ウェブ広告閲覧履歴を用いたユーザ属性の推定

崔　洙瑚† 木村　塁†† 南川　敦宣†† 黒柳　茂††† 申　吉浩†††† 大島　裕明†

† 兵庫県立大学 応用情報科学研究科　〒 650–0047 兵庫県神戸市中央区港島南町 7–1–28
†† 株式会社 KDDI 総合研究所　〒 102–8460 東京都千代田区飯田橋 3–10–10
††† Supership 株式会社　〒 107–0062 東京都港区南青山 5–4–35
†††† 学習院大学　〒 171–8588 東京都豊島区目白 1–5–1
E-mail: †{aa19e505,ohshima}@ai.u-hyogo.ac.jp, ††{ui-kimura,at-minamikawa}@kddi-research.jp,
†††shigeru.kuroyanagi@supership.jp, ††††yoshihiro.shin@gakushuin.ac.jp

あらまし　本研究では，ウェブ広告閲覧履歴の情報から，性別，年齢，職業といったユーザ属性を推定する手法を提案する．ウェブ広告のプロバイダは，あるウェブユーザがどのようなサイトを閲覧しているかということをある程度知ることが可能である．一方で，別の情報源との突合を行わない限り，そのユーザが誰であるかを特定できず，どのような属性を持つ人かといったことも知ることができない．ユーザ属性である性別，年齢，職業などを推定することができれば，ウェブ広告の効率を上げることが可能になると考えられる．そこで，本研究では，ウェブ広告閲覧履歴のみを用いて，どのようなホストに何度ぐらい訪れたかといった情報や，ホストどうしの類似性などを考慮して，ウェブユーザの特徴量を作成し，それを基にユーザ属性を推定する手法を提案する．

キーワード　ユーザ属性推定，ウェブ広告，ウェブ閲覧履歴

1　はじめに

近年，インターネットの利用者は，毎日のように様々なウェブサイトにアクセスし，ウェブページを閲覧している．その際には，ウェブ検索サイトやポータルサイトを利用することや，SNS からのリンクを利用することでウェブサイトにアクセスすることになる．個人が開設したものにしろ，企業が開設したものにしろ，多くのウェブサイトにおいて，広告が設置されていることがあり，ウェブユーザは日々それらの広告を目にすることになる．

ウェブ広告は，ウェブサイト運営者が広告プロバイダによる広告枠をウェブページに設置することで表示されるようになる．ウェブページにアクセスが行われた時点で，広告プロバイダが閲覧者と広告提供者の間でのマッチングを行い，どのような広告が提供されるかが決定される．

ウェブ広告のマッチング手法には様々なものが存在する．たとえば，コンテンツマッチと呼ばれる，検索キーワードや広告枠が設置されたウェブページの内容と，広告の内容の関連度を高いものにするというものや，ユーザのウェブ行動履歴に基づいてユーザとマッチングさせるものなどが存在している（図 1）．

ユーザにより適したウェブ広告を発信するためには，性別，年齢，職業といったユーザ属性を取得することが必要となる．

ユーザ属性を取得するためには，なんらかのウェブサイトにおいて，ユーザから直接ユーザ属性を取得することや，ユーザのウェブ行動履歴から推定することなどが考えられる．ユーザのウェブ行動履歴としては，検索エンジンにどのような検索キーワードを入力したか，どのようなウェブページを閲覧したか，ウェブショッピングサイトでどのような商品を購入したかといった情報が考えられる．しかし，広告プロバイダが入手することができる情報には限度があり，これらの全てを入手することは難しい．

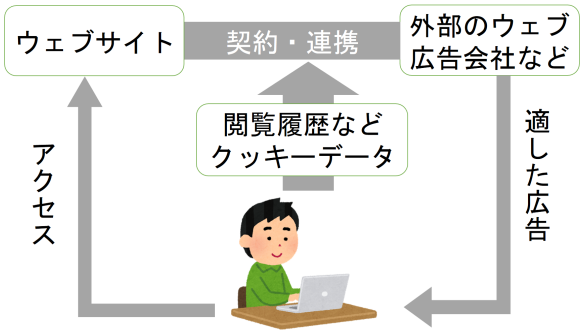

図 1　ターゲティング広告配信の仕組み

そこで，本研究では，広告プロバイダが現在取得することが可能な，ウェブ広告閲覧履歴のみを用いて，性別，年齢，職業といったユーザ属性を推定する手法を提案する．ウェブ広告閲覧履歴は，あるユーザが，いつ，どのウェブページを閲覧したかという情報である．すべてのウェブページの閲覧履歴が得られるわけではなく，その広告プロバイダによるウェブ広告が掲載されたウェブページについてのみの閲覧履歴である．

