連合学習のためのモデル分割，シャッフル，集約による モデル漏洩の防止に関する一考察

増田 大輝[1,a)]　北 健太朗[1,b)]　小泉 佑揮[1,c)]　武政 淳二[1,d)]　長谷川 亨[1,e)]

概要：ユーザが所有する教師データの漏洩を防ぐため，教師データを渡さずに，ユーザ自身で共有モデルを更新する連合学習が提案されている．しかし，教師データの代わりにユーザがサーバへ提供する更新前後の共有モデルの差分 (差分モデルと呼ぶ) からモデル反転攻撃によって教師データが推定される問題がある．この問題に対し，差分プライバシーを用いて，ユーザがノイズを加算した差分モデルをサーバへ送信することで教師データを推定されにくくする手法が提案されているが，ノイズが共有モデルの品質を劣化させる．本研究では，共有モデルの品質を劣化させずに，差分モデルに対するモデル反転攻撃を防ぐ連合学習システムを提案する．主なアイデアは，個々の差分モデルを攻撃者から隠蔽するために，各ユーザが差分モデルを複数のフラグメントに分割し，異なるユーザのフラグメント同士を足し合わせてから集約することである．結果として得られる共有モデルは，従来の連合学習で作製される共有モデルと同一である．

キーワード：連合学習，モデル反転攻撃，プライバシー

A Study on Mitigation of Model Leakage by Model Fragmentation, Shuffle, and Aggregation for Federated Learning

Hiroki Masuda[1,a)]　Kentaro Kita[1,b)]　Yuki Koizumi[1,c)]　Junji Takemasa[1,d)]　Toru Hasegawa[1,e)]

Abstract: Federated learning is a privacy-preserving learning system where users locally update a shared model with their own training data without sending them to the server. However, federated learning is vulnerable to a model inversion attack, which infers training data from the models updated by the users, referred to as individual models. A solution to prevent such attacks is differential privacy, where each user adds noise to the individual model before sending it to the server. Differential privacy, however, sacrifices the quality of the shared model by noise. This paper proposes a federated learning system that is resistant to model inversion attacks without sacrificing the quality of the shared model. The core idea is that each user divides the individual model into fragments, shuffles, and aggregates them to prevent adversaries from inferring training data. The resulting shared model is identical to the shared model generated with the naive federated learning.

Keywords: Federated learning, Model inversion attack, Privacy

1. はじめに

社会に寄与する機械学習モデル (モデルと呼ぶ) の作製には多様で豊富な教師データが必要不可欠であるが，一個人や一組織 (ユーザと呼ぶ) がそのような教師データを準備するのは困難である．この問題に対し，複数のユーザでコ

[1] 大阪大学 大学院情報科学研究科
Graduate School of Information Science and Technology, Osaka University, Japan
[a)] h-masuda@ist.osaka-u.ac.jp
[b)] k-kita@ist.osaka-u.ac.jp
[c)] ykoizumi@ist.osaka-u.ac.jp
[d)] j-takemasa@ist.osaka-u.ac.jp
[e)] t-hasegawa@ist.osaka-u.ac.jp

ンソーシアムを形成し，ユーザが所有する教師データを共有してモデルを作製する協調学習が医療や自動運転を初めとした様々な分野で注目されているが，プライバシーの問題が協調学習の普及の障害となりうる．具体的には，教師データが医療情報や車両の位置情報などセンシティブな情報を含む場合，コンソーシアムを形成する全てのユーザにそれらが漏洩するため，教師データのプライバシーの漏洩を恐れるユーザは協調学習に参加しない．また，欧州の一般データ保護規則 (General Data Protection Regulation: GPDR) のようにセンシティブな情報の取り扱いを法令によって制限する気運も高まっている．

この問題を解決する方法として，協調学習の一種であり，ユーザの教師データを誰とも共有せずにモデルの学習を行う連合学習 [1], [2], [3] が注目されている．連合学習では，ユーザとサーバが共同で一つのモデル (共有モデルと呼ぶ) を作製する．各ユーザはサーバから共有モデルを受信し，自身の教師データを用いて共有モデルの学習を行い，更新前後の共有モデルの差分 (差分モデルと呼ぶ) をサーバへ返送する．サーバはユーザの差分モデルを集約することで共有モデルを更新する．教師データの代わりに差分モデルをサーバへ提供することでユーザの教師データのプライバシーを保護する．

しかし，連合学習においてもプライバシーの問題が指摘されている．差分モデルは教師データの情報を含んでいるため，差分モデルから教師データを推定するモデル反転攻撃が報告されている [4]．文献 [4] では，Generative Adversarial Network (GAN) [5] によって，真のデータ，すなわちモデル反転攻撃におけるユーザの教師データと見分けがつかないデータを生成する．

