クラスタリングを用いたパッシブフィンガープリンティングによる端末分類の試み

市野 雅暉[1,a)]　藤井 達也[1]　渡名喜 瑞稀[1]　神 章洋[1]　升田 尚幸[1]　小玉 直樹[2]　齋藤 孝道[2,3]

概要：
ブラウザフィンガープリンティングにおいて，学習済みの端末の識別はアクセス数が増加すると，推定対象の組み合わせが指数関数的に増加し困難となる．そこで，本論文の提案手法は，クラスタリングアルゴリズムを用いて推定対象のアクセスログの組を削減したのち，2 つのアクセスログの組が同一端末から取得されたか否かを推定し，その結果と Union-Find アルゴリズムを用いて，端末ごとにまとめる．提案手法を検証するために 2 つの実験を行った．1 つ目は，実運用されている Web サイトのアクセスログに対して提案手法を適用した．2 つ目は，推定対象を削減したことによる処理時間と精度への影響を調べるため，前者の実験と同じデータに対して推定対象を削減する処理を省略し実験を行った．結果，10 万件のアクセスログおいて，推定対象を削減したほうが省略したほうに比べて，処理時間が 100 倍以上短縮され，調整ランド指数が約 0.30 改善された．未学習の端末を含むデータに対しても，短時間で端末をまとめることができる可能性を示した．

キーワード：パッシブフィンガープリンティング，ブラウザフィンガープリンティング，クラスタリング

A Proposal for Device-Classification by Passive Fingerprints with Clustering Algorithm

Masaki Ichino[1,a)]　Tatsuya Fujii[1]　Mizuki Tonaki[1]　Akihiro Jin[1]　Naoyuki Masuda[1]
Naoki Kodama[2]　Takamichi Saito[2,3]

Abstract: In browser fingerprints, the identification of learned devices becomes difficult as the number of accesses increases, since the number of combinations to be estimated increases exponentially. The proposed method uses a clustering algorithm to reduce the number of estimation pairs, then estimates whether two access logs were obtained from the same device or not and combines the results with the Union-Find algorithm for each device. The proposed method was applied to the access logs of an actual website, and experiments were also conducted to investigate the impact of the reduction. The results showed that for 100,000 access logs, the processing time was reduced by more than 100 times with the reduced estimation pair compared to the omitted estimation target and the accuracy of the Adjusted Rand index was about 0.30 higher. This result shows that it is possible to quickly summarize devices even when the data contains unlearned devices.

Keywords: Passive Fingerprints, Browser Fingerprints, Clustering

1. はじめに

EC サイトや SNS などの会員制 Web サイトにおいて，さまざまな不正行為が報告されている。たとえば、同一人物が複数アカウントを利用して買い占めを行ったり，サクラ

[1] 明治大学 大学院
Graduate School of Meiji University
[2] 明治大学
Meiji University
[3] レンジフォース株式会社
[a)] ce225003@meiji.ac.jp

行為を行っていることが知られている．そのような利用者は 1 つの端末から複数アカウントを用いて Web サイトにアクセスしていることが多いと考えられ，アクセスが同一端末によるものと判断することができれば，同一人物が複数アカウントを利用していることが検知できる可能性がある．そこで，本論文ではブラウザフィンガープリンティングのうちのパッシブフィンガープリンティングを用いて，Web サイトのアクセスログを端末ごとにまとめる手法を提案する．パッシブフィンガープリンティングとは，Web サイトにアクセスした際に Web サイト側が Web ブラウザから受動的に取得できる情報のみを用いて，アクセスを識別する技術である．

我々 [1], [2] は，2 つのアクセスログの組が同一端末から取得されたか否かを推定する手法を提案し，モバイル端末において F 値が 0.99 以上の精度で推定できることを示した．これを拡張して，考えうる 2 つのアクセスログに対する推定結果に Union-Find アルゴリズム [3] を適用することによって，アクセスログを端末ごとにまとめられると考えられる．ここで用いる Union-Find アルゴリズムは，同一端末から取得されたと推定されたアクセスログをたどり，アクセスログを端末ごとにまとめる操作をする．しかし，アクセスログの組み合わせは件数に対して $O(n^2)$ であるから，アクセスログの件数が多くなるほど推定対象が多くなり困難になる．

そこで，提案手法はこの拡張した方法にアクセスログの組を削減する手順を加える．この方法により，推定対象を削減しつつ，端末をまとめることができると考えた．

2. 関連知識

2.1 ブラウザフィンガープリンティング

ブラウザフィンガープリンティングとは，ブラウザから取得できる情報の組み合わせを用いて，アクセスしてきた端末の識別をする技術である．また，ブラウザから取得できる情報を特徴点といい，その組み合わせをフィンガープリントという．