以下，2 節では関連研究について述べる．3 節では，本研究で用いたデータと問題定義を行う．4 節では，ウェブ広告閲覧履歴からユーザの特徴ベクトルを作成する複数の手法を提案す

る．5 節では，作成された特徴ベクトルを用いて，いくつかの機械学習の分類器を利用したユーザ属性の推定を行う実験について説明する．6 節でまとめと今後の課題について述べる．

2 関 連 研 究

星ら [1] は，本研究と同様にウェブ広告閲覧履歴を用いてユーザのライフイベントの予測を行う手法を提案している．ライフイベントとは，結婚や出産といった生活上の大きなイベントのことである．ユーザはライフイベントの前には，そのイベントに関連した情報を収集すると考えられる．そこで，星らは，ウェブ広告閲覧履歴において，URL を語とみなして Word2Vec を適用することで，ある URL を固定長のベクトルで表現することを行った．あるユーザのウェブ広告閲覧履歴に含まれる URL を学習した Word2Vec のモデルを用いてベクトル化し，それらの平均値をとることでユーザの特徴ベクトルを生成した．そして，生成されたユーザ特徴ベクトルを用いてユーザのライフイベントの予測を行った．

工藤ら [2] は，ユーザが読んだ新聞記事のタイトルからユーザ属性を推定する手法を提案している．ユーザが読む新聞記事にはユーザ属性に応じた傾向が現れると考えられる．そこで，工藤らは Wikipedia を用いて学習された Word2Vec モデルを用いて，ユーザが読んだ新聞記事のタイトルに含まれる語をベクトルにして，それらの平均を用いてユーザ特徴ベクトルを生成した．生成されたユーザ特徴ベクトルから機械学習を用いてユーザの属性推定を行った．

Tagami ら [3] は，ユーザのウェブ閲覧履歴から，広告をクリックするかどうかや，広告主のウェブページを閲覧するかどうかを予測する手法を提案した．ウェブ広告の配信において顧客になりそうなユーザを絞り込むことで広告の効率向上が期待される．Tagami らはウェブ閲覧履歴からウェブページの URL 列を抽出し，Doc2Vec と Word2Vec の手法を用いて URL をベクトル化する手法を提案した．ウェブ閲覧履歴から学習された Doc2Vec と Word2Vec を用いてユーザ特徴ベクトルを生成し，ユーザ特徴ベクトルから広告をクリックするユーザなのか予測を行った．

Kanagasabai ら [4] は，通信業者が取得することができるユーザのウェブ閲覧履歴を用いてユーザの興味を推定する手法を提案している．ウェブ閲覧履歴における URL を Word2Vec によってベクトル化し，それを用いてユーザの特徴ベクトルを生成した．ユーザの興味が推定されると，広告の効率向上などに利用することができると考えられる．

伊藤ら [5] は，Twitter ユーザのユーザ属性を推定する手法を提案している．ツイート集合に加えてユーザプロフィールを用いることや，ブログを行っているユーザの情報を用いて学習のためのデータを増やすといった工夫を行った手法を提案している．他にも Twitter を利用してユーザ属性を推定する研究としては，近藤ら [6] による位置情報付きツイートを用いてユーザの地域クラスタリングを行う研究，Miller ら [7] や Burger ら [8] によるユーザの性別の推定の研究などがある．また，榊ら [9] はユーザの職業の推定を行っている．

3 データと問題定義

本研究では，ウェブ広告閲覧履歴から，性別，年齢，職業といったユーザ属性情報を推定する問題に取り組む．本節では，本研究で用いるウェブ広告閲覧履歴データと，ユーザ属性データについての説明を行い，そのデータを用いたユーザ属性の推定についての問題定義を行う．