この攻撃を防ぐ一つの解決法は差分プライバシー [6], [7] を用いることであり，差分プライバシーを適用した連合学習も提案されている [8], [9]．主なアイデアは，差分モデルから教師データを推定しにくくするために，ユーザは差分モデルに対しノイズを加えてからサーバへ送信する．

しかし，差分プライバシーはモデル反転攻撃に対する耐性とモデルの品質の間にトレードオフの関係があり，以下の二つの問題を抱えている．一つはノイズを加算した差分モデルに対してもモデル反転攻撃の余地が残ること，もう一つはノイズを加えることによって分類精度などの共有モデルの品質が劣化することである．本質的には，強いプライバシーを保証するために差分モデルへ加えるノイズを大きくすればするほど，共有モデルの品質は従来の連合学習で作製する共有モデルと比べて劣化する．

この問題に対処するため，本研究では，共有モデルの品質を劣化させずに差分モデルに対するモデル反転攻撃を防ぐ連合学習システムを提案する．アイデアは，各ユーザは差分モデルを個々のデータから教師データを推定できないような複数のデータ (フラグメントと呼ぶ) に分割すること，さらに，攻撃者が差分モデルを復元できないようにユーザ間でフラグメントを交換し，集約することである．結果として得られる共有モデルは従来の連合学習で作製される共有モデルと同一であるため，共有モデルの品質は劣化しない．また，ある攻撃対象ユーザに対し，他の全てのユーザが結託しなければ，提案手法は攻撃対象ユーザの差分モデルに対するモデル反転攻撃を防ぐことができる．

本研究の貢献は以下の通りである．

- 共有モデルの品質を劣化させずに差分モデルに対するモデル反転攻撃を防ぐ連合学習システムを設計した
- 本研究は通信のオーバーヘッドが従来の連合学習と非常に小さい，すなわち，提案手法の総トラヒック量は従来の連合学習の総トラヒック量と同等である

本章の構成は以下の通りである．2 章では，関連研究について説明する．3 章では，前提知識として，従来の連合学習とその脅威について説明した後，対策となる提案手法のシステムモデルと攻撃者モデルについて説明する．4 章では，提案手法の具体的な設計について説明し，5 章では，プライバシーに関する安全性と通信のオーバーヘッドに関する分析を行う．最後に，6 章で結論を述べる．

2. 関連研究

まず最初に，モデル反転攻撃について説明し，その後，プライバシーを保証する学習システムについて説明する．

モデル反転攻撃の研究が盛んとなった発端は文献 [10] である．文献 [10] では，ランダムな入力データのモデルの出力が，教師データを入力としたときのモデルの出力と近くなるように入力データを調整することで，モデルの教師データを推定する．最近の研究では，GAN を用いることで，教師データをより正確に推定する．文献 [11] では，連合学習とは異なる協調学習における共有モデルから，GAN を用いて教師データを推定する攻撃が報告されている．また，文献 [4] では，連合学習における差分モデルから，GAN を用いて教師データを推定する攻撃が報告されている．従って，モデルに対するモデル反転攻撃を防ぐ対策が必要となる．

モデル反転攻撃を防ぐため，いくつかの防御手法が提案されている．一つ目は，教師データや学習アルゴリズムの命令列を安全でないメモリ領域上では暗号化することで攻撃者から隠蔽し，Intel Software Guard Extensions のような安全なメモリ領域上のみで復号して学習を行う手法である [12], [13]．しかし，学習を高速に実行するためには，GPU などの安全でないメモリ領域上で教師データや命令列を復号して学習を行う必要があるため，教師データのプライバシー漏洩の危険性が伴う．二つ目は，差分プライバシーを用いる手法である．差分プライバシーに基づくノイズを教師データへ加算する手法 [14]，学習アルゴリズムの目的関数へ加算する手法 [15]，更新後のモデルへ加算する

表 1　記号の定義

記号	意味
U	ユーザの集合 ($u \in U$)
U_A	結託している攻撃者の集合
$\overline{U_A}$	オネストなユーザの集合 ($\overline{U_A} = U \setminus U_A$)
w^t	ラウンド t における共有モデル
D_u^t	ラウンド t で u が学習に用いる教師データ
w_u^t	ラウンド t で u が生成した更新済モデル
Δw_u^t	ラウンド t で u が生成した差分モデル
$f_{u,k}^t$	ラウンド t で u が生成したフラグメント
$\mathcal{L}$	学習アルゴリズム
$\mathcal{G}$	フラグメント生成アルゴリズム
$\mathcal{S}$	リーダー選出アルゴリズム
$\mathcal{R}$	乱数生成アルゴリズム