高橋ら [1] はパッシブフィンガープリンティングの実験を行い，結果として，特徴点の中でも特にグローバル IP アドレスと User-Agent 文字列の情報が識別において有用であることを示した．

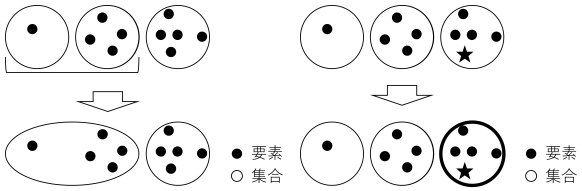
Vastel ら [4] は，時間経過によるフィンガープリントの変化を調査し，時間経過に伴ってフィンガープリントが徐々に変化することを示した．

2.2 パッシブフィンガープリンティング

パッシブフィンガープリンティングとは，ブラウザから特徴点を受動的に取得できるものを利用したブラウザフィンガープリンティングである．本論文ではパッシブフィンガープリンティングを利用する．特徴点としてタイムスタンプ，User-Agent 文字列，グローバル IP アドレスを推定のために用いる．

2.3 Union-Find アルゴリズム

Union-Find アルゴリズム [3] とは，Merge-Find アルゴリズムとも呼ばれ，素結合データ構造において，Union「2 つの集合の和を取る」と Find「要素が属する集合を検索する」の 2 つ操作のことをいう．素結合データ構造とは，全ての集合が互いに共通部分を持たない集合を示すデータ構造である．



(a) Union
集合の和を取る

(b) Find
★が属す集合を探す

図 1: 素結合データに対する Union-Find 操作

3. 提案手法

提案手法は, 以下に示すように，2 つのステップで構成される．

(1) クラスタリングアルゴリズムを用いて推定対象のアクセスログの組を削減する（推定組削減部）

(2) 2 つのアクセスログの組が同一端末から取得されたか否かを推定し，その結果と Union-Find アルゴリズムを用いて，アクセスログを端末ごとにまとめる（推定集約部）

以降，特に断り書きがない限り，2 つのアクセスログの組のことを単に組，この組を作成することを組作成，2 つのアクセスログの組が同一端末から取得されたか否かを推定することを組推定，教師ラベルを用いてアクセスログを端末ごとにまとめた集合のそれぞれを教師クラス，提案手法を用いてアクセスログを端末ごとにまとめた集合のそれぞれを推定クラス，推定組削減部の結果により作られたアクセスログの集合をクラスタと呼ぶ．

提案手法を用いて推定する様子と本論文で定義した語の説明を図 2 に示す．

3.1 推定組削減部

推定組削減部は次の推定集約部の推定対象となる組の削減を行う．

クラスタリングアルゴリズムは類似するデータをまとめる，もしくはデータ同士の関連性を見つけるために用いられる．また，フィンガープリントの特性上，同一端末からのフィンガープリントは類似してることが多い．したがっ

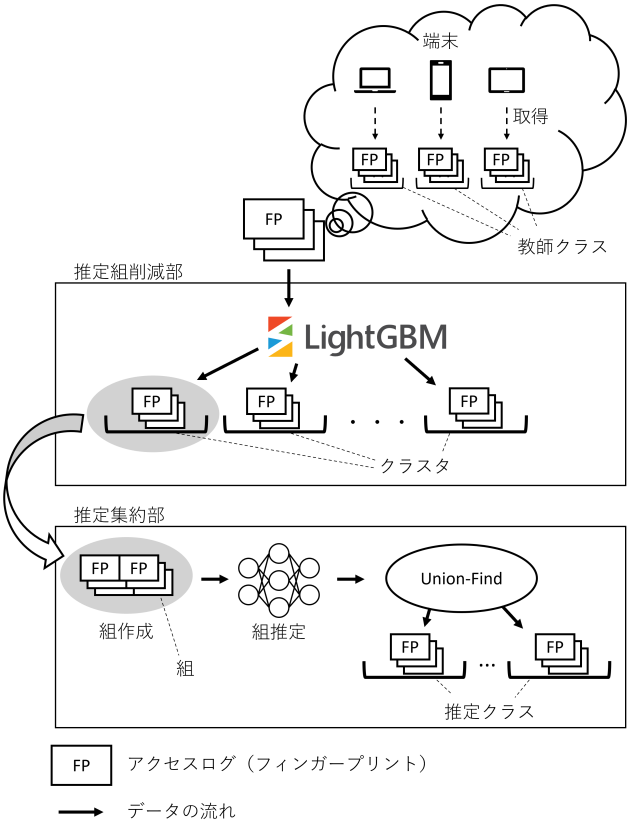


図 2: 提案手法の概念図

て，クラスタリングアルゴリズムにより，同一端末からのアクセスログを同一クラスタに分類することが期待できる．推定組削減部では，クラスタリングアルゴリズムを用いてアクセスログを端末ごとにある程度まとめることによって，次の推定集約部の組数を減らす．

