代表アカウントによる オンデバイスプライバシー保護推薦システム

新田　洸平†　　加藤　　誠††

† 筑波大学大学院 情報学学位プログラム　〒 305–8550 茨城県つくば市春日 1-2
†† 筑波大学 図書館情報メディア系　〒 305–8550 茨城県つくば市春日 1-2
E-mail: †s2221648@s.tsukuba.ac.jp, ††mpkato@acm.org

あらまし　本論文では，ユーザが自身のデータを推薦システムに提供したくない場合に，ユーザがデータを提供しなくても推薦の恩恵を受けられ，システムを改変せずにユーザの持つデバイス上のみで興味に応じた結果を提示できる手法を提案する．具体的には，既にデータ提供しているユーザの特徴量を集約して特徴に応じて推薦結果を出力する代表アカウントを作成，各ユーザの持つデバイスにおいてデータをシステムに提供せず，代表アカウントへの推薦結果をユーザに応じてデバイス上で組み合わせることで興味に応じた推薦を実現する．実験では推薦システムの評価データセットを用いて，既存手法と提案手法を比較検討し，代表アカウントを利用したオンデバイス推薦が性能にどのような影響を与えるかについて検証した．

キーワード　オンデバイス推薦, ユーザモデリング, プライバシ, パーソナライゼーション, 推薦モデル

1　はじめに

推薦システムにおいて，ユーザの興味に応じて提示するアイテムのパーソナライゼーションは重要であり，ユーザ体験の満足度をより向上することにつながる．パーソナライゼーションを行うために，推薦システムは閲覧履歴や購買履歴といったユーザ行動ログデータを収集・利用する必要がある．これまで，推薦システムのパーソナライゼーションに関する多くの研究が盛んに行われている．

しかし，従来の研究はユーザの個人情報やパーソナルデータに該当するようなデータを必要としており，ユーザの個人情報やパーソナルデータに該当するようなデータが含まれている．GDPR[1] や CCPA[2] などをはじめとする個人情報に関する法律が世界各国で整備されており，個人情報やパーソナルデータに該当するようなデータの取り扱いにはますます規制がかかっている．加えて，スマートフォンを利用する多くのユーザが自身のデータを提供しない選択をしていることもわかっており，アプリ分析サービスを提供している Flurry 社 [3]は全世界の iOS ユーザにおけるオプトインの許可率が 15% 程度にとどまっていることを示している [4]．法制度による規制やユーザがデータを提供しないことで安全性は向上するが，データが使えないことで推薦システムの精度の低下につながりユーザ満足度を下げることにつながる．こういった状況に対して，プライバシーを保護しつつユーザデータを利用して推薦システムの精度向上に取り組む研究が多く提案されている．しかし，既存手法のオンデバイス推薦に関する研究の多くは，推薦システムを運用する事業者視点からの提案であり，サーバサイドの改変を必要とする．

そこで，本研究ではユーザがデータを提供しなくても推薦の恩恵を受けられ，システムを改変せずにユーザの持つデバイス上のみで興味に応じた結果を提示できる手法を提案する．具体的には，既にデータ提供しているユーザの特徴量を集約して特徴に応じて推薦結果を出力する代表アカウントを作成，各ユーザの持つデバイスにおいてデータをシステムに提供せず，代表アカウントへの推薦結果をユーザに応じてデバイス上で組み合わせることで興味に応じた推薦を実現する．

具体的な実現方法として既にデータ提供しているユーザの特徴量を集約して特徴に応じて推薦結果を出力する代表アカウントを作成，各ユーザの持つデバイスにおいてデータをシステムに提供せず，代表アカウントへの推薦結果をユーザに応じてデバイス上で組み合わせることで興味に応じた推薦を行った．

提案手法の有効性を検証するために推薦システム評価のデータセットである MovieLens を用いて実験を行った．提案手法では，サーバサイドで利用可能なデータとデバイスサイドで利用可能なデータが分かれるため，実験ではデータセットをサーバサイドとデバイスサイドに対応するように分けて実験を行った．性能比較実験として，データ利用範囲の違いによる性能の比較，システム改変の有無による性能の比較を行った．

本研究の貢献は以下のとおりである：

（1） 推薦システムにおいてユーザが自身のデータを提供しなくても推薦の恩恵を受けられる手法の提案を行った．

（2） 推薦システムにおいてサーバサイドの仕組みを変更することなくプライバシー保護を実現できる手法の提案を行った．

本論文の構成は次の通りである．第 2 節では関連研究として

1：https://gdpr-info.eu/
2：https://oag.ca.gov/privacy/ccpa
3：https://www.flurry.com/
4：https://www.flurry.com/blog/ios-14-5-opt-in-rate-att-restricted-app-tracking-transparency-worldwide-us-daily-latest-update/