手法 [8] など様々な手法が提案されているが，モデルへノイズを加えたモデルの品質は，ノイズを加えていない元々のモデルと比べて必ず劣化する．最後の一つは個々の差分モデルを隠蔽しつつ差分モデルを集約する手法であり，本研究もこの手法に該当する．この手法の多くは，通信モデルとして，ユーザ間の通信は潜在的な攻撃者であるサーバを必ず経由することを仮定しており，また，プロトコルの重要な技術としてディフィーヘルマン鍵交換プロトコルを使用している [16], [17], [18], [19], [20], [21]．従って，信頼できる第三者による安全な認証がない場合，サーバによる中間者攻撃に脆弱である．これに対し，別の通信モデルとして，ピア・ツー・ピアネットワークに代表されるようにユーザ間で直接通信する通信モデルを想定した手法が提案されている [22], [23]．この手法では，オネストなユーザに対し，他のオネストなユーザとの通信が全て傍受されない限り安全なプロトコルとなっている．本研究の位置づけは，ユーザ間で直接通信を行うプロトコルを想定した上で，文献 [23] よりも通信コストが小さいプロトコルを設計する．具体的には，n 人のユーザに対し，モデルのデータサイズに相当するデータの転送回数が，文献 [23] では $\mathcal{O}(nlogn)$ であるのに対し，提案手法および文献 [22] は $\mathcal{O}(n)$ である．

3. システムモデルと攻撃者モデル

本研究では，プライバシーの漏洩を防ぐ連合学習システムを提案する．まず，連合学習とその脅威をそれぞれ 3.1 章と 3.2 章で説明する．その後，提案手法のシステムモデルおよび攻撃者モデルをそれぞれ 3.3 章と 3.4 章で示す．

3.1 連合学習

本研究で使用する主な記号の定義を表 1 に示す．集合とアルゴリズムに対し，それぞれ斜体大文字と筆記体大文字を使用する．また，それ以外の用途に斜体小文字を使用する．

連合学習では，ユーザの教師データをサーバへ提供せずにサーバとユーザの集合 U が共同で共有モデルの学習を行う．仮定として，サーバと U は作製する共有モデルの構造や学習アルゴリズム $\mathcal{L}$ に合意しているとする．各ラウンドにおいて，各ユーザ $u(\forall u \in U)$ はサーバから現在の共有モデル w^t を受信する．ここで，w^t は層の数やニューロンの数に応じた重みベクトルとバイアスベクトルで構成されるニューラルネットワークであると仮定する．次に，u は自身の教師データ D_u を用いて w^t の学習を行い、更新したモデル $w_u^t = \mathcal{L}(w^t, D_u)$ と w^t の差分 $\Delta w_u^t = w_u^t - w^t$ をサーバへ返送する．ここで，Δw_u^t を差分モデルと呼ぶ．サーバは以下のように共有モデルを更新する．

$$w^{t+1} = w^t + \eta \sum_{u \in U} \Delta w_u^t \tag{1}$$

ここで，η は各ラウンドで w をどの程度更新するかを決定する学習率である．

3.2 連合学習におけるプライバシーの脅威

連合学習では，ユーザは教師データを保持したまま学習を行うが，教師データの代わりにサーバへ送信する差分モデル自体が教師データの情報を含んでいる．このことから，差分モデルを受信する悪意のあるサーバは Generative Adversarial Networks (GAN) を用いたモデル反転攻撃によって差分モデルから教師データを推定できる [4]．GAN とは，あるデータ (真のデータ) を学習し，真のデータと似たデータを生成するモデルである．学習するデータがユーザの教師データである場合，ユーザの教師データと似たデータを生成可能だが，連合学習では，サーバがユーザの教師データを得ることはできない．代わりに，サーバは各ラウンドの共有モデル w^t と差分モデル Δw_u^t からユーザの教師データの特徴を有するデータ (代替データ) を生成する．具体的には，サーバは予め用意したランダムな入力データを w^t へ学習させ，差分モデル Δw_s^t を生成する．さらに，サーバは Δw_s^t と Δw_u^t の差を計算し，その差が近くなるように入力データを調整し，調整した入力データを用いて w^t を再度学習し直す．サーバは上記の手順を繰り返し行い，最終的に Δw_s^t が Δw_u^t と近くなったときの入力データを代替データとする．この代替データを学習した GAN が真のデータ，すなわちユーザの教師データを生成可能になる．従って，差分モデルの漏洩を防ぐ対策が必要となる．

3.3 提案手法

提案手法の目的は，任意のオネストなユーザの差分モデルを攻撃者へ漏洩せずに共有モデルを更新することである．そのため，提案手法では，図 1 に示すとおり，従来の連合学習とは異なる方法で差分モデルを集約する．まず初めに，各ユーザ u $(\forall u \in U)$ は差分モデル Δw_u^t を複数の