後述の実験 3 では，ここで用いるクラスタリングアルゴリズムの比較，検討を行った．実験 3 の結果に基づいて採用したクラスタリングアルゴリズムは特性により，モデルを学習してからの推定ができない．したがって，学習用データに対するクラスタリングの結果を LightGBM [5] を用いた多クラス分類モデルに学習させる．

3.1.1 学習時

推定組削減部では，クラスタリングアルゴリズムを用いて，推定集約部で推定する組を削減するための機械学習モデルを作成する．実験 3 の結果から，クラスタリングアルゴリズムに凝集型クラスタリングの Ward 法を用いて，クラスタの生成に 3.1.2 節に示す方法を採用した．この採用したクラスタリングアルゴリズムは学習をして，新たに与えられたデータに対し，既存のクラスタのどれに当たるか推定することはできない．したがって，クラスタリングの結果を新たな正解とするような多クラス分類モデルを LightGBM を用いて作成した．その様子を図 3 に示す．

3.1.2 クラスタの生成方法

提案手法では推定組削減部に用いるクラスタリングアルゴリズムに凝集型クラスタリングの Ward 法を用いた．

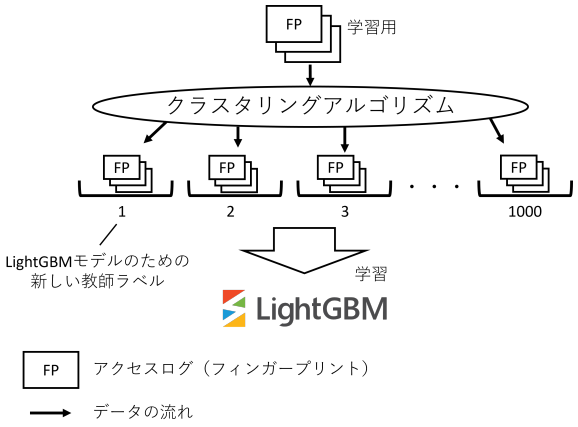


図 3: 推定組削減部の機械学習モデル作成

Ward 法のみではデータ間の距離を測ることしかできず，距離からデータをまとめてクラスタにするアルゴリズムは別に必要である．このアルゴリズムとして，下記に示す手順を採用した．また，この手順を用いて，図 4 にデンドログラムから 6 つのクラスタを生成する様子を示す．

(1) データセットに Ward 法を適用し，デンドログラムを生成する
(2) 独自手法でクラスタを生成する
  (a) デンドログラムを根ノードからたどる
  (b) ノードが持つ葉ノードを 1 つのクラスタとする
  (c) 指定したクラスタ数ができていれば終了
  (d) 作成したクラスタのうち，最も大きいクラスタを生成したノードの子ノードをたどる
  (e) クラスタを子ノードが持つ葉ノードごとに分割し，(c) に戻る

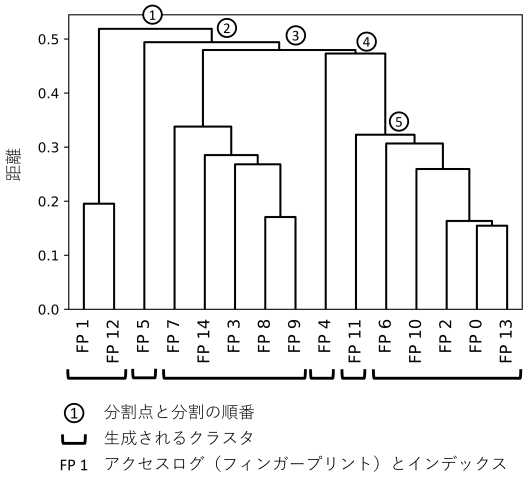


図 4: デンドログラムとクラスタ生成の様子

3.2 推定時

推定には，4.3 節で作成するデータに，以下の操作を行う．推定時は，学習時と異なりクラスタリングアルゴリズムを使わないが，3.1.1 節で作成した多クラス分類モデルは