3.1 ウェブ広告閲覧履歴データ

本研究では，Supership 株式会社が保有するウェブ広告閲覧履歴を利用する．本ウェブ広告閲覧履歴は，あるユーザが，ウェブブラウザを用いて自由にウェブ閲覧を行っている際に，Supership 株式会社が提供する広告が提示されたウェブページを閲覧したということが記録されたものである．この際，ユーザが誰かということは特定されない．一方で，同じユーザが異なるホストの異なるウェブページを閲覧したことは認識することが可能となっている．

以後，その際に用いられている識別子を，ユーザ識別子と呼ぶこととする．ウェブ広告閲覧履歴には様々な情報が含まれているが，本研究で利用するのは，以下の 3 つ組のデータである．

- ユーザ識別子
- 閲覧日時
- 閲覧 URL

本研究で利用するウェブ広告閲覧履歴は，2015 年 1 月から 2016 年 8 月までに得られたこのような 3 つ組のデータの集合である．

ユーザ識別子は，利用するデバイスが異なったり，利用するデバイスが同じであっても利用するブラウザが異なったりする場合には，異なるものが割り当てられる．そのため，ユーザ識別子が異なったとしても，同一の人物である場合も存在する．また，逆に，複数の人物が，同一のデバイスの同一のブラウザを共用している場合には，ユーザ識別子が複数の人物と対応する場合も存在する．実環境においては，これらの状況については判別することは原則としてできない．そのため，本研究で用いるウェブ広告閲覧履歴では，これらの状況については特段の考慮をしないこする．すなわち，フィルタリングなどは行わず，上記期間に収集されたデータ全てを用いるものとする．

3.2 ユーザ属性データ

ウェブ広告閲覧履歴と対応して，あるユーザ識別子を持つユーザに対して，正解データとなる属性情報が必要となる．本研究ではユーザ属性データとして，5 種類のデータを用いる．以下で，属性の名称と，取り得る値を説明する．

- **性別**

男性, 女性

- **年齢カテゴリ**

12 才〜19 才, 20 才〜24 才, 25 才〜29 才, 30 才〜34 才, 35 才〜39 才, 40 才〜44 才, 45 才〜49 才, 50 才〜54 才, 55 才〜59 才, 60 才以上

- **未既婚**

未婚, 既婚

- **子供の有無**

子供なし, 子供あり

- **職業カテゴリ**

公務員, 経営者・役員, 会社員（事務系）, 会社員（技術系）, 会社員（その他）, 自営業, 自由業, 専業主婦（主夫）, パート・アルバイト, 学生, その他, 無職

これらのユーザ属性データを取得するために，2016 年 9 月に実施したアンケート調査の結果を用いる．本調査は，一般的なウェブアンケート調査であり，そこでは，ユーザ属性データを含む様々な質問に対する回答が行われた．アンケート調査の回答を行う際に，ユーザ識別子が取得されており，それによって，あるユーザ識別子を持つユーザに対するユーザ属性データが取得された．

このアンケート調査では個人が特定されている．一方，先述したとおり，ユーザ識別子は，必ずしも一人のユーザを特定するものではない．実際，アンケート調査における 1 人の個人対して複数のユーザ識別子が対応づけられる場合や，1 つのユーザ識別子に対してアンケート調査での複数の個人が対応づけられる場合が存在した．前者の場合については，ある個人に対応する複数のユーザ識別子から，ランダムに 1 つのユーザ識別子を選択することで，アンケート調査における 1 人の個人に対して 1 つのユーザ識別子を対応させることとした．後者の場合については，当該のユーザ識別子を本研究のユーザ属性データを推定する対象から除外することとした．

このようにして，本研究で用いるウェブ広告閲覧履歴に存在するユーザ識別子のうち，対象のアンケート調査においてユーザ属性データが一意に取得できた 2,132 名のデータが，本研究で用いているユーザ属性データである．

以降，ユーザ識別子によって特定される 2,132 名をユーザと呼ぶこととする．閲覧 URL の件数があまりに少ないユーザは，適切な特徴ベクトルの生成を行うことが難しいと考えられる．そこで，閲覧 URL の件数が 100 件以下のユーザをフィルタリングして本研究の対象から除外することとした．その結果，属性データが得られた 2,132 名のユーザのうち 1,991 名のユーザを，本研究で対象とするユーザとした．

3.3 問題定義

本研究の問題定義を行う．本研究で取り組むのは，ウェブ広告閲覧履歴があるときに，その中に存在するあるユーザ識別子に対して，ユーザ属性データを推定するという分類問題である．