オンデバイス推薦の研究と，ユーザ特徴の集約の研究について述べる．第 3 節では提案手法である代表アカウントによるオンデバイス推薦の詳細について述べる．第 4 節ではデータセットや比較するベースライン，評価指標などの実験設定について述べる．第 5 節では実験結果を示し考察について述べる．第 6 節では本論文の結論と今後の課題を述べる．

2 関 連 研 究

本節では，関連研究について述べる．関連研究としてプライバシー保護推薦システムの研究，ユーザ特徴に注目した推薦システムの研究という 2 つのトピックについてそれぞれ述べる．

2.1 プライバシー保護推薦システム

プライバシー保護推薦システムの関連研究について述べる．プライバシーに関する法制度の整備やユーザのデータ提供の拒否の増加などの課題から，プライバシー保護を目的とした推薦システムの研究が盛んに行われている．プライバシー保護に関する研究として連合学習を用いた推薦の研究，オンデバイスな推薦の研究について述べる．

2.1.1 連合学習による推薦

連合学習 [1] [2] とは，クライアントのデータをサーバに送信することなく，クライアントにおけるモデルのパラメータのみをサーバに送信することで，サーバのモデルを更新して学習を行うような機械学習手法の一つである．データを直接第三者に提供する必要がないためユーザの観点からはプライバシー保護につながり，サーバの観点からは個人情報やパーソナルデータを保持するリスクなく安全にモデルの更新ができる．また，プライバシー保護という観点以外にも，エッジコンピューティングの観点から通信の効率化などの側面もある．これらのことから，連合学習の研究は近年盛んに行われており [3] [4] [5] [6]，特に，情報推薦や情報検索の文脈においても研究が行われるようになってきている [7] [8] [9] [10] [11] [12].

これらの連合学習による推薦と提案手法の違いは，提案手法においてサーバの改変が不要であるという点である．連合学習による推薦手法は，サーバサイドの観点から取り組まれており，サーバにおけるグローバルモデルをいかに更新するかということを前提としている．一方で，提案手法では，既に実装されているモデルの出力から代表アカウントを生成し，ユーザサイドであるデバイス上にある推薦モデルによって推薦を行うため，サーバサイドの改変は不要である．

2.1.2 オンデバイス推薦

オンデバイスとは，ユーザが持つスマートフォンやデスクトップコンピュータなどのデバイス上で行われる処理のことである．連合学習と同様に，プライバシーにおけるメリットやオフラインでも機械学習モデルを利用できるなどのメリットがあり，特に，オンデバイス機械学習モデルの研究が近年盛んに行われている [13]．連合学習との違いとして，オンデバイス機械学習モデルはサーバからモデルを一方向で受け取ってモデルの学習を行うもしくは完全にデバイス上のみでモデルの学習を行う．つまり，パラメータであってもサーバに送信することがないような手法である．

このようなオンデバイス機械学習モデルを用いた研究が，情報推薦の文脈においても研究が行われるようになってきている [14] [15] [16] [17] [18] [19] [20] [21] [22]．Han らは，モバイルデバイス上で行われる E コマースサービスのシーケンシャル推薦をアイテム候補とモデルをデバイスに送信して，デバイス上のみで行動ログデータを利用することで，データやパラメータを外部に漏らすことなく推薦する手法の提案を行なっている [16]．Xu らは，モバイルデバイスにおける文字入力のパーソナライズにおける予測モデルをオンデバイスで学習する手法の提案を行なっている [21]．Epasto らは，プライバシー保護を目的としてソーシャルネットワークを利用したオンデバイス推薦システム手法の提案を行っている [19]．また他にも関連したオンデバイス手法の研究として，モバイルデバイスでショートメッセージの類似検索を手法の研究 [23]，デバイス上の類似文検索タスクにおける文の埋め込みを精度をなるべく落とさず軽量化する手法の研究 [24] などがある．

さらに，ユーザサイド検索・ユーザサイド推薦と呼ばれる手法がいくつかある [25] [26] が，これらの研究もサーバサイドに対してデータやパラメータの送信を行わず，デバイス上のみでモデルの構築・学習やデータの処理を行うことから，オンデバイス推薦手法と問題設定を共有しており強く関連する．