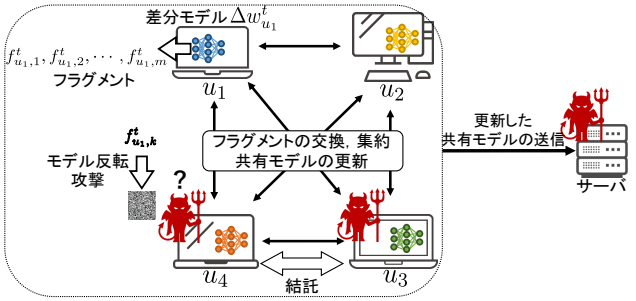

図 1 システムモデル，攻撃者モデル

データ (フラグメント) $F^t_u = \{f^t_{u,k}\}$ へ分割する．攻撃者が個々のフラグメント $f^t_{u,k}$ から u の教師データ D_u が推定できない方法で各ユーザはこれらのフラグメントを生成する．また，各フラグメントは共有モデルや差分モデルと同じネットワーク構造を持ち，以下の式を満たすように生成される．

$$\Delta w^t_u = \sum_k f^t_{u,k}. \tag{2}$$

次に，攻撃者が個々の差分モデルを復元できないようにユーザはこれらのフラグメントをユーザ同士で交換し，異なるユーザのフラグメント同士を足しあわせる．足し合わせたフラグメント (結合したフラグメントと呼ぶ) λ^t_u をサーバ，もしくは，あるユーザが収集し，全てを足し合わせ，以下のように共有モデルを更新する．

$$w^{t+1} = w^t + \eta \sum_{u \in U} \lambda^t_u. \tag{3}$$

更新後の共有モデルはサーバが保持する．

もし，フラグメントの損失がなかった場合，$\sum_{u\in U} \lambda^t_u = \sum_{u\in U} \Delta w^t_u$ より，提案手法の共有モデルの更新方法は従来の共有モデルと同一になる．具体的な設計は攻撃者モデル (3.4 章) と攻撃者モデルに対するプライバシーの要求条件 (4.1 章) を示した後に説明する．

3.4 攻撃者モデル

従来の連合学習では，攻撃者がサーバ単体であるのに対し，提案手法では，ユーザが差分モデルの集約に参加することから，ユーザの一部とサーバが結託している攻撃者を想定する．ただし，提案手法では，以下の理由からサーバを攻撃者として想定する必要はない．まず，4.3 章で示すフラグメントを交換して集約するプロトコルにサーバが関与しない，すわなち結合したフラグメントの収集や共有モデルの更新を選択されたユーザが行う (ただし，もし，サーバがフラグメントの収集や共有モデルの更新を担った場合でも，プロトコルは差分モデルに対するモデル反転攻撃を防ぐ)．さらに，既存研究において連合学習における共有モデルに対するモデル反転攻撃が報告されていないため，共有モデルから教師データを推定することはできないと仮定する．これにより，共有モデルに対してのみモデル反転攻撃を実行可能なサーバに攻撃の余地はない．

提案手法では，攻撃者として結託しているセミオネストなユーザの集合 U_A ($U_A \subset U$) を仮定する．攻撃者の目的はあるオネストなユーザ (攻撃対象ユーザ) の教師データを推定することである．また，攻撃方法として，プロトコルに従うが，プロトコルで受信するフラグメントを共有し，モデル反転攻撃を実行して攻撃対象ユーザの教師データを推定する．

U_A に対し，オネストなユーザの集合を $\overline{U_A}$ ($\overline{U_A} = U \backslash U_A$) とする．$\overline{U_A}$ に関する仮定として，攻撃対象ユーザの他にオネストなユーザが 1 人以上存在しているとする ($|\overline{U_A}| \geq 2$)．そうでない場合，提案手法は，攻撃者は攻撃対象ユーザのフラグメントを全て収集でき，収集したフラグメントを全て足すことで差分モデルを復元できるため，攻撃対象ユーザの教師データが推定される．また，オネストなユーザは攻撃ユーザの数および誰が攻撃者か知らないとする．図 1 は攻撃者モデルを例を示している．$U = \{u_1, u_2, u_3, u_4\}$, $U_A = \{u_2, u_3\}$, $\overline{U_A} = \{u_1, u_4\}$ に対し， u_2 と u_3 は結託して u_1 と u_4 教師データ ($D^t_{u_1}$ と $D^t_{u_4}$) の推定を試みる．

4. アーキテクチャ

本章では，3.3 章で説明した提案手法の具体的な設計について説明する．まず，攻撃者モデルに基づくプライバシーの要求条件を説明する．その後，プライバシーの要求条件を満たすプロトコルについて説明する．

4.1 プライバシーの要求条件

フラグメントから教師データを推定されることを防ぐためのプライバシーの要求条件を以下の通りである．(a) 攻撃者はプロトコルで受信する攻撃対象ユーザ $u(\forall u \in \overline{U_A})$ の単一のフラグメント，もしくは複数のフラグメントの和から D_u を推定できてはならない，(b) 攻撃者は攻撃対象ユーザ $u(\forall u \in \overline{U_A})$ のフラグメントの差分モデル Δw^t_u を復元できてはならない．要求条件 (a) と (b) はそれぞれ 4.2 章のフラグメント生成アルゴリズムと 4.3 章のフラグメント交換プロトコルで満たす．