あたかもクラスタリングアルゴリズムと同じように機能するため，この結果をクラスタと呼ぶ．

(1) 作成した多クラス分類モデルを用いて，推定を行いクラスタを作成する
(2) クラスタごとにデータセットを分ける

3.3 推定集約部

推定集約部では，推定組削減部の推定結果であるクラスタごとに組作成を行い，より詳細な推定を行い最終的な結果を導く．1 つの深層学習モデルを組推定に用い，その結果を端末ごとにまとめる操作に Union-Find アルゴリズムを用いる．なお，先に示した通り，2 つのアクセスログの組が同一端末から取得されたか否かを推定することを組推定と呼んだ．

組推定で用いる機械学習モデルは，北條ら [2] の方法と同様に作成したものを用いる．

3.3.1 推定時

推定時は，推定組削減部の結果である各クラスタ内で組作成を行う．つまり，推定組削減部はここで作成される組数を削減する．

推定時の手順を以下に示す．

また，推定結果を用いて Union-Find アルゴリズムでまとめる様子を図 5 に示す．この図について，例えば，3 列目に示す FP1 と FP2，FP3 と FP4 の推定クラスがあったとき，FP4 と FP1 が同一端末と判断された場合，この 2 つの推定クラスに対して Union（結合）の操作をして，4 行目に示す F1，FP2，FP3，FP4 の推定クラスが作成される様子を示している．

(1) 推定組削減部の結果による各クラスタ内のアクセスログに対して，4.4 節に示す組作成をする
(2) 作成した深層学習モデルに組推定をさせる
(3) 推定結果を用いて Union-Find アルゴリズムでまとめる
  (a) アクセスログそれぞれを全て別の端末と見なす
  (b) 推定結果が「同一端末から取得された」である 2 つのアクセスログに対して Union の操作をし，推定クラスを作成する

4. データセットとベクトルデータの作成

4.1 データセットの分割

収集期間は 8 日間とし，期間内で平均 10 分に 1 回以上アクセスがあった端末からのアクセスログを正常にログが取得できなかったものとみなして削除した．また，タイムスタンプ，User-Agent 文字列，グローバル IP アドレスのいずれかに欠損値があるものも正常にログが取得できなかったとして削除した．

学習用データセットに 1 日目から 7 日目の 7 日間のアクセスログを用い，テスト用データセットに 8 日目のアクセスログを用いた．学習用データセットは前半 7 日間の

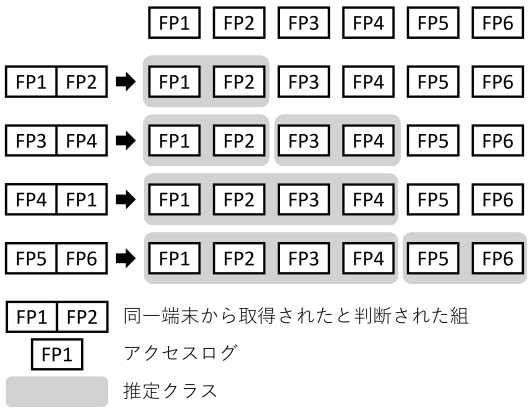


図 5: 推定結果を用いて Union-Find アルゴリズムでアクセスログを上から順にまとめる様子

アクセスログから無造作に 22,520,449 件抽出したものを推定集約部の学習用データセット，さらにそこから無造作に 100,000 件抽出したものを推定組削減部の学習用データセット及び実験 3 のデータセットとした．使用したサーバのメモリ制限により，推定組削減部と実験 3 で用いるクラスタリングアルゴリズムは全てのアクセスログをクラスタリングできないため，100,000 件とした．実験 1，2 で使用するテスト用データセットは，最終日の 8 日目のアクセスログから 10,000，50,002，100,002 件を無造作に抽出したものを用いた．

表 1: 使用するデータセット

用途	アクセスログ件数	端末数
推定集約部の学習時と実験 3	100,000	98,342
推定組削減部の学習時	22,520,449	10,456,082
テスト時（実験 1，実験 2）	10,000	8,157
	50,002	40,948
	100,002	80,805

4.2 使用する特徴点

実験に使用するフィンガープリントは表 2 に示す特徴点である。教師ラベルは，端末識別子とする．端末識別子は HTTP Cookie にランダムな値を付与し，他の端末を区別できるようにしたものであり，これを教師ラベルとして利用した．

4.3 推定組削減部で用いるデータの作成

推定組削減部で用いるデータは，User-Agent 文字列，グローバル IP アドレスと表 3 に示す値である．表 3 に示す特徴点は，表 2 の特徴点の値を利用して，北條らの方法 [2] と同様に作成した．

表 2: フィンガープリントと端末識別子の例

特徴点	例
タイムスタンプ	2022-01-01 12:00:00
User-Agent 文字列	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.5112.81 Safari/537.36 Edg/104.0.1293.47
グローバル IP アドレス	133.26.81.168
端末識別子	glF0leOtAZEat...