たとえば，ユーザ属性データのうち，**性別**を対象とすると，入力と出力は以下のようになる．

入力　ウェブ広告閲覧履歴，ユーザ識別子

出力　男性 or 女性

この場合，男性または女性の 2 つのカテゴリに分類する二値分類問題となる．同様に，**未既婚**と**子供の有無**についても二値分類問題となる．**年齢カテゴリ**は 10 カテゴリの，**職業カテゴリ**は 12 カテゴリの多クラス分類問題となる．

推定の対象とするユーザ識別子は，ユーザ属性データが存在する 1,991 件のみとする．ウェブ広告閲覧履歴には推定の対象とはならないユーザ識別子に対応するデータが存在しており，それらについてもユーザ特徴の作成のために用いることが可能である．

評価手法については，ユーザ属性ごとに，正解率，適合率，再現率，F 値などで行うものとする．

4 ユーザ特徴ベクトルの作成

4.1 ベクトル作成の概要

本節では，ウェブ広告閲覧履歴からユーザ特徴ベクトルを作成する手法を説明する．

ウェブ広告閲覧履歴から，ユーザごとのウェブページの訪問の有無や訪問頻度の多寡を知ることができる．ユーザ属性が異なれば，興味，関心，ライフステージなどが異なるため，アクセスするウェブページの傾向にも違いがあると考えられる．そこで，どのようなウェブページに何度ぐらいアクセスしたかということを基にして，ユーザ特徴ベクトル作成を行うこととした．

これは，類似するユーザ属性を持つユーザ同士は，類似するウェブページにアクセスするだろうというアイデアに基づくものである．このアイデアがうまく機能する前提としては，複数のユーザが同じウェブページにアクセスしていなくてはならない．しかし，実際には，ウェブページの数は膨大なものであり，一つのウェブページに多くのユーザがアクセスしているわけではない．そこで，ウェブページの URL ではなく，URL から得られるホスト名を用いることとした．すなわち，類似するユーザ属性を持つユーザは，類似するウェブホストにアクセスするだろうというアイデアに基づいて，ユーザ特徴ベクトルの作成を行う．

提案アプローチでは，以下の 3 つのステップでユーザ特徴ベクトルを作成する．

（1） ウェブ広告閲覧履歴からウェブホスト列を取得する．

（2） ウェブホスト列に対して，いくつかの前処理手法を適用する．

（3） 前処理が行われたウェブホスト列に対して，いくつかの重み付け手法を適用し，特徴ベクトルを作成する．

まず，ウェブ広告閲覧履歴から，ユーザごとに，閲覧日時の順番に閲覧 URL を取得する．そこで得られた閲覧 URL をウェブホストに変換することで，ユーザごとに，ウェブホスト列を取得することができる．次に，そのウェブホスト列では，ウェブホストが重複して現れている．それに対して，何も行わないことも含め，いくつかの前処理手法を提案する．その後，前処理手法が適用されたウェブホスト列から特徴ベクトルを作成するためのいくつかの重み付け手法を提案する．

4.2 ウェブ広告閲覧履歴からのウェブホスト列の取得

ウェブ広告閲覧履歴の各レコードは，ユーザ識別子，閲覧日時，閲覧 URL からなる．あるユーザ識別子で識別されるユー

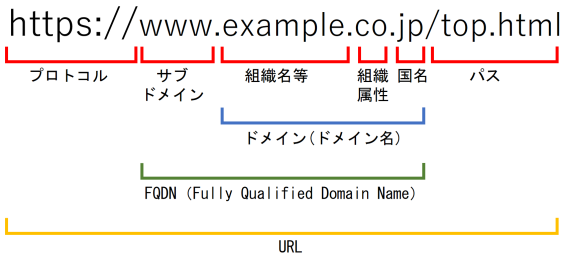

図 2　URL の構造

ザに対して，閲覧日時順に閲覧 URL を取得することで，そのユーザの閲覧 URL 列を取得することができる．URL をウェブホストに変換することによって，ウェブ広告閲覧履歴からウェブホスト列を取得することが実現される．