4.2 フラグメント生成アルゴリズム

要求条件 (a) を満たす一つの方法は，乱数を用いることである．差分モデルは教師データの情報が含むため，差分モデルの重みベクトルやバイアスベクトルをそのままフラグメントとして使用することはできない．従って，差分モデルと同じ次元の重みベクトルとバイアスベクトルに対し，各ベクトルを乱数で構成したものをフラグメントとする．具体的には，フラグメントは以下のように生成する．

$$f_{u,k}^t = \begin{cases} \mathcal{R}() & k \neq m \\ \Delta w_u^t - \sum_{i=1}^{m-1} f_{u,i}^t & \text{otherwise} \end{cases}, \qquad (4)$$

ここで，m はフラグメントの数であり，$\mathcal{R}$ は乱数生成アルゴリズムである．また，各ベクトルが乱数で構成された $m-1$ 個のフラグメントを乱数フラグメントと呼び，差分モデルから $m-1$ 個の乱数フラグメントの和を減算することで生成する最後のフラグメントを差分フラグメントと呼ぶ．

式 (4) によって生成されるフラグメントは要求条件 (a) を満たしている．乱数フラグメントは乱数で構成されているため，ユーザの教師データに関する情報を含んでいない．従って，乱数フラグメントから教師データを推定することはできない．一方，差分フラグメントは差分モデルから $m-1$ 個の乱数フラグメントの和 $\sum_{i=1}^{m-1} f_{u,i}^t$ を減算することで生成しているため，教師データに関する情報を含んでいる．しかし，$\sum_{i=1}^{m-1} f_{u,i}^t$ のランダム性が十分に大きい場合，差分フラグメントも乱数で構成されているとみなすことができる．従って，分散の大きい乱数を乱数フラグメントに割り当てた場合，攻撃者は差分フラグメントから教師データを推定することができない．また，同様の理由から，攻撃対象ユーザの m 個のフラグメントのうち，攻撃者が差分フラグメントを含む任意の $m-1$ 個のフラグメントを収集したとしてもフラグメントから教師データを推定することはできない．差分フラグメントの安全性の評価は 5 章で説明する．

4.3 フラグメント交換プロトコル

本章では，要求条件 (b) を満たすフラグメント交換プロトコルについて説明する．また，フラグメント交換プロトコルを含む学習アルゴリズムの全容はアルゴリズム 1 の通りである．

アルゴリズムの入力は全ユーザの ID の集合 I である．各ラウンドにおいて，まず，リーダ選出アルゴリズムによってユーザの中からリーダ l を選出する．リーダは，結合したフラグメントの収集及び共有モデルの更新を担う．プライバシーの観点においてリーダの選出に関する制約はない，すなわち，以下に示すプロトコルでは，リーダが攻撃者であっても要求条件 (b) は満たされる．また，サーバもリーダになりうるが，3.4 章で示した通り，提案手法では，サーバをリーダとして選出しない．

各ユーザ u_i は共有モデルをダウンロードし (アルゴリズム 1 における 4 行目)，学習アルゴリズム $\mathcal{L}$ に従って，教師データ $D_{u_i}^t$ を用いて差分モデル $\Delta w_{u_i}^t$ を生成する (5 行目)．次に，u_i はフラグメント生成アルゴリズムによって $\Delta w_{u_i}^t$ をフラグメント $F_u^t = \{f_{u_i,k}^t\}$ に分割する (6 行目)．フラグメントは u_i を含む全てのユーザ交換するため，$n = |U| = |F_{u_i}^t|$ 個生成する．

Algorithm 1: u_i による連合学習アルゴリズム

Input: 全ユーザの ID の集合 I

1 $n \leftarrow |I|$
2 **foreach** $t \leftarrow 1, 2, \ldots$ **do**
3 　$l \leftarrow \mathcal{S}(I)$ // リーダの選出
4 　Download w^t from the server
5 　$\Delta w_{u_i}^t \leftarrow \mathcal{L}\left(w^t, D_{u_i}^t\right)$ // 差分モデルの生成
6 　$f_{u_i,1}^t, \ldots, f_{u_i,n}^t \leftarrow \mathcal{G}\left(\Delta w_{u_i}^t\right)$ // フラグメントの生成
7 　**foreach** $j \in I \setminus \{i\}$ **do** // フラグメントの交換
8 　　**if** $j < i$ **then**
9 　　　Send $f_{u_i,j}^t$ to u_j
10 　　　Receive $f_{u_{j-1},i}^t$ from u_j
11 　　**else**
12 　　　Send $f_{u_i,j-1}^t$ to u_j
13 　　　Receive $f_{u_j,i}^t$ from u_j
14 　$\lambda_{u_i}^t \leftarrow f_{u_i,n}^t + \sum_{j \in I \setminus \{i\}} f_{u_j,k_j}^t$ // 受信したフラグメントの和の計算
15 　**if** $i = l$ **then**
16 　　**for** $j \in I \setminus \{i\}$ **do**
17 　　　Receive $\lambda_{u_j}^t$ from u_j
18 　　$w^{t+1} \leftarrow w^t + \eta \sum_{u \in U} \lambda_u^t$
19 　**else**
20 　　Send $\lambda_{u_i}^t$ to l