表 3: 新たに作成する特徴点

元の特徴点	新たに作成する特徴点
User-Agent 文字列	OS，Web ブラウザそれぞれの名前，バージョン，メジャー・マイナー及びメンテナンスバージョン
	機種名，機種ブランド名
グローバル IP アドレス	第 1 オクテット，第 2 オクテット，第 3 オクテット，第 4 オクテット，ISP 名，緯度，経度，国名，都市名，市区町村名

4.4 推定集約部で用いるデータの作成

推定集約部では，表 2 に示した特徴点を用いて，北條らの方法 [2] と同様に新たな特徴点の生成し，2 つのアクセスログの組を作成したものを用いる．その様子を図 6 に示す．

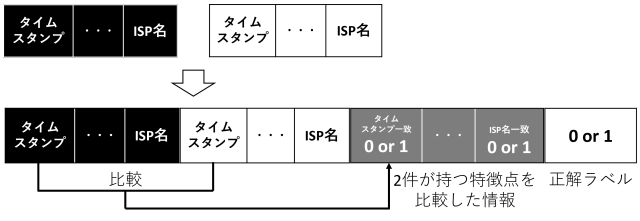


図 6: アクセスログの組の作成と新たな特徴点

5. 実験

5.1 評価指標

実験 1，2 に用いる評価指標は，調整ランド指数と完全正解率，間違い端末なし正解率，取りこぼし端末なし正解率，部分正解率を利用する．なお，完全正解率，間違い端末なし正解率，取りこぼし端末なし正解率，部分正解率は，独自に定義したものである．

5.1.1 調整ランド指数

調整ランド指数は，ランド指数から無造作に予測した結果から得られるランド指数を引いたクラスタリングの評価指標である．値が大きいほど正しい予測が行えていると捉えることができる．

ランド指数はクラスタリングの評価指標の一つであり，0 から 1 の値を取る．1 に近いほどクラスタリングの予測結果が教師ラベルに近く，0 に近づくにつれてどのクラスタにも同じ教師ラベルを持つデータが少なくなることを示す．

5.1.2 完全正解率

完全正解率とは，独自に定めた指標で，正しく端末分類できた割合を示す．例えば，図 7 に示すように，教師クラスと推定クラスが一致しているとき，正解と見なす．教師クラスの数に対して，教師クラスと推定クラスが同一であるものの数の割合である．1 が最も良く，0 が最も悪いと言える．

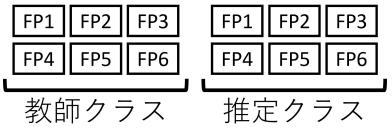


図 7: 完全正解の例

5.1.3 間違い端末なし正解率

間違い端末なし正解率とは，独自に定めた指標で，正しく端末分類ができた割合を示す．ただし，完全正解率に加えて，教師クラスが複数の推定クラスに分けられていても，全ての推定クラスに他の教師クラスのアクセスログ，つまり，他の端末からのアクセスログが含まれていない場合はその教師クラス対して正しく予測できたと見なす．図 8 のように推定クラスが複数あっても，教師クラスに対して数え上げるから，図のような状態で間違い端末なし正解が 1 つあることになる．この指標は，他の端末を間違って同一と見なすことを良しとしないケースに適する．

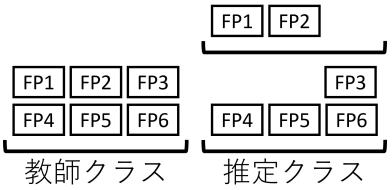


図 8: 間違い端末なし正解の例

5.1.4 取りこぼし端末なし正解率

取りこぼし端末なし正解率とは，独自に定めた指標で，正しく端末分類できた割合を示す．ただし，完全正解率に加えて，推定クラスに他の教師クラスのアクセスログ，つまり，他の端末からのアクセスログが含まれていても，推定クラスが教師クラスを包含している場合は正解と見なす．この様子を図 9 に示す．