URL をウェブホストに変換する方法には，いくつかの方法が考えられる．図 2 は，URL の構造を説明するための例を示している．ここでは，「`https://www.example.co.jp/top.html`」という URL が示されている．このとき，いわゆるホスト名とは，「`www.example.co.jp`」の部分のことである．この部分は，FQDN（Fully Qualified Domain Name）と呼ばれる．それに対して，FQDN からサブドメインなどの情報を削除した部分である「`example.co.jp`」が，Registered Domain Name，いわゆる，ドメイン名と呼ばれる部分である．

URL からウェブホストを取得する場合，FQDN を利用する方法，Registered Domain Name を利用する方法などが考えられる．本研究では，予備実験を通して得られた結果から，ドメイン名を用いるよりも，FQDN を用いた場合の方がより良い結果が得られると考えられた．そこで，本研究では，ウェブ広告閲覧履歴における URL から FQDN を取得して，それをウェブホストとして用いることとした．

4.3　ウェブホスト列の前処理手法

ウェブホスト列を観察すると，同じウェブホストが何度も出現することが分かる．これは，同じウェブサイトにおいて，複数のページを次々と閲覧していた場合であると考えられる．同じウェブサイトのいくつかのページを閲覧するということは一般的であり，頻繁に行われると考えられる．ウェブホスト列からユーザの特徴を得るという意味では，そのような，同じホストが重複して現れることをどのように扱うことが適切なのかを明らかにする必要がある．そこで，我々は，以下の 3 種類のウェブホスト列に対する前処理手法を検討した．

（1）　**オリジナル**
（2）　**連続部の集約**
（3）　**集合化**

たとえば，ウェブ広告閲覧履歴から取得されたウェブホスト列が，表 1 においてオリジナルと示したものだった場合を考えてみる．

前処理手法の一つ目は，**オリジナル**をそのまま用いるというものである．この場合は，何の処理も行わないため，ユーザがあるウェブホストに実際に何回アクセスしたかという情報を維持することができる．

前処理手法の二つ目は，**連続部の集約**である．これは，2 回以上同一のウェブホストへのアクセスが連続した場合に，それらを 1 回のウェブホストへのアクセスとして集約してしまう方法である．この場合は，あるウェブホストから，次にどのウェブホストへアクセスしたかといった，ウェブホスト単位の移動の情報を維持することができる．そのため，あるウェブホストに対して，他のウェブホストにアクセスした後にでも何度もアクセスしているといったことを知ることができる．

前処理手法の三つ目は，**集合化**である．集合化とは，あるユーザのウェブホスト列を集合として扱うという方法である．この場合は，あるウェブホストにアクセスしたことがあるかどうかという情報を維持することができる．一方で，アクセス頻度については失われてしまう．

これらの前処理手法を全て試すことで，どれが適切な処理なのかを明らかにする．

4.4　重み付け手法

ウェブホスト列からユーザ特徴ベクトルを作成する手法には様々なものが考えられる．本節では，ウェブホスト列からユーザ特徴ベクトルを作成するための重み付け手法について説明する．我々は，以下の 4 種類の重み付け手法を検討した．

（1）　数え上げ（Count）
（2）　逆ユーザ頻度（IUF）
（3）　Word2Vec と Average Pooling（w2vave）
（4）　Doc2Vec（d2v）

以下で，これらの手法の詳細を説明する．その際，対象となるユーザのウェブホスト列を $H = [h_1, h_2, h_3, \ldots, h_n]$ と表すものとする．

4.4.1　数え上げ（Count）手法

数え上げ（**Count**）手法では，与えられたウェブホスト列において出現したウェブホストを次元として，その次元に対して出現回数を重みとして与える．文書検索における Term Freqency による重み付け手法と同様の手法ととらえると分かりやすいかもしれない．この手法で作成されるベクトルの次元数は，あり得るウェブホストの種類数ということとなるが，現実には，データ中に出現するウェブホストの種類数となる．

Count 手法でのユーザ特徴ベクトル v におけるウェブホスト h_i に対する重み $v(h_i)$ は，ウェブホスト列 H における h_i の出現回数を $\mathrm{h\_freq}(h_i, H)$ と表すと，以下のように表される．

$$v(h_i) = \mathrm{h\_freq}(h_i, H) \tag{1}$$

ウェブホスト列は，事前に前処理が行われており，前処理の方法によって，作成されるユーザ特徴ベクトルの重みは変化することとなる．たとえば，ウェブホスト列の前処理手法として**集合化**が行われた場合，**Count** 手法によるユーザ特徴ベクトルの重みは，対象の次元のウェブホストにアクセスがあった場合には 1，なかった場合には 0 となる．ウェブホスト列の前処理手法が**オリジナル**の場合，**Count** 手法によるユーザ特徴ベクトルの重みは，対象の次元のウェブホストへのアクセス回

