a single organization (e.g., a hospital) are often not enough for this purpose. Hence, collaborative model training is necessary for this field to benefit from AI technologies (Warnat-Herresthal et al. 2021).

Medical data are often highly sensitive in nature. Thus, data sharing among healthcare institutions has always been a challenge. This is exacerbated by recent data privacy protection laws around the world, such as the General Data Protection Regulation (GDPR) (GDPR 2018). On the flip side, this development has also accelerated the advancement of the field of federated learning (FL) (Yang et al. 2019; Kairouz, McMahan, and et al. 2021), an emerging machine learning paradigm which supports distributed collaborative model training while preserving data privacy. It has been applied in fields from safety management (Liu et al. 2020) to banking (Long et al. 2020).

In recent years, smart healthcare applications powered by FL are starting to emerge (Kaissis et al. 2020; Xu et al. 2020; Sheller et al. 2020; Rieke et al. 2020; Sadilek et al. 2021). These applications generally build on top of the popular Federated Averaging (FedAvg) approach (McMahan et al. 2017) to implement FL across multiple healthcare institutions. While such frameworks are helpful for supporting privacy-preserving collaborative model training, it is less well suited for situations involving data heterogeneity due to non-i.i.d. statistical data distributions across the data silos belonging to different healthcare institutions. In addition, raw healthcare data are often scattered, unstructured, and non-standardized. Thus, data quality across multiple healthcare institutions may vary.

Apart from data issues, the different stakeholders involved in a healthcare ecosystem may have additional needs on top of training useful FL models. For example, a pharmaceutical company may wish to build a model to facilitate drug research by leveraging data from multiple hospitals through FL. In order to compensate the participating hospitals, the pharmaceutical company may need to offer incentive payouts. However, without being able to directly assess the quality of each hospital's local data, fairly compensating them can be challenging (Lyu et al. 2020).

FL participant contribution evaluation is an active subfield of FL (Ghorbani and Zou 2019; Jia et al. 2019; Song, Tong, and Wei 2019; Wang et al. 2020; Wei et al. 2020). The aim is to estimate the value of each FL participant by evaluating its impact on the performance of the resulting FL model, without exposing their sensitive local data. To bridge the aforementioned gaps in FL frameworks for smart healthcare, we propose the *Contribution-Aware Federated Learning (CAreFL)* framework. The advantages are:

1. *Fast and Accurate Contribution Evaluation*: it is incorpo-

[*] These authors contributed equally.

rated with our proposed GTG-Shapley (Liu et al. 2022) approach, which can evaluate fair and accurate FL participant contribution in a highly efficient manner.

2. *Contribution-Aware FL Model Aggregation*: during the contribution evaluation process, GTG-Shapley builds a large number of aggregated FL sub-models involving local model updates from different combinations of FL participants. With this knowledge, CAreFL provides a novel FL aggregation approach which selects the best performing sub-model to be distributed to the FL participants for the next round of local training. This differs from FedAvg-based approaches (which always aggregate all received local models), and can better deal with data heterogeneity issues.
3. *Contribution-based FL Participant Reputation Management*: historical contribution evaluation records are converted into reputation values for the FL participants. This information can serve as a basis stakeholder management decision support.

Compared to existing FL-empowered smart healthcare frameworks, CAreFL offers unique new capabilities which can support more sophisticated use cases.

The CAreFL framework has been deployed through a collaboration between *WeBank*[1] and *Yidu Cloud Technology Inc.*[2] since March 2021. It supports FL model training under server-based horizontal FL settings (Yang et al. 2019), in which participants are from the same domain and their datasets have large overlaps in the feature space but little overlap in the sample space. It has helped eight well-established healthcare institutions in China train AI models for healthcare decision support. It can perform contribution evaluations 2.84 times faster than the best existing approach. Compared to the previous FedAvg-based FL model training approach used by Yidu Cloud, CAreFL achieved a 2.62% model accuracy improvement on average, which is significant in industrial smart healthcare applications. To the best of our knowledge, it is the first contribution-aware federated learning successfully deployed in the healthcare industry.

## Application Description

Yidu Cloud offers smart healthcare solutions with AI technologies. It provides FL model training services to help healthcare customers such as hospitals, pharmaceutical, biotech and medical device companies, research institutions and insurers to build the required models. Its FL service provision framework is based on the opensource Federated AI Technology Enabler (FATE) platform developed by WeBank (Liu et al. 2021). The CAreFL framework is added on top of this infrastructure to provide functionalities related to and derived from FL participant contribution evaluation.

In this section, we provide detailed descriptions of the CAreFL framework. It consists of three tiers (Figure 1): 1) the FL infrastructure tier, which consists of the online FL server (currently training an FL model) and the offline FL server (storing previously trained models), 2) the contribution evaluation (CE) tier, and 3) the user interaction tier.

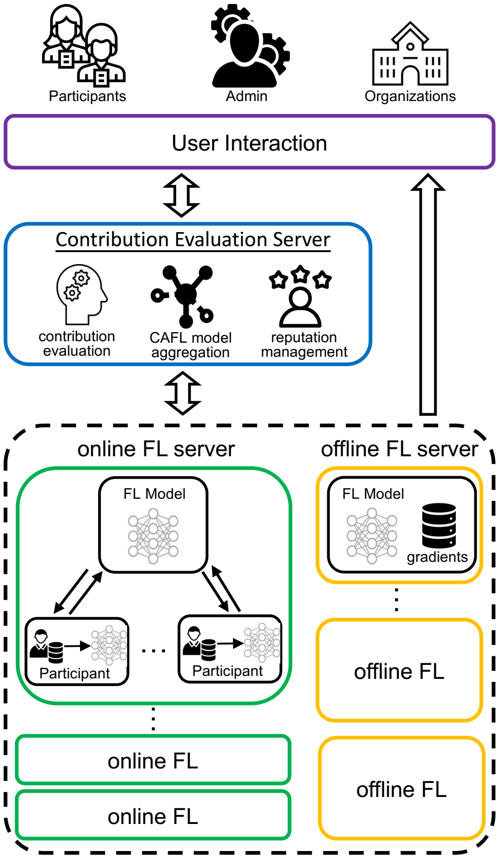


Figure 1: The CAreFL system architecture.