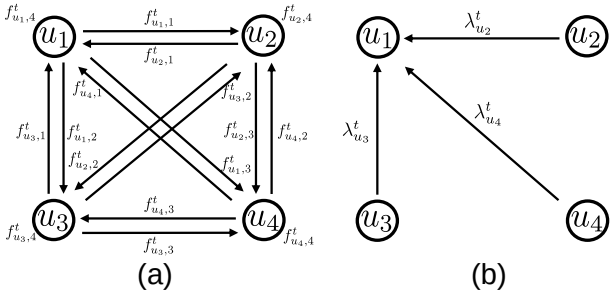

図 2 フラグメント交換プロトコル

次のステップ (7–13 行目) がフラグメント交換プロトコルである．まず，図 2(a) に示すように，u_i は差分フラグメントを手元に残し，残りの $n-1$ 個のフラグメントを他のユーザへ一つずつ送信する．次に，u_i は他のユーザから受信した乱数フラグメントの和 $\sum_{j \in I \setminus \{i\}} f_{u_j,k_j}^t$ を計算し，さらに自身の差分フラグメント $f_{u_i,n}^t$ を足し合わせる．ここで，k_j は u_j から受信した乱数フラグメントの ID である．最後に，図 2(b) に示すように，リーダ l は結合したフラグメント $\lambda_{u_i}^t$ を収集し，共有モデルを更新する (15–20 行目)．

要求条件 (b) が満たされるポイントは，1) フラグメントの交換時 (アルゴリズム 7–13 行目)，攻撃対象ユーザ u_i の全てのフラグメントを攻撃者が収集できないこと，2) アルゴリズム 1 の 14 行目が不可逆操作であることから，攻撃

者は未収集の u_i のフラグメントを任意のオネストなユーザ u の λ_u^t から算出できないことである．1) に関しては，攻撃者は u_i の手元にある差分フラグメントおよび u_i がオネストなユーザへ送信した乱数フラグメントを収集できない (3.4 章で示した通り，攻撃対象ユーザ u_i の他にオネストなユーザが一人以上存在している)．2) に関しては，攻撃者が未収集の u_i のフラグメントを λ_u^t から算出するためにはそれ以外のフラグメントの値を知っている必要がある．例えば，$\lambda_{u_i}^t$ から $f_{u_i,n}^t$ を算出するために，攻撃者は $\{f_{u_j,k_j}^t\}_{j\in I\setminus\{i\}}$ の値を知っている必要がある．しかし，λ_u^t には，u_i のフラグメントの他に必ず一つ以上のオネストなユーザのフラグメントが含まれているため，攻撃者はこの攻撃にも失敗する．従って，攻撃者は u_i の全てのフラグメントを収集できず，結果として，$\Delta w_{u_i}^t$ を復元することはできない．

4.4 トラヒック量の削減

フラグメントのデータサイズは共有モデルと同じであるため，フラグメントの交換は通信コストが大きい．各ユーザは他の全てのユーザへ乱数フラグメントを送信するため，フラグメントの転送回数は $n(n-1)$ に比例する．ここで n はユーザ数である．

この問題を解決するため，フラグメント交換時，各ユーザは乱数フラグメントを生成するためのシードを選択し，乱数フラグメントの代わりに他のユーザへ送信する．もし，全てのユーザが乱数生成アルゴリズム $\mathcal{R}(s)$ に合意している場合，各ユーザはシードから乱数フラグメントを生成することができる．ここで s はアルゴリズムのシードである．シードのデータサイズはフラグメントと比べて非常に小さいため，通信コストを大幅に削減できる．従って，フラグメント生成アルゴリズムを以下のように修正する．

$$f_{u,k}^t = \begin{cases} \mathcal{R}(s_k) & k \neq m \\ \Delta w_u^t - \sum_{x=1}^{m-1} f_{u,x}^t & \text{otherwise} \end{cases}. \qquad (5)$$

各ユーザ u は $f_{u,k}^t$ の代わりに s_k を他のユーザへ送信する．

5. 分析

本章では，モデル反転攻撃に対する攻撃耐性の評価と通信のオーバーヘッドに関して評価を行う．

5.1 差分フラグメントの安全性の評価

この章では，式 (4) や式 (5) において，分散の大きい乱数が差分フラグメントに含まれる教師データの隠蔽にどの程度貢献するか評価する．モデル反転攻撃に成功したかどうかはしばしば人が推定結果を見て判断するため，定量的に評価することは難しいが，本研究では，分散を変化させた乱数をモデルへ加え，乱数を加えたモデルのアキュラシィによって評価する．もし，モデルのアキュラシィがあ