表 1　ウェブホスト列に対する前処理手法

オリジナル	[a.com, a.com, b.com, b.com, c.com, a.com, b.com, b.com, b.com, a.com, a.com, a.com]
連続部の集約	[a.com, b.com, c.com, a.com, b.com, a.com]
集合化	{a.com, b.com, c.com}

数となる．

4.4.2　逆ユーザ頻度（IUF）手法

逆ユーザ頻度（**IUF**）手法では，文書検索における TF-IDF 重み付けの IDF の考え方と同様に，多くのユーザにアクセスされているウェブホストの重みを下げる重み付け手法である．**Count** 手法と同様に，出現回数に応じて数え上げを行うが，その際に，IDF 的な重み付けを行う．**IUF** 手法での重み $v(h_i)$ は，ユーザ集合 U における h_i にアクセスしたユーザ数を $\mathrm{u\_freq}(h_i, U)$ と表すと，以下のように表される．

$$v(h_i) = \mathrm{h\_freq}(h_i, H) \cdot \left(\log \frac{|U|+1}{\mathrm{u\_freq}(h_i, U)+1} + 1 \right) \quad (2)$$

なお，このときのユーザ集合 U は，訓練データとして用いる全ユーザとしている．

4.4.3　Word2Vec と Average Pooling（w2vave）手法

Word2Vec と Average Pooling（**w2vave**）による手法は，Word2Vec [10] を用いて，それぞれのウェブホストを固定長のベクトルとして表現し，それらを利用してウェブホスト列をベクトル化する手法である．このときのユーザ特徴ベクトルの次元数は，Word2Vec の次元数と一致する．**w2vave** 手法は，ウェブホスト列に現れるすべてのウェブホストに対して，固定長のベクトルを取得し，それらの平均をしたものをユーザ特徴ベクトルとする手法である．

Word2Vec では，ある単語を規定するのは周辺に現れる単語であるということを仮説とする．本研究では，あるウェブホストを規定するのは，そのウェブホストを訪れる前後に訪れたウェブホストであるということを仮説とする．この仮説に基づき，Word2Vec を用いたウェブホストのエンベディングを行う．仮説が正しければ，類似したホストが類似するベクトルで表現されるということになると考えられる．

通常の Word2Vec では，大量の文書を与えて，単語のエンベディングを行うモデルを学習する．このとき，まず，文書が単語列に変換される．本研究では，大量のウェブホスト列が存在しており，それを Word2Vec で学習することによって，ウェブホストのエンベディングを行うことができる．これは，星ら [1] が用いていた手法と同様の方法である．

作成された Word2Vec のモデルは，あるウェブホスト h_i を与えると，固定長のベクトル $\mathrm{w2v}(h_i)$ を返すものである．ウェブホスト列 H に含まれるウェブホスト h_i が，Word2Vec のモデルに含まれていなかった場合は，本来はベクトルを取得することができないが，ここでは便宜上，$\mathrm{w2v}(h_i)$ としてゼロベクトルが返されるものとする．ウェブホスト列 H からユーザ特徴ベクトルを作成するにあたって，まず，Word2Vec のモデルでベクトル化できないウェブホストを全て取り除いたウェブホスト列 H' を生成する．その上で，**w2vave** 手法でのユーザ特徴ベクトル v は，以下のように表される．

$$v = \frac{1}{|H'|} \sum_{i}^{n} \mathrm{w2v}(h_i) \quad (3)$$

ただし，$|H'|$ はウェブホスト列 H' の長さとする．

4.4.4　Doc2Vec（d2v）手法

Word2Vec が，単語のエンベディングを行うのに対して，Doc2Vec [11] は，文書のエンベディングを行う手法である．そこでの文書とは単語列のことである．よって，前節で述べた Word2Vec のモデル学習と同様にして Doc2Vec のモデル学習を行うことで，あるユーザのウェブホスト列に対するベクトルを得ることができるようになる．