The CE server interacts with FL infrastructure to assess FL participants' contributions. To protect participants' privacy, only their obfuscated IDs are made available to the CE server. The CE server contains the CAreFL model aggregation approach to guide the online FL server as to select the best performing intermediate aggregated model to be distributed to the FL participants for local training. The CE server also tracks the participants' historical contribution evaluation outcomes and updates their reputation scores. The CAreFL process visualization (Wei et al. 2019) and management functions are made available through the user interaction tier for authorized personnel to access. As the AI Engine of CAreFL resides in the CE server, we describe the three main functionalities of the CE server in more details in the following parts of this section.

### Contribution Evaluation

The contribution evaluation workflow of CAreFL is shown in Figure 2. It can be performed in two modes: online and offline, which correspond to the types of federations the system serves. The online mode evaluates participants' contributions during the FL model training process. The offline mode evaluates participants' contributions for FL mod-

[1] https://www.webank.com/

[2] https://www.yiducloud.com.cn/en/

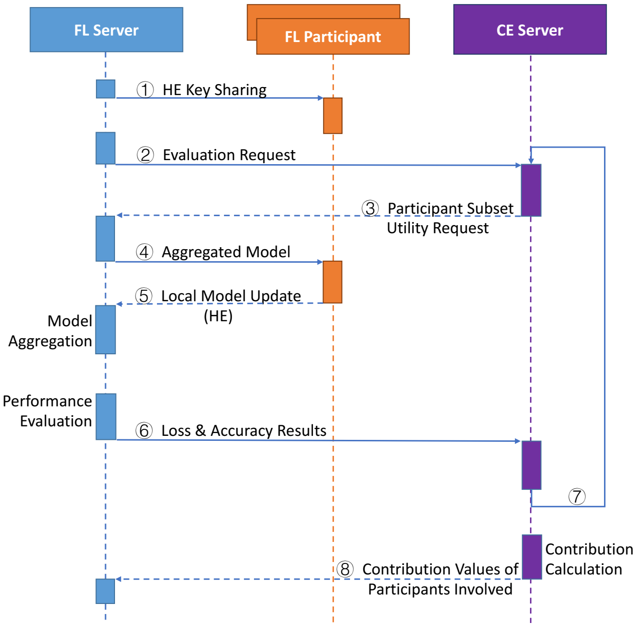


Figure 2: The CAreFL contribution evaluation workflow integrated into the normal FL model training process.

els which have already been built, but with historical local model updates stored in the system. Even though the same evaluation algorithm is used for both modes, the interaction processes are slightly different.

For online FL, the evaluation process is as follows:

1. The FL server initiates an FL session (under the FATE framework, this is done by sending a homomorphic encryption (HE) key to the FL participants).
2. The FL server sends an evaluation request to the CAreFL CE server.
3. The FL server proceeds with FL model training.
4. The FL server sends the obfuscated IDs of FL participants joining the current round of training to the CE server.
5. The CE server sends the combination of participant IDs selected by the proposed GTG-Shapley algorithm (Liu et al. 2022) to the FL server for Shapley Value (SV)-based contribution evaluation.
6. The FL server reconstructs an FL sub-model with only the model updates from the set of participants selected by the CE Server. Then, it evaluates the performance of the sub-model and sends the results to the CE server.
7. The CE server analyzes the received results and adjusts the selected participants set. The new set of participants are sent to the FL server to be evaluated.
8. When the CE server determines that enough information has been gathered, it stops further sub-model evaluation. The contribution values for each participant are sent to FL the server.
9. The FL server proceeds with the next round of training.

For offline FL, the evaluation process is the same as above. However, since the model updates are already stored by the FL server, the offline mode skips the FL training process and uses stored model updates to reconstruct sub-model directly.

## CAreFL Model Aggregation

During the course of contribution evaluation, the performance of different FL sub-models (aggregated using local updates from different combinations of selected FL participants) are calculated as a by product. The CE server can thus identify the sub-model with the best performance. This information is used to guide the online FL server to improve the final FL model performance as follows:

1. After computing the FL participant contribution values for a training round, the CE server returns an additional subset of participants whose local model updates produce the best performing aggregated FL model. This subset can be viewed as the "best subset" of participants for FL model aggregation.
2. With this information, the FL server reconstructs an FL model with the local model updates only from participants in the "best subset".
3. This "best subset" FL model is then distributed to all the participants as the new global FL model for the next round of collaborative training.

## Reputation Management

When starting a new federation, the FL initiator often needs to select a number of participants from the candidate pool. During this process, participants' track records can be useful information to support decision-making. The CAreFL framework includes a reputation management module to facilitate FL participant selection.

CAreFL adopts well-established principles of reputation evaluation from the multi-agent systems literature (Yu et al. 2013; Shen et al. 2011), emphasizing on context-awareness and temporal sensitivity. For example, a participant's data might be more valuable for building an FL model for cancer research than one for bone fracture identification. Moreover, as a participant's data quality depends on its data collection and processing effort, its perceived contribution value in the same application context may also change over time. Hence, the historical contribution evaluation records for each participant are organized according to the application context and the time stamp to facilitate reputation evaluation.

The CAreFL reputation model is designed based on the Beta Reputation System (BRS) (Josang and Ismail 2002). Based on participants' contribution records, it discretizes each record into "good" and "bad" to make it suitable for BRS. If a participant's contribution value is higher than the average contribution value of all the participants in this round, this record is categorized as "good"; otherwise, it is categorized as "bad". CAreFL calculates every participants' reputation under each context based on BRS. A participant's overall reputation is the average of their reputation values over all the contexts they have joined before. As the reputation information is mainly for decision support during manual FL participant selection in the current application, it is not considered part of the CAreFL AI Engine.