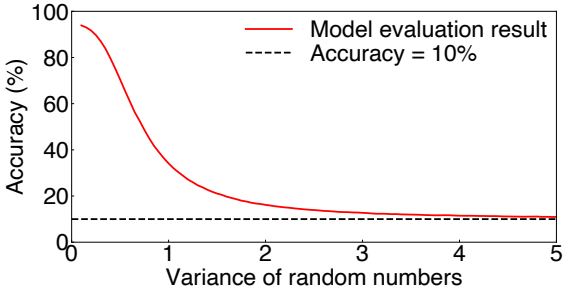

図 3 乱数を加えたモデルのアキュラシィ

てずっぽうと同じ精度である，具体的には，モデルが y 個のラベルを持っているとき，モデルのアキュラシィが $1/y$ である場合，そのモデルに含まれる教師データはモデル反転攻撃によって推定できない程度に隠蔽されているとする．

本研究では，MNIST 手書き数字画像データセット [24] を学習した画像認識モデルを評価モデルとして使用する．モデルは入力層，隠れ層，出力層からなる 3 層のモデルであり，各層のニューロン数はそれぞれ，784，1000，10 である．また，隠れ層と出力層の活性化関数はそれぞれシグモイド関数とソフトマックス関数である．10 個のラベル (数字の 0〜9) を持つ 60000 枚の教師データをモデルへ学習させ，10000 枚のテストデータでモデルのアキュラシィを測定した．

乱数を加えない場合のモデルのアキュラシィは 94.5%である．本研究では，平均が 0 かつ分散を変化させた正規乱数をモデルに加え，そのアキュラシィを測定した．結果は図 3 の通りである．横軸と縦軸はそれぞれ乱数の分散とモデルのアキュラシィを示している．分散が大きくなるに連れ，モデルのアキュラシィは急速にあてずっぽうの精度，すなわち 10%に収束している．この結果は，分散が十分に大きい乱数を加えたモデルは教師データに関する情報を完全に隠蔽していることを示している．

5.2 攻撃者モデルの妥当性

攻撃対象ユーザの他にオネストなユーザが 1 人以上存在している限り攻撃者は攻撃対象ユーザの差分モデルを復元することはできない，この章では，3.4 章で示したこの攻撃者モデルの仮定が妥当かどうか評価する．攻撃対象ユーザの他にオネストなユーザが 1 人以上存在している確率 $\mathbb{P}$ は $1-a^{n-1}$ である．ここで，a はあるユーザが攻撃者である確率であり，n はユーザ数である．図 4 はユーザ数が多くなるにつれて，$\mathbb{P}$ が急速に 1 に収束していることを示している．

5.3 通信のオーバーヘッド

最後に，従来の連合学習の総トラヒック量に対する提案手法の通信のオーバーヘッドを評価する．本章の目的はト

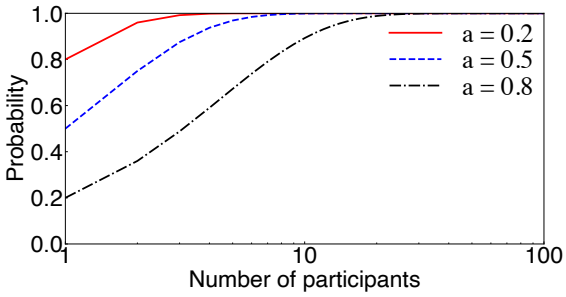

図 4　攻撃対象ユーザの他にオネストなユーザが 1 人以上いる確率

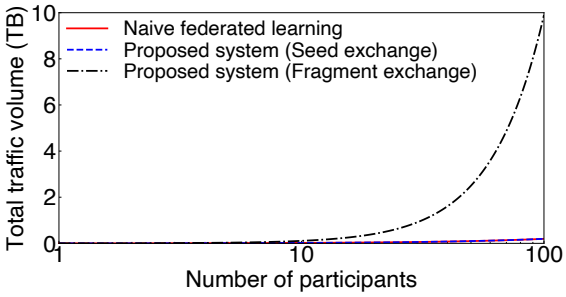

図 5　総トラヒック量

ラヒック量を測定することであるため，5.1 章で用いたモデルよりも実際に社会で活用されうるサイズが大きいモデルを評価モデルとする．具体的には，畳み込みニューラルネットワークである ResNet-50 モデル [25] を使用する．ResNet-50 モデルのデータサイズは約 98 MBytes であるため，フラグメントのデータサイズも 98 MBytes である．これに対し，乱数のシードのデータサイズは 32 Bytes (256 bits) とする．