作成された Doc2Vec のモデルは，あるウェブホスト列 H を与えると，固定長のベクトル $\mathrm{d2v}(H)$ を返すものである．**d2v** 手法でのユーザ特徴ベクトル v は，以下のように表される．

$$v = \mathrm{d2v}(H') \quad (4)$$

ただし，H' は Doc2Vec のモデルでベクトル化できないウェブホストを H から全て取り除いたウェブホスト列 H' である．

4.5　Word2Vec と Doc2Vec のモデル学習

ウェブ広告閲覧履歴には，推定の対象となるユーザの情報以外にも，大量のユーザの情報が存在している．用いたデータは 2016 年 4 月から 2016 年 8 月のウェブ広告閲覧履歴である．それらから，ユーザごとにウェブホスト列を作成し，Word2Vec や Doc2Vec のモデル学習を行った．ウェブ広告閲覧履歴からウェブホスト列の取得については，先述した方法と同様に行った．さらに，ウェブホスト列の前処理手法で説明した，**連続部の集約**によってウェブホスト列の前処理を行った．

学習には Skip-gram を用い，ベクトルの次元数とウインドウサイズについては，以下のように複数の設定を用いた．

- 次元数 $= \{128, 256, 512\}$
- ウインドウサイズ $= \{3, 5, 10\}$

これらの組み合わせにより，9 種類のモデルが作成された．Word2Vec と Doc2Vec の実装には gensim を使用した．

5　評 価 実 験

5.1　実 験 設 定

ウェブホスト列の前処理手法と重み付け手法を組み合わせることで，様々な方法でユーザ特徴ベクトルが作成される．それらの中で，どのような組み合わせによるユーザ特徴ベクトルがユーザ属性データの推定に有用なベクトルであるかを明らかにするための実験を行った．

ウェブホスト列の前処理手法は，3 節で説明した 3 種類を試す．重み付け手法は，**Count**，**IUF** の 2 種類と，**w2vave** でモデル学習の際のパラメータの違いによる 9 種類，**d2v** でモデ

ル学習の際のパラメータの違いによる 9 種類の合計 20 種類を試す．これらの組み合わせにより，60 種類のユーザ特徴ベクトルの作成方法を試すこととなる．

実験に先立って，1,991 名のユーザのデータを訓練データとテストデータに 8:2 に分割を行った．ユーザ特徴ベクトルの作成において，**Count** や **IUF** では，訓練データに出現したウェブホストのみがベクトルの次元として扱われることとして，テストデータにしか出現しないウェブホストは訓練時には未知であるとして，テストデータをベクトル化する際には無視することとした．また，**IUF** におけるユーザの全体集合は，訓練データにおけるユーザのみとして，テストデータについては完全に未知であると扱った．

機械学習の分類器としては，以下の 3 種を用いた．

- ロジスティック回帰
- SVM
- XGBoost

分類器のハイパーパラメータ決定のためには，訓練データにおけるグリッドサーチを行う．グリッドサーチでは，訓練データにおける 5 分割交差検証を行い，正解率（accuracy）が最大となるパラメータを決定した．グリッドサーチで試すハイパーパラメータは，以下のとおりである．

ロジスティック回帰

- $C = \{0.001, 0.01, 0.1, 1, 10, 100\}$

SVM

- カーネル $= \{RBF\}$
- $C = \{0.001, 0.01, 0.1, 1, 10, 100\}$

XGBoost

- 学習率 $= \{0.05, 0.1\}$
- 木の深さの最大値 $= \{4, 10\}$
- サブサンプルデータの抽出割合 $= \{0.1, 1\}$

グリッドサーチによって得られたパラメータで訓練データ全体を用いて学習を行い，分類器のモデル構築を行った．

テストデータに対してユーザ属性データの推定を行い，推定結果の評価を行った．

5.2 結　　果

性別の推定の結果

表 2 は，性別の推定の結果を示している．テストデータの性別の割合は男性 51%，女性は 49%である．正解率は最良の場合に 0.844 となった．その際のユーザ特徴ベクトルの作成方法は，ウェブホスト列の前処理手法に**集合化**を用い，重み付け手法に **w2vave** を用いたものであった．Word2Vec のモデル学習における設定は，次元数が 256，ウインドウサイズが 5 であった．分類器には，ロジスティック回帰が用いられ，グリッドサーチによってパラメータ C は 10 と決められた．表では，3 種類のウェブホスト列処理手法と 3 種類の分類器のあらゆる組み合わせにおいて，最も高い正解率が得られた重み付け手法の結果を示している．