この指標は，同一端末からのアクセスログを全て取りこぼしたくないケースに適する．例えば，不正なアクセスを見落としなくまとめたいときに用いられると考えられる．

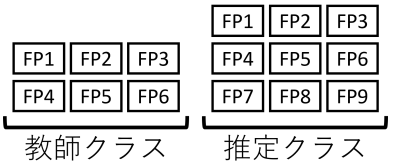


図 9: 取りこぼし端末なし正解の例

5.1.5 部分正解率

教師クラスの数に対して，完全正解率，間違い端末なし正解率，取りこぼし端末なし正解率のいずれかの正解の基準に当てはまるものの割合である．

5.2 実験環境

実験に利用したサーバの仕様と主なライブラリのバージョンを表 4 と表 5 に示す．

表 4: サーバの仕様

CPU	Intel Xeon Silver 4214R
Memory	192GB
GPU	NVIDIA Tesla V100 32GB

表 5: 主なライブラリのバージョン

名称	バージョン
Python	3.7.10
Keras	2.4.3
LightGBM	3.3.2
Numpy	1.19.2
pandas	1.2.4
pyisp[6]	0.2.2
scikit-learn	0.24.2
SciPy	1.6.3
TensorFlow	2.4.0
user_agents [7]	2.2.0

5.3 実験 1

実験 1 では, 提案手法の精度と処理時間を調べる目的で，テスト用データセットに提案手法を適用する．テスト用データセットは 4 節に示したように，8 日間の収集期間の最終日にあたる 8 日目のデータを利用する．また，使用するアクセスログの数に応じた精度や処理時間を比較することを目的として，10,000 件，50,002 件，100,002 件のアクセスログを無造作に抽出して新たなテスト用データセットとしている．

5.4 実験 2

実験 2 では，提案手法の推定組削減部による精度や処理時間に対する影響を調べる目的として，推定組削減部を省略して推定集約部のみを実験 1 と同じデータセットに適用する．

5.5 実験 3

実験 3 では，推定組削減部の学習時に用いる最適なクラスタリングアルゴリズムの比較，検討を行う．推定組削減部における最適な結果とは，各クラスタに含まれるアクセスログの数に偏りがなく，同一端末からのアクセスログが同一のクラスタに存在する状態である．一般的に，n 個の要素を m 個の集合に分け，分けた集合内で全ての要素の組み合わせを考える場合，要素を均等に分けるときが最も組み合わせの合計が少なくなる．同様に推定集約部では同一クラスタ内に含まれるアクセスログの組をすべて作成するので，クラスタサイズに偏りが小さいとき，推定集約部で作成される組数が減少し，推定集約部の推定時間短縮が期待できる．

学習用データセットは，4 節に示したものを用いる．比較対象のクラスタリングアルゴリズムとして，k-means 法と凝集型クラスタリングの Ward 法，群平均法，最短距離法，最長距離法を用いた．凝集型クラスタリングはクラスタの生成方法を指定する必要があるが，この実験では，scikit-learn のデフォルト値を用いてクラスタリングを行った．ただし，Ward 法を用いた凝集型クラスタリングに対しては，3.1.2 節に示した独自のクラスタ生成法を用いる実験も行った．データセットは，4 節に示したものを用い，作成するクラスタ数は 1,000 である．

実験では，調整ランド指数，クラスタサイズの標準偏差，1 つの端末からのアクセスログが含まれるクラスタ数，クラスタサイズの分布を調べた．1 つの端末からのアクセスログが含まれるクラスタ数を調べることによって，同一端末からのアクセスログが同一のクラスタ内に存在する状態になっているか調べることができる．クラスタ数は 1 だと良く，2 以上は大きいほどより悪い．例えば，クラスタ数が 2 で端末数が 3 の場合は，アクセスログが 2 つのクラスタに分けられている端末は 3 つあることを示す．

6. 実験結果

6.1 実験 1

実験 1 の結果を表 6 に示す．表 6 より推定対象のアクセスログ件数が少なくなるにつれて精度が高くなっている．また処理時間に関しては，アクセスログ件数が増えるにつれて大きくなることが予想されたが，100,002 件のアクセスログを使った実験のほうが，50,002 件のものに比べて，作成された組数が多いのにも関わらず 30 秒短い結果となった．

6.2 実験 2

実験 2 の結果を表 7 に示す．表より，推定対象のアクセスログ件数が増加するにつれて処理時間が大幅に増えてい

表 6: 実験 1 の結果

アクセスログ件数	作成された組数	調整ランド指数	完全正解率	間違い端末なし正解率	取りこぼし端末なし正解率	部分正解率	処理時間 [s]
10,000	187,393	0.3783	0.8303	0.9092	0.9076	0.9864	1,738
50,002	2,855,972	0.3142	0.7969	0.8613	0.9175	0.9820	2,088
100,002	9,841,949	0.2978	0.7750	0.8377	0.9183	0.9810	2,058