本研究における提案手法は，オンデバイス推薦の手法の一つとして位置付けられる．既存研究と提案手法の違いは，サーバの改変が不要であるという点である．既存手法は，サーバサイドの観点から取り組まれており，サーバにおけるモデルをデバイス上でいかに利用するかという点に焦点が当たっている．一方で，提案手法ではサーバサイドの改変は不要である．Sato によるオンデバイス推薦の研究 [25] は，完全にオンデバイスな推薦手法の研究ではあるが，公平性を担保するような手法であり目的が異なっており，アイテムを入力としてアイテムを出力する推薦であるという点で問題設定が異なる．

2.2 ユーザ特徴に注目した推薦システム

ユーザ特徴に注目した推薦システムの関連研究について述べる．ユーザの興味は 1 つではなくさまざまな興味を持つ．また，時間や場所，これまでの消費アイテムなどといったコンテキストに応じて興味は時間変化する．そのため，その時々のユーザの興味に応じた推薦結果を提示することが重要であり，ユーザの特徴に注目してその時々の興味をいかに捉えるかが重要である．そこで，ユーザの特徴や興味をモデル化する研究が盛んに行われている [27] [28] [29] [30]．ユーザの行動ログから特徴量生成モデルによってユーザの特徴をモデル化することで，ユーザの特徴をより詳細に捉えて精度良く推薦を行うことができる．

これらのユーザ特徴に注目した推薦システムと提案手法の違いは，提案手法においてサーバにユーザ自身のデータ送信が不要であるという点である．既存研究では，ユーザ自身のデータをサーバに送る必要があり，送信されたデータを利用することでユーザ特徴を捉えようとしている．一方，本研究ではユーザ

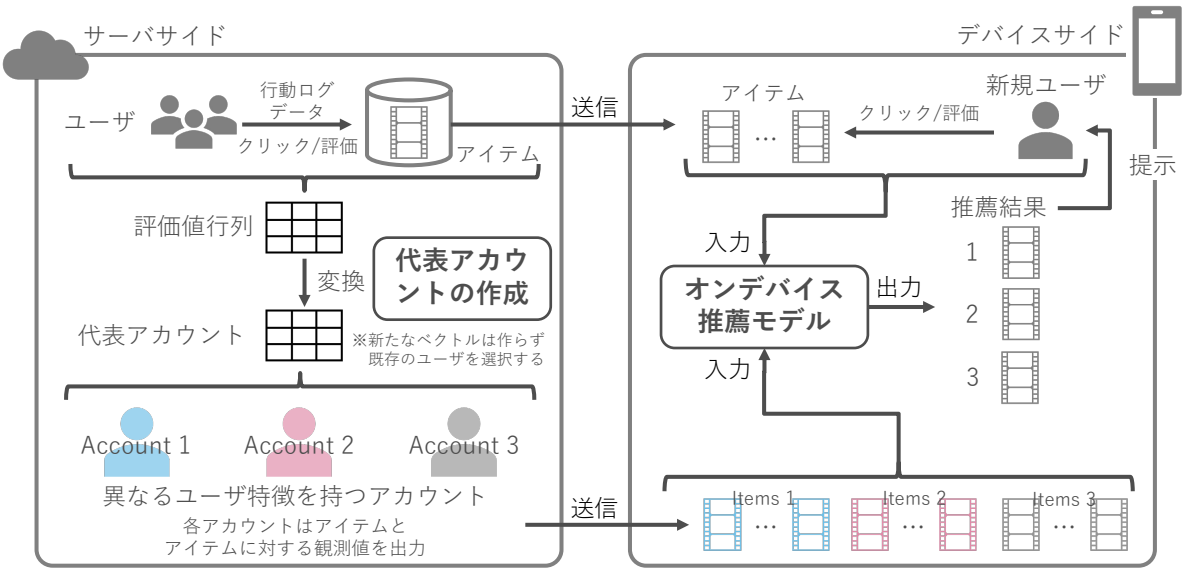

図 1　代表アカウントによるオンデバイスプライバシー保護推薦システムの構成

がサーバにデータを送信しない場合を想定しているため，あるユーザの興味を他のユーザの興味から再現する必要があり既存研究とは異なる．

3　提 案 手 法

本節では提案手法について述べる．2 節で述べたように，推薦システムにおけるプライバシーを実現するために，2.1 のプライバシー保護推薦システムの研究において 2.1.1 の連合学習による推薦，2.1.2 のオンデバイス推薦ではサーバの改変を必要としている．また，2.2 のユーザ特徴に注目した推薦システムでは，ユーザ自身のデータをサーバに送信する必要がある．そこで，本研究では，ユーザが自身のデータをサーバに提供しなくても推薦の恩恵を受けられ，システムを改変せずにユーザの持つデバイス上のみで興味に応じた結果を提示できるシステムを提案する．