## Use of AI Technology

In this section, we describe the AI Engine of CAreFL. It can be divided into two main parts: 1) an efficient Shapley Value (SV)-based participant contribution evaluation algorithm - GTG-Shapley (Liu et al. 2022), and 2) a contribution-aware FL model aggregation algorithm. The system architecture of the AI Engine is illustrated in Figure 3. GTG-Shapley computes the participants' contributions in an efficient manner and returns the results to the FL server. In addition, it also identifies the "best subset" and passes this information to the FL server to improve model aggregation. This function is only relevant for online FL training during which the global FL model is still in the process of being established.

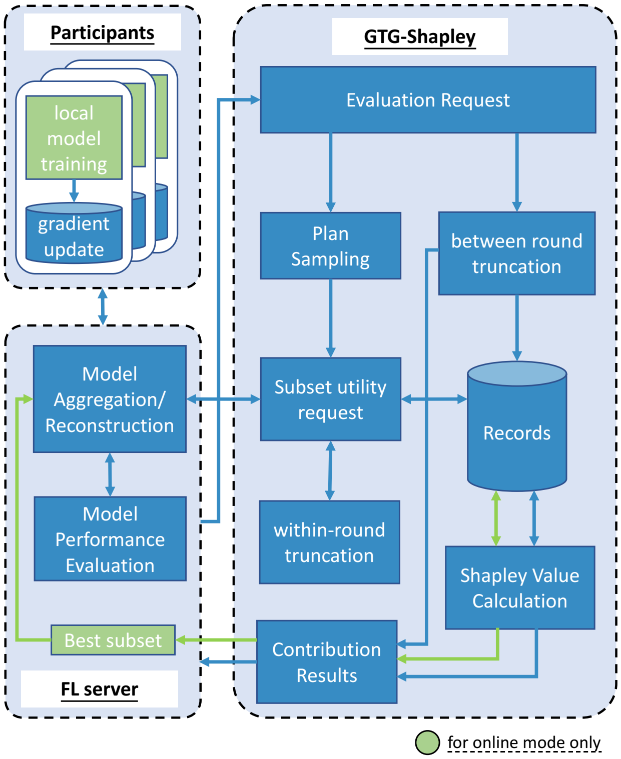


Figure 3: The CAreFL AI Engine

Suppose there are $N = \{1, \dots, n\}$ hospitals, each with a local dataset $D_i, i \in \{1, \dots, n\}$. For a general FL process, there are a total of $T$ collaborative training rounds. During each round $t \in \{1, \dots, T\}$, participant $i$ downloads the global model $M^{(t)}$, and computes a local model $M_i^{(t+1)}$ with its local dataset $D_i$. Then, each participant sends its gradient update $\Delta_i^{(t+1)} = M_i^{(t+1)} - M^{(t)}$ to the FL server. After the FL server has gathered participants' gradient updates, it executes an aggregation approach to obtain a global model $M^{(t+1)}$ for next round of training. The aggregation approach can be any algorithm. For example, if FedAvg (McMahan et al. 2017) is adopted, then:

$$M^{(t+1)} = M^{(t)} + \sum_{i} \frac{|D_i|}{|D_N|} \Delta_i^{(t+1)} \tag{1}$$

where $|D_i|$ denotes the size of dataset $D_i$. $|D_N| = \sum_{i=1}^{n} |D_i|$ denotes the combined size of all $N$ datasets.

### Contribution Evaluation

When evaluating a participant's contribution under the FL paradigm, any direct access to its local data is prohibited. Therefore, the contribution evaluation process must be carried out without examining the actual data. Shapley Value (SV) (Shapley 1953) is a classic approach to fairly quantify the contributions of individuals within a coalition. Moreover, it only requires the final utility achieved by the coalition for calculation. This makes it suitable as a fair contribution evaluation principle for FL.

A participant's SV is the average of all its marginal contributions under all possible permutations within the coalition. The SV, $\phi_i(N, V)$, is expressed as:

$$\phi_i(N, V) = \sum_{S \subseteq N \setminus \{i\}} \frac{V(S \cup \{i\}) - V(S)}{\binom{|N|-1}{|S|}}. \tag{2}$$

$S$ denotes a subset of participants from coalition $N$. The utility function $V(\cdot)$ evaluates the joint utility of the input set. It can be of any form. In machine learning, the utility evaluation function $V(S)$ is based on the performance of the model learned using $S$. That is, $V(S) = V(M_S)$, where $M_S$ is the FL model trained with the subset of participants' datasets $D_S = \{D_i\}, \forall i \in S$:

$$V(S) = V(M_S) = V(\mathcal{A}(M^{(0)}, D_S)) \tag{3}$$

where $\mathcal{A}$ is the learning algorithm and $M^{(0)}$ denotes the initial model.

From Eq. (2), it is obvious that computing the canonical SV takes exponential time with regard to the number of FL participants involved. In practice, FL in the healthcare domain often involves multiple participants (e.g., hospitals). Training the model once on large local data silos is already very time consuming. Furthermore, the sizes of the FL models can be large. Thus, aggregating and evaluating the FL models can also be time consuming. Hence, the canonical SV cannot be directly used for contribution evaluation in the context of FL.

In FL, approaches to accelerate SV calculation have been proposed. (Ghorbani and Zou 2019) samples part of the entire set of possible participant combinations to reduce SV calculation complexity. However, as it still required retraining of the subset FL models, it is still too computationally expensive. Therefore, gradient-based FL model reconstruction approaches (Song, Tong, and Wei 2019; Wang et al. 2020) have been proposed to avoid FL sub-model retraining. These approaches do relieve a large proportion of the SV computational overhead. However, they still require $\mathcal{O}(2^N)$ FL sub-model reconstructions. Thus, they cannot be scaled up to large federations.

In the FL for smart healthcare application scenario, a contribution evaluation solution that can fairly assess participants' contributions in a highly efficient manner is required. Therefore, the CAreFL AI Engine is incorporated with our proposed Guided Truncation Gradient Shapley (GTG-Shapley) approach (Liu et al. 2022). It not only significantly improves computation efficiency, but also achieves higher accuracy compared to the state-of-the-art SV-based FL participant contribution evaluation approaches.