一ラウンドにおける総トラヒック量は図 5 の通りである．図 5 の横軸と縦軸はそれぞれユーザ数と総トラヒック量を示している．フラグメントそのものをユーザ間で交換する場合，総トラヒック量はユーザ数が増えるにつれて急速に増大する．一方，フラグメントの代わりにシードをユーザ間で交換する場合，シードのデータサイズはフラグメントと比べて無視できる程小さいため，従来の連合学習と提案手法の総トラヒック量は同等である．

6. おわりに

本研究では，共有モデルの品質を劣化させずに差分モデルに対するモデル反転攻撃を防ぐ連合学習システムを提案した．攻撃対象ユーザの教師データの漏洩を防ぐためのプライバシーの要求条件を定義し，要求条件を満たすアルゴリズムを設計した．提案手法は攻撃対象ユーザに対し，他の全てのユーザが結託しなければ，差分モデルに対するモデル反転攻撃を防ぐことができる．攻撃対象ユーザ以外の全てのユーザが結託することは難しいため，攻撃対象ユーザの教師データのプライバシーは保護される．

謝辞　本研究は，科研費 21H03442 によるものである．

参考文献

[1] Jakub Konečný, H. Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*, October 2016.

[2] Jakub Konečný, H. Brendan McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, October 2016.

[3] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol.*, No. 2, January 2019.

[4] Zhibo Wang, Mengkai Song, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *Proceedings of IEEE INFOCOM*, 2019.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Adv. Neural Inf. Process. Syst.*, Vol. 27. Curran Associates, Inc., 2014.

[6] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of Theory of Cryptography Conference*. Springer, 2006.

[7] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, Vol. 9, No. 3-4, 2014.

[8] Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H Yang, Farhad Farokhi, Shi Jin, Tony QS Quek, and H Vincent Poor. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Trans. Inf. Forensics Security*, 2020.

[9] Stacey Truex, Nathalie Baracaldo, Ali Anwar, Thomas Steinke, Heiko Ludwig, Rui Zhang, and Yi Zhou. A hybrid approach to privacy-preserving federated learning. In *Proceedings of ACM AISec*, 2019.

[10] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of ACM CCS*, 2015.

[11] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. Deep models under the GAN: Information leakage from collaborative deep learning. In *Proceedings of ACM CCS*, 2017.

[12] Florian Tramer and Dan Boneh. Slalom: Fast, verifiable and private execution of neural networks in trusted hardware. In *Proceedings of ICLR*, 2019.

[13] Zhongshu Gu, Hani Jamjoom, Dong Su, Heqing Huang, Jialong Zhang, Tengfei Ma, Dimitrios Pendarakis, and Ian Molloy. Reaching data confidentiality and model accountability on the caltrain. In *Proceedings of IEEE/IFIP DSN*, 2019.

[14] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. Extremal mechanisms for local differential privacy. *J. Mach. Learn. Res.*, Vol. 17, No. 1, 2016.

[15] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of*

ACM CCS, 2016.

[16] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of ACM CCS*, 2017.

[17] K. A. Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for federated learning on user-held data. In *Proceedings of PMPML*, 2016.

[18] Swanand Kadhe, Nived Rajaraman, O Ozan Koyluoglu, and Kannan Ramchandran. Fastsecagg: Scalable secure aggregation for privacy-preserving federated learning. *arXiv preprint arXiv:2009.11248*, 2020.

[19] Guowen Xu, Hongwei Li, Sen Liu, Kan Yang, and Xiaodong Lin. Verifynet: Secure and verifiable federated learning. *IEEE Transactions on Information Forensics and Security*, Vol. 15, pp. 911–926, 2019.

[20] James Henry Bell, Kallista A Bonawitz, Adrià Gascón, Tancrède Lepoint, and Mariana Raykova. Secure single-server aggregation with (poly) logarithmic overhead. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1253–1269, 2020.

[21] Xiaojie Guo, Zheli Liu, Jin Li, Jiqiang Gao, Boyu Hou, Changyu Dong, and Thar Baker. V eri fl: Communication-efficient and fast verifiable aggregation for federated learning. *IEEE Transactions on Information Forensics and Security*, Vol. 16, pp. 1736–1751, 2020.

[22] Hiroki Masuda, Kentaro Kita, Yuki Koizumi, Junji Takemasa, and Toru Hasegawa. Model fragmentation, shuffle and aggregation to mitigate model inversion in federated learning. In *2021 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, pp. 1–6, 2021.

[23] Jinhyun So, Başak Güler, and A Salman Avestimehr. Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning. *IEEE Journal on Selected Areas in Information Theory*, Vol. 2, No. 1, pp. 479–489, 2021.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. The mnist database of handwritten digits, 1998. http://yann.lecun.com/exdb/mnist/.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF CVRP*, 2016.