3.1　提案システムの概要

提案システムの概要について述べる．提案システムにおいて，（1）ユーザのデータをデバイスからサーバに送信しない，（2）サーバの推薦モデル自体を改変しない，という 2 つの要件を設定する．これら 2 つの要件を達成するために，代表アカウントモデル，オンデバイス推薦モデルを含むプライバシー保護推薦システムを提案する．提案システムの全体構成図を図 3 に示す．

3.2　問 題 設 定

推薦システムを利用するユーザ u のアイテム v に対する評価を r とする．ここで,，推薦システムに対する入力としてユーザ集合 $\mathcal{U} = \{u_1, ..., u_n\}$, アイテム集合 $\mathcal{V} = \{v_1, ..., v_m\}$，評価行列 $r_{i,j} \in \mathcal{R}$ が与えられる．また，各ユーザの特徴量を x とする．

本研究では，サーバとデバイスで利用可能なデータが異なる．想定として，サーバではサーバに対してデータ提供を許可したユーザのデータのみが利用可能であり，デバイスではデバイスを利用しているユーザ自身のデータのみが利用可能である．ここで，サーバにおけるユーザ集合を $\mathcal{U}^{\mathrm{S}} = \{u_1^{\mathrm{S}}, ..., u_n^{\mathrm{S}}\}$，サーバにおけるアイテム集合を $\mathcal{V}^{\mathrm{S}} = \{v_1^{\mathrm{S}}, ..., v_m^{\mathrm{S}}\}$，サーバにおける評価行列を $r_{i,j}^{\mathrm{S}} \in \mathcal{R}^{\mathrm{S}}$ とする．また，デバイスにおけるユーザ集合を $\mathcal{U}^{\mathrm{D}} = \{u_1^{\mathrm{D}}, ..., u_n^{\mathrm{D}}\}$，デバイスにおけるアイテム集合を $\mathcal{V}^{\mathrm{D}} = \{v_1^{\mathrm{D}}, ..., v_m^{\mathrm{D}}\}$，デバイスにおける評価行列を $r_{i,j}^{\mathrm{D}} \in \mathcal{R}^{\mathrm{D}}$ とする．デバイスにおける評価行列は，デバイスを利用しているユーザのデータにのみアクセスできるため，$1 \times m$ となる．

最終的な出力として，システムはユーザに対して推薦アイテムの順位付けされたリストを出力する．

3.3　代表アカウントモデル

代表アカウントモデルについて述べる．代表アカウントモデルの目的は，サーバに送信されたユーザのアイテムに対する行動ログデータから多くのユーザの興味をよりカバーするような特徴を持つユーザを選択することである．選択したユーザの特徴を用いることで，データ提供した引用なユーザに対しても，他のユーザの特徴とその特徴に対する推薦結果から，興味に応じた推薦を再現するためである．

代表アカウントモデルは，サーバにおけるユーザ，アイテム，行動ログを入力として，複数のアカウントと各アカウントに紐づく推薦アイテムを出力する．本研究における代表アカウントモデルは 3 段階で構成される．

（1）　全てのユーザのアイテムに対する行動ログデータから

ユーザ特徴量を生成

（2） ユーザ特徴量から代表となるような特徴量を複数選択

（3） 選択した特徴量と近似するようなユーザを選択してアカウントに割り当て

以上の操作は，ユーザ・アイテム評価データを提案システムに入力するだけであり，サーバにおいて機能やモデルの改変を行っていない．

まず，ユーザ特徴量を生成するために，ユーザのアイテムに対する行動ログを行列として，特異値分解（Singular Value Decomposition, SVD）を行い，サーバサイドにおける全てのユーザの特徴量を生成する．評価値行列 $\mathbf{R}$ に対する特異値分解を式 1 に示す：

$$\mathbf{R}_{n\times m} = \mathbf{P}^{\mathrm{T}}_{n\times k}\mathbf{Q}_{k\times m} \tag{1}$$