GTG-Shapley is designed to perform contribution evaluations during any given FL training round. To protect participants' privacy, GTG-Shapley only requires participants' IDs as the input. It is worth mentioning that these IDs do not have to be the actual identifiers used by the FL server. The FL server can obfuscate the participants' IDs in any form as long as they are unique. 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.

**Elimination of Sub-Model Retraining** GTG-Shapley leverages the performance information of reconstructed FL models to generate the necessary sub-models. The utility produced by a subset of FL participants represents the collective contributions of participants to the subset. These contributions can be in the form of their local datasets since the FL model is trained on them. Alternatively, it can also be their gradient updates which represent each participant's knowledge derived from its local dataset.

Therefore, when GTG-Shapley determines that the utility of a given subset of participants needs to be evaluated, the FL server reconstructs an FL model based on these participants' model updates, instead of retraining from scratch. In this way, the evaluation of the utility of $S$ no longer involves retraining the sub-model $M_S$. Thus, Eq. (3) can be re-expressed as:

$$V(S) = V(M_S) = V\left(M + \sum_{i \in S} \frac{|D_i|}{|D_S|}\Delta_i\right). \quad (4)$$

**Guided Truncation of Model Evaluations** With the need for sub-model retraining eliminated, the majority of the time required to estimate SVs is incurred by the exponential number of FL sub-model reconstructions and evaluations. For models with a large number of parameters and federations with large test datasets, SV estimation can be significantly slowed down. Thus, it is advantageous to strategically eliminate unnecessary sub-models.

SV estimation in FL differ from traditional cooperation games in two major ways.

1. An FL participant's SV is estimated based on its marginal gain in different participant permutations, and the FL server performs SV utility evaluation. Similar to traditional machine learning, FL shows patterns of diminishing returns both on the marginal gains of within-round participant evaluation orders in a sampled permutation, and the marginal gains across different rounds. A participant can be categorized as "valuable" or "not valuable" based on its marginal gain. Thus, GTG-Shapley only needs to evaluate those "valuable" participants in a sampled permutation. The rest can be omitted without significantly affecting SV estimation.
2. The marginal utility gains concentrate at the leading positions in a given FL participant permutation. Thus, it is important to ensure that participants have equitable opportunities to occupy different positions across multiple permutations in order to evaluate their contributions fairly.

Algorithm 1: GTG-Shapley

**Input**: final FL model $M^{(t)}$'s utility $v_0$, final FL model $M^{(t+1)}$'s utility $v_N$, evaluation request function $V(\cdot)$
**Output**: SVs for round $(t+1)$, $\phi_i^{(t+1)}$, for all $\{i \in \{1, \ldots, n\}$ participants

```
$\phi_i^{(t+1)} = 0, \forall i \in \{1, \ldots, n\}$;
$k = 0$;
# *between round truncation*;
**if** $|v_N - v_0| > \epsilon_b$ **then**
    **while** Convergence criteria not met **do**
        $k = k + 1$;
        $\pi^k$: Partial $(n - m)$ permutation of participants; # *guided sampling*
        $v_0^k = v_0$;
        # *within-round truncation* ;
        **for** $j = 1, \ldots, n$ **do**
            **if** $|v_N - v_{j-1}^k| \geq \epsilon_i$ **then**
                $C = \{\pi^k[1], \ldots, \pi^k[j]\}$;
                $v_j^k = V(M_C^{(t+1)})$;
            **else**
                $v_j^k = v_{j-1}^k$;
            **end if**
            $\phi_{\pi^k[j]}^{(t+1)} = \frac{k-1}{k}\phi_{\pi^k[j]}^{(t+1)} + \frac{1}{k}(v_j^k - v_{j-1}^k)$;
        **end for**
    **end while**
**end if**
**return** $\{\phi_1^{(t+1)}, \ldots, \phi_n^{(t+1)}\}$;
```

To take advantage of these insights, GTG-Shapley is incorporated with a between-round truncation policy to opportunistically skip entire rounds of SV calculation when the remaining marginal utility gain is deemed to be insignificant. In addition, a within-round truncation policy is put in place for opportunistically terminating the ongoing sub-model evaluation when the marginal gain of the remaining FL participants in the current permutation is deemed to be insignificant. Lastly, GTG-Shapley has a guided sampling policy to fairly place FL participants in different positions across multiple participant subsets.

Algorithm 1 shows the details of GTG-Shapley (without the CAreFL model aggregation part). The initial FL model $M^{(t)}$, which was the final FL model from the $(t-1)$-th round, involves an empty set ($S = \{\}$) of participants for the $(t)$-th round of SV calculation. The final model $M^{(t+1)}$ is the global FL model learned during the $t$-th round of training, which includes the full set of participants ($S =$

$\{1, ..., n\}$). Lines 1-2 show parameter initialization. GTG-Shapley performs Monte-Carlo sampling with the truncation policy at two levels. In Line 4, it performs between-round truncation. If the marginal gain of the $t$-th round $|v_N - v_0|$ is not larger than a pre-defined threshold $\epsilon_b$, the entire round $t$ is truncated, and GTG-Shapley returns 0 for every participant in this round of evaluation. Otherwise, GTG-Shapley proceeds to estimate the SVs for the $t$-th round. Line 7 shows the permutation sampling policy of GTG-Shapley. The partial permutation is the proposed guided sampling policy with the following rule: the leading $m$ ($m << n$) bits in the sequence are circulated in a fixed order $P(n, m)$ by the $n$ participants, and the last $(n-m)$ bits are randomly sampled permutations of the remaining participants. This is to avoid unfair SVs estimation and improve convergence. Lines 10-18 show the within-round truncation operation at the sequence level. The utility evaluation for subsequent sub-models in an evaluation sequence can be truncated if the remaining marginal gain is smaller than a pre-defined threshold $\epsilon_i$. Otherwise, the CE server sends a request with an ID list $C$ to the FL server to carry out utility evaluation. After receiving the request, the FL server assembles the FL sub-model $V(M_C^{(t+1)})$ based on participants in $C$, and returns the utility evaluation result to the CE server as shown in Line 13. Lastly, the participants' SVs are updated by their marginal gain in $\pi^k$.