ることがわかる．アクセスログ件数が 50,002 件，100,002 件での調整ランド指数は 10,000 件での結果に比べて 6%以下の精度である．しかし，独自指標での比較においては調整ランド指数より大きな差はない結果で，例えば，完全正解率は 10,000 件に比べて 86%以上の精度であった．

6.3 実験 3

実験 3 の結果を表 8，9，図 10 に示す．表より凝集型クラスタリングの Ward 法及び k-means 法が他に比べて，調整ランド指数の値が高いことがわかる．同様に，クラスタサイズの標準偏差が他に比べて小さいことから，クラスタに偏りが比較的少ないと言える．

図 10 より凝集型クラスタリングの最短距離法を使った場合は，同一端末からのアクセスが多数のクラスタに分けられることは少ない．また，クラスタサイズの標準偏差の値が大きく，表 9 より 1 つのクラスタのみが 10,000 に近い数の端末を含んでいることがわかる．

提案した独自のクラスタ生成法について，調整ランド指数及びクラスタサイズの標準偏差が比較的良い k-means 法とクラスタ生成方法に scikit-learn のデフォルトを用いた Ward 法に比べてどちらの指標も値が悪化するものの，比較的低い群平均法，最短距離法，最長距離法に比べて大きく上回った．

一方で，表 9 から，独自のクラスタ生成法を用いることで最大クラスタサイズが 423 から 195 と約 1/2 に減少していることがわかる．また，最短距離法のように 1 つのクラスタにほとんどのデータが分けられることはなかった．

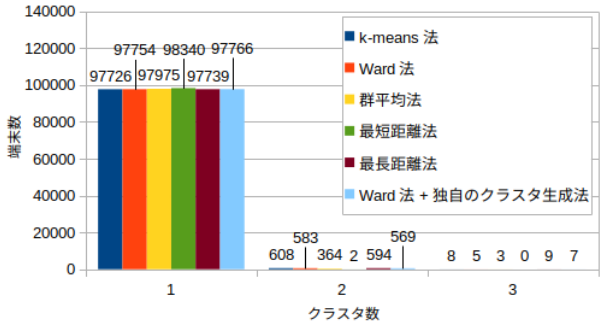


図 10: 1 つの端末からのアクセスログが含まれるクラスタ数

7. 考察

表 6，表 7 より，推定組削減部を用いた場合，調整ランド指数，完全正解率，間違い端末なし正解率は向上し，取りこぼし端末なし正解率は低下したことがわかる．取りこぼし端末なし正解率が低くなる理由としては，アクセスログを分けた際に，同一端末からのアクセスが異なるクラスタに分けられるからだと考えられる．しかし，全体の評価指標を比較すると推定組削減部は精度向上に役立っていると言える．また，処理時間はおおよそ 1/10 以下となるため，処理時間の面においても有効に作用している．

実験 3 の結果について，表 9 より独自の方法で作成したクラスタは既存のものに比べ，最大クラスタサイズが約 1/2 に縮小した．推定時も同様の割合でアクセスログが分けられると考えると，最大クラスタサイズが小さくなることは推定時も同様に最大クラスタサイズが小さくなると考えられる．推定集約部では作成されたクラスタ内で考えうる全ての組作成をするが，組み合わせは $O(n^2)$ であることを考慮すると，クラスタに含まれるアクセスログの件数が 1/2 になれば，組数は約 1/4 となるため，最大クラスタサイズの縮小は処理時間への影響が大きいと考えられる．また，調整ランド指数は独自のクラスタ生成方法を用いない最良のものに比べて 92.6%の精度となっているが，比較的低い結果となった群平均法，最短距離法，最長距離法のなかで最も良い群平均法と比べても 257.1%上回っている．クラスタサイズの偏りも同様である．以上より，最大クラスタサイズの縮小と相対的な精度を考慮すると，独自のクラスタ作成法の方が優れていると考えられる．

8. 今後の課題

本研究では，推定組削減部においてクラスタリングアルゴリズムを用いて，組の削減を行った．今回提案した手法には 2 つの課題がある．

1 つ目は，同一端末におけるフィンガープリントの変化によって精度が低下する点である．短い期間においては，同一端末からのフィンガープリントが類似している可能性が高い．そのため，クラスタリングアルゴリズムの特性上，同一端末からのアクセスログをまとめるには適してるため，実験では高い精度となったと考えられる．しかし，期間を長くすると同一端末であってもフィンガープリントが変化する可能性が高まるため，期間を長くして提案手法の有用性を確認する必要がある．