ここで，$\mathbf{P}$ はユーザの潜在因子を表す行列，$\mathbf{Q}$ はアイテムの潜在因子を表す行列であり，n はユーザ数,m はアイテム数，k は特徴量となる因子の数を表す．また，未知のアイテムに対するユーザの予測評価値を式 2 で求める：

$$\mathbf{r}'_{nm} = p_n^{\mathrm{T}} q_m \tag{2}$$

次に，代表となるような特徴量を選択するために k-medois 法を用いて，式 1 で求めたユーザ特徴量 $\mathbf{P}$ に対してクラスタリングを行い，各クラスタの重心となる特徴量を取得する．式 3 に示す．

$$\arg\min_{x\in X_i} \sum_{y\in(X_i-\{x\})} d(x,y) \tag{3}$$

ここで，X_i はユーザ特徴のクラスタであり，x はユーザ特徴量，$d(x,y)$ はデータ間の非類似度を表す．k-medois 法によって選択した特徴量と最も近似するユーザの特徴量をアカウントとして選択し，この操作をクラスタ数に応じて繰り返すことで，代表アカウントとする．

そして，求めた代表アカウントに式 2 で求めた予測評価値を対応づけることで各代表アカウントに対してアイテムの推薦を行う．代表アカウントに対する推薦アイテムのみを利用することで，デバイスにおけるユーザへの推薦を可能とする．

3.4 オンデバイス推薦モデル

オンデバイス推薦モデルについて述べる．オンデバイス推薦モデルの目的は，複数の代表アカウントの中からをユーザの興味に適合するアカウントを選択して，選択したアカウントの出力結果をユーザの興味に適合するように混合し並べて出力することである．本研究において注目しているユーザは，サーバに対してデータを送っておらず，推薦システムの恩恵を受けられない．そのため，代表アカウントとデバイス上のデータを利用することで興味に応じた推薦を行う．

オンデバイス推薦モデルは 2 段階で構成される．

（1） 複数の代表アカウントの中でユーザの興味により応じた出力を得られそうなアカウントの選択

（2） 選択したいくつかのアカウントの出力をユーザの興味に応じてランキングの生成

まず，全ての代表アカウントの中からユーザの興味に適合するアカウントを選択する．選択方法として，Oosterhuis と de Rijke によって提案されたインターリービング手法の一つである Pairwise Preference Multileaving [31] を用いる．代表アカウントをランキングアルゴリズムと捉えて，代表アカウントによる出力を交互に配置してランキングを生成する．配置されたランキングに対してデバイス上の学習データをクリックと捉えて，ユーザに対してどの代表アカウントが良いかを算出して重み付けを行う．予測時には，より重みの大きい代表アカウントの出力から優先的にユーザに対して出力するようにランキングを生成する．

4 実験設定

本節では実験設定について述べる．本研究では，ユーザが自身のデータをサーバに提供しなくても推薦の恩恵を受けられ，システムを改変せずにユーザの持つデバイス上のみで興味に応じた結果を提示できるシステムを提案し，ユーザのデータをデバイスからサーバに送信しないこと，サーバの推薦モデル自体を改変しないことという 2 つの要件を設定した．要件を達成するために提案した代表アカウントモデル，オンデバイス推薦モデルを含む提案システムについて実験を通して性能評価を行う．既存の推薦システムの実験では，データセットを全て利用可能であるが，本研究ではデータ利用に制約をかけるため，既存手法を性能で上回ることはないと想定される．そのため，本研究においては既存手法に対して提案手法の性能をいかに近づけられるかという観点で実験を行う．

本研究におけるリサーチクエスチョンは次の通りである．（**RQ**）提案手法は既存の推薦手法に対してどの程度パフォーマンスを近づけられるか？

表 1 データセットの統計

Dataset	Users	Items	Actions
MovieLens 1M	6,040	3,900	1,000,209

4.1 データセット

データセットについて述べる．提案モデルの性能を評価するために推薦システムのデータセットである MovieLens 1M Dataset[5] を用いた．MovieLens 1M Dataset の統計情報を表 1 に示す．提案システムでは，サーバサイドとデバイスサイドで利用可能なデータが分かれている．また，一般的な推薦システムの実験では評価やクリックなどの行動データを軸に学習データとテストデータを分割するが，本研究ではユーザを軸にサーバサイドのユーザ，デバイスサイドのユーザという設定で分割する．そのため，まず，データセットにおけるユーザをサーバサイドとデバイスサイドそれぞれ 5 : 5 の割合で分割する．次に，分割したデバイスサイドのユーザに対応する行動データをユーザごとに学習とテストそれぞれ 5 : 5 の割合で分割して実

5：https://grouplens.org/datasets/movielens/1m/

表 2　Train/Test の統計

	Server			Device					
	Train			Train			Test		
Dataset	Users	Items	Actions	Users	Items	Actions	Users	Items	Actions
MovieLens 1M	3,020	3,616	491,281	3,020	3,440	253,743	3,020	3,525	255,185

験に用いる．サーバサイドに該当するユーザに対応する行動データは全てサーバサイドのモデルの学習に用いる．分割結果を表 2 に示す.