## CAreFL Model Aggregation

During the calculation of SVs, the CE server collects performance information of FL sub-models formed by many alternative combinations of FL participants. In the online mode of operation shown in Figure 2, the CE server selects the subset $S_p \subseteq N$ with the best performance (i.e., the "best subset"). Then, $S_p$ is sent to the FL server. The FL server then aggregates a new global FL model with gradients from participants $\forall i \in S_p$. Thus, the following model aggregation equation is used to replace Eq. (1):

$$M^{(t+1)} = M^{(t)} + \sum_{i \in S_p} \frac{|D_i|}{|D_{S_p}|} \Delta_i^{(t+1)}. \quad (5)$$

# Application Development and Deployment

The CAreFL framework has been developed mainly using the Python programming language by teams from the Joint NTU-WeBank Research Centre on Fintech, Nanyang Technological University (NTU), Singapore, Yidu Cloud Technology Inc., and the Institute for AI Industry Research (AIR), Tsinghua University, China. When developing the AI Engine, we have evaluated seven existing SV-based FL participant contribution evaluation approaches. They are:

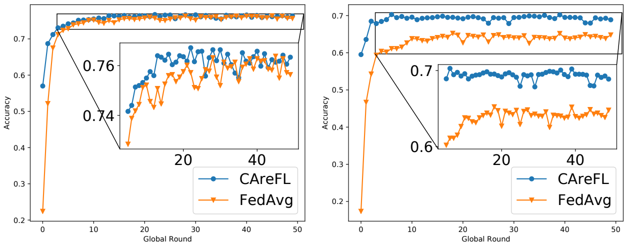
1. **Canonical SV**: This method follows the canonical SV calculation according to Eq. (2).
2. **TMC Shapley** (Ghorbani and Zou 2019): Utility evaluation of a subset involves re-training a sub-model with participants' local datasets. Monte-Carlo estimation of SVs is performed by sampling random participant permutations and truncating unnecessary sub-model utility evaluations.
3. **Group Testing** (Jia et al. 2019): It samples a number of subsets of FL updates and evaluates the corresponding sub-model utility. Then, it estimates the Shapley differences instead of SVs. Later, it infers SVs by solving a feasibility problem with the Shapley differences.
4. **MR** (Song, Tong, and Wei 2019): The utility of a subset is evaluated by reconstructing the FL sub-model with gradient updates. The SV of each participant is calculated according to Eq. (2). The final SV for a participant is the sum of its SVs in all rounds.
5. **Fed-SV** (Wang et al. 2020): It approximates the "federated Shapley value" via group testing-based estimations. The differences are: 1) the subset utility values used for estimating the Shapley differences is evaluated based on reconstructed sub-models; and 2) SVs are estimated independently each round and aggregated in the end.
6. **TMR** (Wei et al. 2020): SVs are calculated independently in each round with reconstructed FL sub-models, with a decay parameter $\lambda$ which serves as: 1) weights to amplify SVs from earlier rounds; and 2) a truncation factor to eliminate unnecessary sub-model reconstructions.
7. **GTG-Shapley** (Liu et al. 2022): our proposed approach.

To compare the contribution evaluation performance of these approaches under different FL settings, we designed i.i.d. and non-i.i.d. FL scenarios involving 10 participants. The datasets used in the experiments are derived from the MNIST dataset (LeCun, Cortes, and Burges 2010). The results in terms of the time duration taken and the accuracy of the contribution evaluation (measured by the Euclidean Distance (ED) between the estimated SVs and the value computed by the Canonical SV approach) are shown in Table 1. In the table, both metrics are presented in $\log_{10}$ scale. It can be observed that GTG-Shapley consistently achieves the highest efficiency and accuracy under both i.i.d. and non-i.i.d. settings. The results helped the design team to make the decision to select GTG-Shapley for the task of FL participant contribution evaluation in CAreFL.

| | i.i.d | | non-i.i.d | |
|---|---|---|---|---|
| | Duration | ED | Duration | ED |
| Canonical SV | 4.615 | - | 4.615 | - |
| MR | 3.833 | -2.35 | 3.733 | -2.148 |
| TMC | 4.168 | -1.687 | 4.213 | -1.369 |
| TMR | 3.531 | -2.353 | 3.678 | -2.27 |
| GroupTesting | 4.583 | -0.894 | 4.557 | -0.667 |
| Fed-SV | 3.784 | -0.757 | 3.711 | -0.789 |
| GTG-Shapley | **2.662** | **-2.427** | **2.733** | **-2.323** |

Table 1: Contribution evaluation experiments on MNIST.

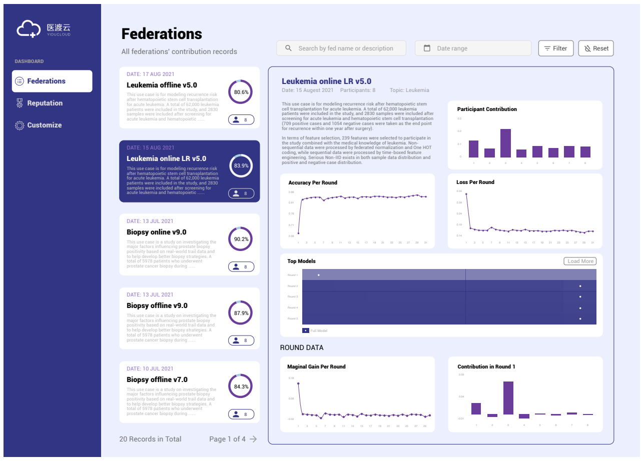
Furthermore, we experimentally compared CAreFL model aggregation with FedAvg, which is part of the FATE framework Yidu Cloud uses, under the same i.i.d. and non-i.i.d. FL scenarios with 10 participants. The datasets used in the experiments are derived from the CIFAR-10 dataset (Krizhevsky 2009). The results are illustrated in Figures 4(a) and 4(b). It can be observed that CAreFL outperforms Fe-



(a) i.i.d. case

(b) non-i.i.d. case

Figure 4: Model aggregation experiments on CIFAR-10.