2 つ目は，推定組削減部において同一端末からのアクセスが別端末と推定される点である．特徴点ごとの重みづけ

表 7: 実験 2 の結果

アクセスログ件数	作成された組数	調整ランド指数	完全正解率	間違い端末なし正解率	取りこぼし端末なし正解率	部分正解率	処理時間 [s]
10,000	49,995,000	0.1618	0.6683	0.7179	0.9349	0.9846	15,410
50,002	1,250,075,001	0.0097	0.6158	0.6563	0.9465	0.9871	187,297
100,002	5,000,150,001	0.0005	0.5807	0.6207	0.9488	0.9888	223,670

表 8: 調整ランド指数とクラスタサイズの標準偏差

アルゴリズム	調整ランド指数	クラスタサイズの標準偏差
k-means 法	0.0003503	48.14
Ward 法	0.0003419	54.56
群平均法	0.0000113	652.90
最短距離法	-0.0000082	2,972.12
最長距離法	0.0001263	154.27
Ward 法＋独自のクラスタ生成法	0.0003247	61.00

表 9: クラスタサイズの分布

アルゴリズム	min	25%	50%	75%	max
k-means 法	1	73	94	122.25	423
Ward 法	1	66	91	124	547
群平均法	1	1	4	14	12,262
最短距離法	1	1	1	1	95,638
最長距離法	1	9	49	115	1,404
Ward 法+独自クラスタ生成法	1	63	116	146	195

を行うことや，別端末として分けられてしまったアクセスログの扱いを検討する必要がある．

また，推定集約部での精度向上として深層学習モデルの作成法の改善があげられる．本研究での推定集約部で用いる深層学習モデルは 1 つであった．クラスタリングアルゴリズムによって分けられたクラスタには類似したフィンガープリントが入っている．そのためクラスタごとに深層学習モデルを作成した場合，端末ごとの厳密な違いをより学習する可能性があり，精度向上が見込まれる．

9. まとめ

本論文では，Web サイトのアクセスログを端末ごとにまとめる手法を提案した．クラスタリングアルゴリズムによる組の削減は良い方向に働き，組の削減を行わないものに対して，100,002 件のアクセスログにおいて，約 108 倍短い時間で推定ができ，調整ランド指数，完全正解率，間違い端末なし正解率が改善された．

10. 研究倫理

我々は，Menlo report [8] の精神に則り倫理的配慮をして実験を行った．実験を行う際，個人識別はせずプライバシーを遵守した．論文中では，オリジナルデータの統計的処理によりオリジナルデータについての推察をされることがないようにした．また，研究に使用したデータセットは，学術的な目的にのみ使用し，我々の管理下で厳重に保管されており，他者への提供をしない．

謝辞

本研究の成果の一部は，JSPS 科研費 18K11305 の助成を受けたものです．また，本研究はレンジフォース株式会社の支援により実施しています．

参考文献

[1] 高橋和司, 安田昂樹, 種岡優幸, 田邉一寿, 細谷竜平, 野田隆文, 齋藤祐太, 小芝力太, 齋藤孝道. HTTP ヘッダのみを用いた Browser Fingerprinting の考察. 暗号と情報セキュリティシンポジウム 2018, 2018.

[2] 北條大和, 齋藤祐太, 齋藤孝道. 深層学習を用いたパッシブフィンガープリンティング手法の提案と実装. コンピュータセキュリティシンポジウム 2019 論文集, Vol. 2019, pp. 252–259, 10 2019.

[3] Bernard A. Galler and Michael J. Fisher. An improved equivalence algorithm. *Commun. ACM*, Vol. 7, No. 5, p. 301–303, may 1964.

[4] Antoine Vastel, Pierre Laperdrix, Walter Rudametkin, and Romain Rouvoy. Fp-stalker: Tracking browser fingerprint evolutions. In *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 728–741, 2018.

[5] Microsoft Corporation. LightGBM. https://lightgbm.readthedocs.io.

[6] pyisp. https://github.com/ActivisionGameScience/pyisp/. (Accessed on 08/20/2022).

[7] user_agent. https://github.com/selwin/python-user-agents. (Accessed on 08/20/2022).

[8] Erin Kenneally and David Dittrich. The menlo report: Ethical principles guiding information and communication technology research. *Available at SSRN 2445102*, 2012.