4.2 評価指標

評価指標について述べる．まず，サーバサイドの性能を測定するために，ベースラインの設定に対して行う評価の指標として推薦システムの評価に一般に用いられる MAE（Mean Absolute Error）と RMSE（Root-Mean-Square Error）を用いる．MAE を式 4， RMSE を式 5 にそれぞれ示す.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{4}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{5}$$

ここで， $\hat{y}_i$ はモデルによる予測評価値， y_i は正解である実際の評価値， n は評価対象となるテストデータ数，つまり評価対象となるユーザ数であるである．MAE と RMSE はどちらも予測評価値と実際の評価値の誤差をデータ数で平均しているため，値がより小さいほどモデルの性能が良いことが示される．

次に，デバイスサイドのユーザに対する最終的な推薦結果の評価指標として MRR(Mean reciprocal rank) と nDCG(Normalized discounted cumulative gain) を用いる．RR を式 6，MRR を式 7，DCG を式 8,nDCG を式 9 にそれぞれ示す.

$$\mathrm{RR}_u = \max_{\langle i,v\rangle \in R_u} \frac{rel_u(v)}{i} \tag{6}$$

$$\mathrm{MRR}(U) = \frac{\mathrm{RR}(u_1) + ... + \mathrm{RR}(u_n)}{|U|} \tag{7}$$

$$\mathrm{DCG}_u = \sum_{\langle i,v\rangle \in R_u} \frac{2^{rel_u(v)}}{\log_2(i+1)} \tag{8}$$

$$\mathrm{nDCG}_u = \frac{\mathrm{DCG}_u}{\mathrm{IDCG}_u} \tag{9}$$

ここで，$u \in U$ は評価対象のユーザ,$\langle i,v\rangle \in R_u$ はユーザに対する推薦アイテム v と順位 i のペア，$rel_u(v)$ はユーザに対するアイテムの適合性である．MRR はユーザごとの Reciprocal rank を平均した値となる．nDCG はテストデータから算出した理想的な DCG と評価対象となるユーザへの推薦結果の適合性から算出した DCG によって算出した値となる．本研究では，Burges らによって提案された Microsoft バージョンの nDCG を用いる [32].

4.3 ベースライン

ベースラインについて述べる．提案手法の比較手法となるベースラインは， SVD による推薦を用いる．ベースライン手法では，既存研究における一般的な推薦システムにおける実験と同じ設定にするために，学習において利用するデータの制約を設けない．つまり，サーバとデバイスで分けた区別は関係なく，単に学習データとテストデータで分けて学習と評価を行うこととなり，表 2 における Server Train と Device Train で学習して　Device Test によって評価することとなる．データセットにおける評価値は疎であることから，学習データに出現しないユーザとアイテムの予測評価値が多く存在するが，この場合は平均評価値を用いる.

また，ベースライン手法の性能について参考として以下の表 3 にベースライン手法に対して RMSE, MAE によって性能評価を行った結果を示す．ベースライン手法が一般的な推薦システムにおける性能であると考えることができ，ベースラインの性能に対して提案手法がどの程度性能を近づけられるかを明らかにすることが実験の要点となる.

表 3　ベースラインにおける RMSE, MAE

Model	RMSE	MAE
Baseline	0.8436	0.6687

5 実験結果と考察

実験結果と考察について述べる．表 4 に実験結果を示す．@K となっている数字は上位 K 件を評価したことを表す.

表 4 の実験結果より，ベースライン手法の MRR, nDCG のスコアがいずれ評価対象数においても提案手法より大きく上回っており，提案手法がベースラインよりも性能が低いことがわかる.

リサーチクエスチョンとして，提案手法は既存の推薦手法に対してどの程度パフォーマンスを近づけられるか？という問いを立てたが，一般的な推薦モデルであるベースライン手法をかなり下回る結果となり，ほぼ近づけられなかったことがわかる．考察として，提案手法は利用可能なデータに制限があり，制限なくデータを利用できるベースラインよりも低いことは想定したため，妥当であると考えられる．また，デバイス上でのランキング生成において，利用可能なデータをより多く使ったり，データに基づいて単純なランキング生成と重み付けだけでなく学習をおこなったりする必要があると考えられる.