Federations

All federations' contribution records

DATE: 17 AUG 2021
Leukemia offline v5.0
This use case is for modeling recurrence risk after hematopoietic stem cell transplantation for acute leukemia. A total of 62,000 leukemia patients were included in the study, and 2830 samples were included after screening for acute leukemia and hematopoietic ......

DATE: 15 AUG 2021
Leukemia online LR v5.0
This use case is for modeling recurrence risk after hematopoietic stem cell transplantation for acute leukemia. A total of 62,000 leukemia patients were included in the study, and 2830 samples were included after screening for acute leukemia and hematopoietic ......

DATE: 13 JUL 2021
Biopsy online v9.0
This use case is a study on investigating the major factors influencing prostate biopsy positivity based on real-world trail data and to help develop better biopsy strategies. A total of 5978 patients who underwent prostate cancer biopsy during ......

DATE: 13 JUL 2021
Biopsy offline v9.0
This use case is a study on investigating the major factors influencing prostate biopsy positivity based on real-world trail data and to help develop better biopsy strategies. A total of 5978 patients who underwent prostate cancer biopsy during ......

DATE: 10 JUL 2021
Biopsy offline v7.0
This use case is a study on investigating the major factors influencing prostate biopsy positivity based on real-world trail data and to help develop better biopsy strategies. A total of 5978 patients who underwent prostate cancer biopsy during ......

20 Records in Total     Page 1 of 4

Leukemia online LR v5.0
Date: 15 Augest 2021     Participants: 8     Topic: Leukemia

This use case is for modeling recurrence risk after hematopoietic stem cell transplantation for acute leukemia. A total of 62,000 leukemia patients were included in the study, and 2830 samples were included after screening for acute leukemia and hematopoietic stem cell transplantation (709 positive cases and 1054 negative cases were taken as the end point for recurrence within one year after surgery).

In terms of feature selection, 239 features were selected to participate in the study combined with the medical knowledge of leukemia. Non-sequential data were processed by federated normalization and One HOT coding, while sequential data were processed by time-boxed feature engineering. Serious Non-IID exists in both sample data distribution and positive and negative case distribution.

ROUND DATA

Figure 5: The main user interface of CAreFL.

dAvg under both settings. The outperformance is more pronounced under the non-i.i.d. setting, which is more prevalent in healthcare data silos. The results helped the design team to make the decision to incorporate this new FL aggregation approach into CAreFL.

Figure 5 shows the user interface through which CAreFL visualizes FL participant contribution evaluation for the system administrators. The left sidebar enables switching between the three perspectives. On the main page of "Federations", the left-hand side illustrates different federations as clickable cards with brief overview information, such as federation name, task description, creation time, number of participants and FL model performance. The list of federations displayed can be refined with the filters on the top. The right-hand portion of the main page shows key information about participant contribution and CAreFL model aggregation for the selected federation (highlighted in dark blue). At the top, textual descriptions about the selected federation is displayed to the left, while the evaluated contributions of the participants are displayed to the right. The training accuracy and loss of the aggregated FL model over time are shown below them. Underneath, a grid heat map of FL sub-model utility values during each round of contribution evaluation is plotted. Each row represents one training round. FL models consisting of all participants' model updates are highlighted with a white dot. In each row, the sub-models are arranged in descending order of their performance from left to right. When clicking on a cell, a box will pop up with more detailed information about model performance and the IDs of the participants involved. Lastly, at the bottom, the marginal gains of each round of evaluation are displayed to the left. When clicking on a round this chart, the contributions of all FL participants involved in it are plotted as a barchart to the right. It should be noted that the actual deployed system user interface is in Chinese. Figure 5 has been translated for English speaking readers.

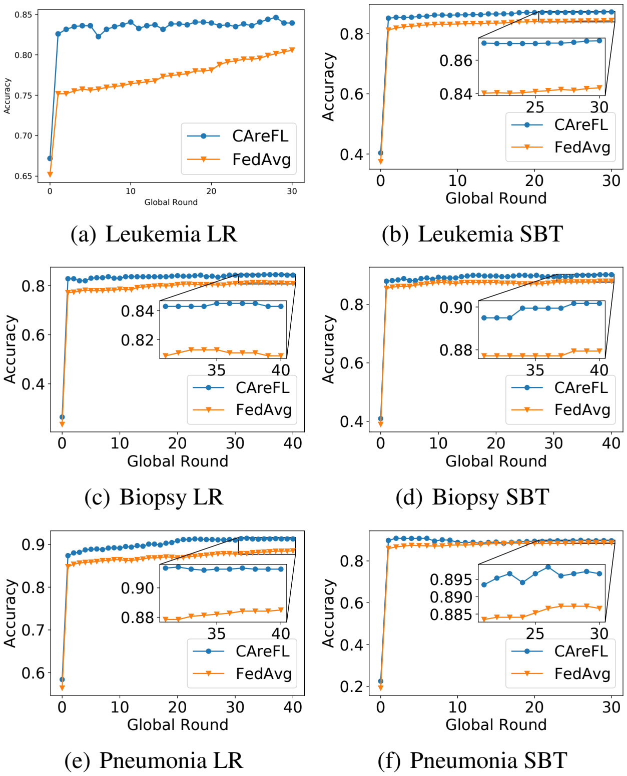
## Application Use and Payoff

The CAreFL framework has been deployed in Yidu Cloud Technology Inc. since March 2021 in two lines of their business: 1) clinical research services, and 2) real-world trial research services. Clinical research focuses on training FL models involving data silos from multiple hospitals. Real world trial research is often initiated by a pharmaceutical company which aims to leverage data from multiple hospitals to build models. Both services require data which need to be collected by the hospitals over months or years under their respective Institutional Review Board (IRB) supervision. So far, CAreFL has been used to help eight well-known medical institutions in China to train AI models for risk prediction, disease diagnosis and influence factor analysis.

**Leukemia:** CAreFL has been used to train models for recurrence risk prediction after hematopoietic stem cell transplantation for acute leukemia treatment. This is a clinical research business case. A total of 62,000 patients were included in the study, and 2,830 samples are included after acute leukemia screening and hematopoietic stem cell transplantation (709 positive cases and 1,054 negative cases at the end point of the recurrence study one year after the surgery). The results for this case are shown in Figures 6(a) and 6(b). A logistic regression (LR) model and a homo SecureBoost Tree (SBT) model (Cheng et al. 2021) have been trained with FedAvg in the previous system, and later retrained with CAreFL after its deployment. In both cases, CAreFL quickly reaches the performance plateau. It outperforms the previous system (with FedAvg) by 3.34% and 2.83% for the LR model and the SBT model, respectively.

**Biopsy:** CAreFL has been used to train models for analyzing major factors influencing prostate biopsy positivity based on real-world trial data. This is a clinical research business case. A total of 5,978 patients who underwent prostate cancer biopsy during a 5-year period were screened, and 2,426 patients are selected. The model mainly analyzes the relationship between patient age, family history, tPSA/fPSA, PSAD, preoperative testosterone, preoperative MRI, preoperative DRE, Gleason score and other factors with positive biopsy diagnosis. The results for this case are shown in Figures 6(c) and 6(d). CAreFL outperforms the previous system by 3.41% and 2.22% for the LR model and the SBT model, respectively.

**Pneumonia:** CAreFL has also been used to train models for classifying whether hospitalized pneumonia patients will be transferred to the intensive care unit (ICU) or pass away.



(a) Leukemia LR

(b) Leukemia SBT

(c) Biopsy LR

(d) Biopsy SBT

(e) Pneumonia LR

(f) Pneumonia SBT

Figure 6: Deployment comparison results (test accuracy).

This is a real-world trial research business case. A total of 103,455 sample data were selected based on hospitalization, death, and transfer to ICU for continued treatment during hospitalization. These samples contain 57 features, including basic patient information, vital signs, test results, symptoms and other information. The results for this case are shown in Figures 6(e) and 6(f). CAreFL outperforms the previous system by 2.87% and 1.00% for the LR model and the SBT model, respectively.

With the help of CAreFL, Yidu Cloud has avoided the problems of high computation costs and low accuracy of existing FL participant contribution evaluation approaches. This capability not only allows it to provide more detailed analysis about the value each FL participant can bring into building any given FL model without exposing their sensitive data, but also enables a new contribution-aware FL model aggregation approach to be developed. Overall, CAreFL is 2.84 times faster than the best performing baseline, MR, in terms of evaluating participants' contributions. It has improved the average accuracy of FL models trained by Yidu Cloud by 2.62% compared to its previous system, which is significant in industrial smart healthcare settings.

## Maintenance

As time goes by, there are additions of new types of healthcare related machine learning tasks, changes in personnel access rights, and changes in operating parameters in the system. Since the platform architecture follows a modular design approach around tasks and personnel to achieve separation of concerns, such updates can be performed without affecting the AI Engine. Since deployment, there has not been any AI maintenance task.

## Lessons Learned During Deployment

During the deployment process of the CAreFL framework, there are several lessons worth sharing.

Firstly, contribution evaluation seeks to explain the impact of each FL participant on model performance. A fine balance needs to be struck between this goal and the primary design objective of FL, which is privacy preservation. Thus, careful designs of access control and participant ID obfuscation need to be worked out together with our industry partners to ensure that private information is properly protected to a level that they can accept.

Secondly, contribution evaluation results need to be shared with the FL participants involved. Thus, it is important for our industry partners to ensure that the evaluations reflect participants' contributions fairly. Based on this consideration, they strongly prefer adopting an evaluation approach that is grounded in well-established CE principles. These lessons has shaped the final design of CAreFL.

## Conclusions and Future Work

In this paper, we reported on our experience using contribution-aware federated learning to enhance privacy-preserving collaborative training of machine learning models involving multiple healthcare industry data owners. We developed the CAreFL framework which provides fair and explainable FL participant contribution evaluation in an efficient and privacy-preserving manner, and optimizes the FL model aggregation approach based on the evaluation results. Since its deployment in March 2021 in Yidu Cloud, CAreFL has helped eight well-established medical institutions in China to train machine learning models for healthcare decision support, and has made significant positive impact by improving contribution evaluation efficiency and model performance compared to the previous system. To the best of our knowledge, it is the first successfully deployed contribution-aware federated learning framework in the healthcare industry.

In future, we will continue the explore the applicability of CAreFL in other smart healthcare application scenarios. We will also extend the CAreFL framework with contribution-based data pricing mechanisms (Pei 2020) to support the emergence of an FL-based healthcare data exchange marketplace. Eventually, we aim to incorporate these functionalities into the opensource FATE framework and make them available to more developers, researchers and practitioners.

## Acknowledgments

This research is supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG2-RP-2020-019); the NRF Investigatorship Programme (NRFI Award No: NRF-NRFI05-2019-0002); the Joint NTU-WeBank Research Centre on Fintech (Award No: NWJ-2020-008); the Nanyang Assistant Professorship

(NAP); and the RIE 2020 Advanced Manufacturing and Engineering (AME) Programmatic Fund (No. A20G8b0102), Singapore. Qiang Yang is supported by the Hong Kong RGC theme-based research scheme (T41-603/20-R) and the National Key Research and Development Program of China under Grant No. 2018AAA0101100. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

## References

Cheng, K.; Fan, T.; Jin, Y.; Liu, Y.; Chen, T.; Papadopoulos, D.; and Yang, Q. 2021. SecureBoost: A Lossless Federated Learning Framework. *arXiv preprint arXiv:1901.08755*.

GDPR. 2018. General data protection regulation. https://gdpr-info.eu/. Accessed: 2021-12-08.

Ghorbani, A.; and Zou, J. 2019. Data Shapley: Equitable Valuation of Data for Machine Learning. In *ICML*, 2242–2251.