表 4　ベースラインと提案手法の実験結果

Model	MRR@1	MRR@30	MRR@50	nDCG@1	nDCG@30	nDCG@50
Baseline	0.8351	0.8763	0.8764	0.8351	0.8244	0.8075
Proposal	0.0997	0.2289	0.2289	0.0206	0.3033	0.3033

6　まとめと今後の課題

本論文では，ユーザが自身のデータを推薦システムに提供したくない場合に，ユーザがデータを提供しなくても推薦の恩恵を受けられ，システムを改変せずにユーザの持つデバイス上のみで興味に応じた結果を提示できる手法を提案した．具体的には，既にデータ提供しているユーザの特徴量を集約して特徴に応じて推薦結果を出力する代表アカウントを作成，各ユーザの持つデバイスにおいてデータをシステムに提供せず，代表アカウントへの推薦結果をユーザに応じてデバイス上で組み合わせることで興味に応じた推薦を実現した．実験では推薦システムの評価データセットを用いて，既存手法と提案手法を比較検討し，代表アカウントを利用したオンデバイス推薦が性能にどのような影響を与えるかについて検証した．実験結果として，提案手法がベースラインよりも性能が低いことが性能評価実験から明らかとなった．

今後の課題として，代表アカウントモデルがよりユーザを表現できるような改善やオンデバイス推薦システムのユーザの興味により応じた結果を提示できるような改善，複数ドメインのデータセットを用いた実験による性能検証などが考えられる．また，コールドスタート問題への対策やシーケンシャル推薦への対応も考えられる．

謝辞　本研究は JSPS 科研費 21H03775 の助成を受けたものです．ここに記して謝意を表します．

文　献

[1] Qiang Yang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen, and Han Yu. *Federated Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2019.

[2] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. 2016.

[3] M. G. Sarwar Murshed, Christopher Murphy, Daqing Hou, Nazar Khan, Ganesh Ananthanarayanan, and Faraz Hussain. Machine learning at the network edge: A survey. *ACM Comput. Surv.*, 54(8), oct 2021.

[4] Jie Zhang, Zhihao Qu, Chenxi Chen, Haozhao Wang, Yufeng Zhan, Baoliu Ye, and Song Guo. Edge learning: The enabling technology for distributed big data analytics in the edge. *ACM Comput. Surv.*, 54(7), jul 2021.

[5] Paolo Bellavista, Luca Foschini, and Alessio Mora. Decentralised learning in federated deployment environments: A system-level survey. *ACM Comput. Surv.*, 54(1), feb 2021.

[6] Sin Kit Lo, Qinghua Lu, Chen Wang, Hye-Young Paik, and Liming Zhu. A systematic literature review on federated machine learning: From a software engineering perspective. *ACM Comput. Surv.*, 54(5), may 2021.

[7] Matthias Paulik, Matt Seigel, Henry Mason, Dominic Telaar, Joris Kluivers, Rogier van Dalen, Chi Wai Lau, Luke Carlson, Filip Granqvist, Chris Vandevelde, Sudeep Agarwal, Julien Freudiger, Andrew Byde, Abhishek Bhowmick, Gaurav Kapoor, Si Beaumont, Áine Cahill, Dominic Hughes, Omid Javidbakht, Fei Dong, Rehan Rishi, and Stanley Hung. Federated evaluation and tuning for on-device personalization: System design & applications, 2021, 2102.08503. https://arxiv.org/abs/2102.08503, (accessed 2023-1-11).

[8] Jialiang Han and Yun Ma. *c^3drec*: Cloud-client cooperative deep learning for temporal recommendation in the post-gdpr era. *arXiv.org e-Print archive*, 2021, 2101.05641. https://arxiv.org/abs/2101.05641, (accessed 2023-1-11).

[9] Shuchang Liu, Shuyuan Xu, Wenhui Yu, Zuohui Fu, Yongfeng Zhang, and Amelie Marian. Fedct: Federated collaborative transfer for recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 716–725, New York, NY, USA, 2021. Association for Computing Machinery.

[10] Qinyong Wang, Hongzhi Yin, Tong Chen, Junliang Yu, Alexander Zhou, and Xiangliang Zhang. Fast-adapting and privacy-preserving federated recommender system. *The VLDB Journal*, 31(5):877–896, oct 2021.

[11] Ben Tan, Bo Liu, Vincent Zheng, and Qiang Yang. A federated recommender system for online services. In *Proceedings of the 14th ACM Conference on Recommender Systems*, RecSys '20, page 579–581, New York, NY, USA, 2020. Association for Computing Machinery.

[12] Liu Yang, Ben Tan, Vincent W. Zheng, Kai Chen, and Qiang Yang. *Federated Recommendation Systems*, pages 225–239. Springer International Publishing, Cham, 2020.