Jia, R.; Dao, D.; Wang, B.; Hubis, F. A.; Hynes, N.; Gürel, N. M.; Li, B.; Zhang, C.; Song, D.; and Spanos, C. J. 2019. Towards Efficient Data Valuation Based on the Shapley Value. In *AISTATS*, 1167–1176.

Josang, A.; and Ismail, R. 2002. The Beta Reputation System. In *Proceedings of the 15th Bled Electronic Commerce Conference*, volume 5, 2502–2511.

Kairouz, P.; McMahan, H. B.; and et al. 2021. Advances and Open Problems in Federated Learning. *Foundations and Trends in Machine Learning*, 14(1-2): 1–210.

Kaissis, G. A.; Makowski, M. R.; Rückert, D.; and Braren, R. F. 2020. Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence*, 2: 305–311.

Krizhevsky, A. 2009. Learning Multiple Layers of Features from Tiny Images. Technical report, MIT and NYU.

LeCun, Y.; Cortes, C.; and Burges, C. 2010. MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2.

Liu, Y.; Fan, T.; Chen, T.; Xu, Q.; and Yang, Q. 2021. FATE: An Industrial Grade Platform for Collaborative Learning With Data Protection. *Journal of Machine Learning Research*, 22(226): 1–6.

Liu, Y.; Huang, A.; Luo, Y.; Huang, H.; Liu, Y.; Chen, Y.; Feng, L.; Chen, T.; Yu, H.; and Yang, Q. 2020. FedVision: An Online Visual Object Detection Platform Powered by Federated Learning. In *IAAI*, 13172–13179.

Liu, Z.; Chen, Y.; Yu, H.; Liu, Y.; and Cui, L. 2022. GTG-Shapley: Efficient and Accurate Participant Contribution Evaluation in Federated Learning. *ACM Transactions on Intelligent Systems and Technology*.

Long, G.; Tan, Y.; Jiang, J.; and Zhang, C. 2020. Federated Learning for Open Banking. In Yang, Q.; Fan, L.; and Yu, H., eds., *Federated Learning: Privacy and Incentive*, 240–254. Springer.

Lyu, L.; Xu, X.; Wang, Q.; and Yu, H. 2020. Collaborative Fairness in Federated Learning. In Yang, Q.; Fan, L.; and Yu, H., eds., *Federated Learning: Privacy and Incentive*, 185–199. Springer.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *AISTATS*, 1273–1282.

Pei, J. 2020. A Survey on Data Pricing: from Economics to Data Science. *arXiv preprint arXiv:2009.04462*.

Rieke, N.; Hancox, J.; Li, W.; Milletarì, F.; Roth, H. R.; Albarqouni, S.; Bakas, S.; Galtier, M. N.; Landman, B. A.; Maier-Hein, K.; Ourselin, S.; Sheller, M.; Summers, R. M.; Trask, A.; Xu, D.; Baust, M.; and Cardoso, M. J. 2020. The future of digital health with federated learning. *npj Digital Medicine*, 3(119): doi:10.1038/s41746–020–00323–1.

Sadilek, A.; Liu, L.; Nguyen, D.; Kamruzzaman, M.; Serghiou, S.; Rader, B.; Ingerman, A.; Mellem, S.; Kairouz, P.; Nsoesie, E. O.; MacFarlane, J.; Vullikanti, A.; Marathe, M.; Eastham, P.; Brownstein, J. S.; y. Arcas, B. A.; Howell, M. D.; and Hernandez, J. 2021. Privacy-first health research with federated learning. *npj Digital Medicine*, 4(132): doi:10.1038/s41746–021–00489–2.

Shapley, L. S. 1953. A value for n-person games. *Contributions to the Theory of Games*, 2(28): 307–317.

Sheller, M. J.; Edwards, B.; Reina, G. A.; Martin, J.; Pati, S.; Kotrotsou, A.; Milchenko, M.; Weilin Xu, D. M.; Colen, R. R.; and Bakas, S. 2020. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific Reports*, 10(12598): doi:10.1038/s41598–020–69250–1.

Shen, Z.; Yu, H.; Miao, C.; and Weng, J. 2011. Trust-based web service selection in virtual communities. *Web Intelligence and Agent Systems: An International Journal*, 9(3): 227–238.

Song, T.; Tong, Y.; and Wei, S. 2019. Profit Allocation for Federated Learning. In *IEEE BigData*, 2577–2586.

Wang, T.; Rausch, J.; Zhang, C.; Jia, R.; and Song, D. 2020. A Principled Approach to Data Valuation for Federated Learning. In Yang, Q.; Fan, L.; and Yu, H., eds., *Federated Learning: Privacy and Incentives*, 153–167. Springer.

Warnat-Herresthal, S.; Schultze, H.; Shastry, K. L.; and et al. 2021. Swarm Learning for decentralized and confidential clinical machine learning. *Nature*, 594: 265–270.

Wei, S.; Tong, Y.; Zhou, Z.; and Song, T. 2020. Efficient and Fair Data Valuation for Horizontal Federated Learning. In Yang, Q.; Fan, L.; and Yu, H., eds., *Federated Learning: Privacy and Incentives*, 139–152. Springer.

Wei, X.; Li, Q.; Liu, Y.; Yu, H.; Chen, T.; and Yang, Q. 2019. Multi-Agent Visualization for Explaining Federated Learning. In *IJCAI*, 6572–6574.

Xu, J.; Glicksberg, B. S.; Su, C.; Walker, P.; Bian, J.; and Wang, F. 2020. Federated Learning for Healthcare Informatics. *Journal of Healthcare Informatics Research*, 5: 1–19.

Yang, Q.; Liu, Y.; Cheng, Y.; Kang, Y.; Chen, T.; and Yu, H. 2019. *Federated Learning*. Morgan & Claypool Publishers.

Yu, H.; Shen, Z.; Leung, C.; Miao, C.; and Lesser, V. R. 2013. A survey of multi-agent trust management systems. *IEEE Access*, 1(1): 35–50.