[13] Sauptik Dhar, Junyao Guo, Jiayi (Jason) Liu, Samarth Tripathi, Unmesh Kurup, and Mohak Shah. A survey of on-device machine learning: An algorithms and learning theory perspective. *ACM Trans. Internet Things*, 2(3), jul 2021.

[14] Cong Wang, Yifeng Zheng, Jinghua Jiang, and Kui Ren. Toward privacy-preserving personalized recommendation services. *Engineering*, 4(1):21–28, 2018. Cybersecurity.

[15] Renjie Gu, Chaoyue Niu, Yikai Yan, Fan Wu, Shaojie Tang, Rongfeng Jia, Chengfei Lyu, and Guihai Chen. On-device learning with cloud-coordinated data augmentation for extreme model personalization in recommender systems, 2022.

[16] Jialiang Han, Yun Ma, Qiaozhu Mei, and Xuanzhe Liu. Deeprec: On-device deep learning for privacy-preserving sequential recommendation in mobile commerce. In *Proceedings of the Web Conference 2021*, WWW '21, page 900–911, New York, NY, USA, 2021. Association for Computing Machinery.

[17] Sugam Garg, Harichandana SS, and Sumit Kumar. On-device document classification using multimodal features. In *Proceedings of the 3rd ACM India Joint International Conference on Data Science & Management of Data (8th ACM IKDD CODS & 26th COMAD)*, CODS-COMAD '21, page 203–207, New York, NY, USA, 2021. Association for Computing Machinery.

[18] Tong Chen, Hongzhi Yin, Yujia Zheng, Zi Huang, Yang Wang, and Meng Wang. Learning elastic embeddings for customizing on-device recommenders. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery*

& Data Mining, KDD '21, page 138–147, New York, NY, USA, 2021. Association for Computing Machinery.

[19] Alessandro Epasto, Hossein Esfandiari, and Vahab Mirrokni. On-device algorithms for public-private data with absolute privacy. In *The World Wide Web Conference*, WWW '19, page 405–416, New York, NY, USA, 2019. Association for Computing Machinery.

[20] Qingqing Cao, Noah Weber, Niranjan Balasubramanian, and Aruna Balasubramanian. Deqa: On-device question answering. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '19, page 27–40, New York, NY, USA, 2019. Association for Computing Machinery.

[21] Mengwei Xu, Feng Qian, Qiaozhu Mei, Kang Huang, and Xuanzhe Liu. Deeptype: On-device deep learning for input personalization service with minimal privacy concern. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(4), dec 2018.

[22] Mikko Honkala and Yanqing Cui. Automatic on-device filtering of social networking feeds. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design*, NordiCHI '12, page 721–730, New York, NY, USA, 2012. Association for Computing Machinery.

[23] Arun D Prabhu, Nikhil Arora, Shubham Vatsal, Gopi Ramena, Sukumar Moharana, and Naresh Purre. On-device sentence similarity for sms dataset. In *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*, pages 140–146, 2021.

[24] Dinghan Shen, Pengyu Cheng, Dhanasekar Sundararaman, Xinyuan Zhang, Qian Yang, Meng Tang, Asli Celikyilmaz, and Lawrence Carin. Learning compressed sentence representations for on-device text processing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 107–116, Florence, Italy, jul 2019. Association for Computational Linguistics.

[25] Ryoma Sato. *Private Recommender Systems: How Can Users Build Their Own Fair Recommender Systems without Log Data?*, pages 549–557.

[26] Ryoma Sato. Retrieving black-box optimal images from external databases. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22, page 879–887, New York, NY, USA, 2022. Association for Computing Machinery.

[27] Aditya Pal, Chantat Eksombatchai, Yitong Zhou, Bo Zhao, Charles Rosenberg, and Jure Leskovec. Pinnersage: Multimodal user embedding framework for recommendations at pinterest. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 2311–2320, New York, NY, USA, 2020. Association for Computing Machinery.

[28] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. Multi-interest network with dynamic routing for recommendation at tmall. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 2615–2623, New York, NY, USA, 2019. Association for Computing Machinery.

[29] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, DLRS 2016, page 7–10, New York, NY, USA, 2016. Association for Computing Machinery.

[30] Jason Weston, Ron J. Weiss, and Hector Yee. Nonlinear latent factorization by embedding multiple user interests. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, page 65–68, New York, NY, USA, 2013. Association for Computing Machinery.

[31] Harrie Oosterhuis and Maarten de Rijke. Sensitive and scalable online evaluation with theoretical guarantees. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, page 77–86, New York, NY, USA, 2017. Association for Computing Machinery.

[32] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 89–96, New York, NY, USA, 2005. Association for Computing Machinery.