年齢カテゴリの推定の結果

表 3 は，年齢カテゴリの推定において，最も正解率が高くなった場合の，各種パラメータを示している．正解率は 0.315 であった．その際のユーザ特徴ベクトルの作成方法は，ウェブホスト列の前処理手法に**集合化**を用い，重み付け手法に **IUF** を用いたものであった．分類器には，SVM 回帰が用いられ，グリッドサーチによってパラメータ C は 10 と決められた．図 3 は，結果の混同行列を示している．おおむね，対角線上に値が集まっていることが見て取れる．

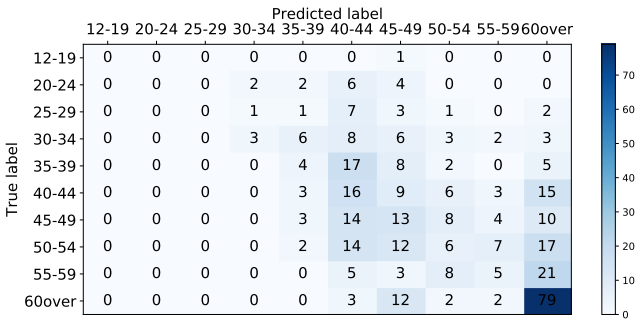

図 3　年齢カテゴリ推定の最適結果の混同行列

未既婚の推定の結果

表 4 は，未既婚の推定において最も正解率が高くなった場合の，各種パラメータを示している．正解率は 0.741 となった．その際のユーザ特徴ベクトルの作成方法は，ウェブホスト列の前処理手法に**集合化**を用い，重み付け手法に **w2vave** を用いたものであった．Word2Vec のモデル学習における設定は，次元数が 128，ウインドウサイズが 5 であった．分類器はロジスティック回帰分類で最適パラメータはグリッドサーチによってパラメータ C は 1 と決められた．表 5 は，適合率，再現率，F 値を示している．

子供の有無の推定の結果

表 6 は，子供の有無の推定において最も正解率が高くなった場合の，各種パラメータを示している．正解率は 0.694 となった．その際のユーザ特徴ベクトルの作成方法は，ウェブホスト列の前処理手法に**集合化**を用い，重み付け手法に **w2vave** を用いたものであった．Word2Vec のモデル学習における設定は，次元数が 128，ウインドウサイズが 3 であった．分類器はロジスティック回帰分類で最適パラメータはグリッドサーチによってパラメータ C は 1 と決められた．表 7 は，適合率，再現率，F 値を示している．

職業カテゴリの推定の結果

表 8 は職業カテゴリの推定において最も正解率が高くなった場合の，各種パラメータを示している．正解率は 0.305 となった．その際のユーザ特徴ベクトルの作成方法は，ウェブホスト列の前処理手法に**集合化**を用い，重み付け手法に **w2vave** を用いたものであった．Word2Vec のモデル学習における設定は，次元数が 512，ウインドウサイズが 3 であった．分類器はロジスティック回帰分類で最適パラメータはグリッドサーチによってパラメータ C は 1 と決められた．図 4 は，結果の混同行列を示している．

表 2　性別の推定の結果

ウェブホスト列処理手法	重み付け手法		分類器		正解率
オリジナル	d2v	size=256, window=3	ロジスティック回帰	C=0.01	.774
オリジナル	d2v	size=128, window=3	SVM	C=1	.771
オリジナル	IUF		XGBoost	eta=0.1, max depth=4, subsample=1	.771
連続部の集約	d2v	size=256, window=3	ロジスティック回帰	C=0.01	.781
連続部の集約	d2v	size=256, window=3	SVM	C=1	.754
連続部の集約	IUF		XGBoost	eta=0.1, max depth=4, subsample=1	.764
集合化	w2vave	size=256, window=5	ロジスティック回帰	C=10	**.844**
集合化	d2v	size=128, window=5	SVM	C=1	.834
集合化	w2vave	size=512, window=10	XGBoost	eta=0.1, max depth=4, subsample=1	.796

表 3　年齢カテゴリ推定の最適結果

ウェブホスト列処理手法	集合化
重み付け手法	IUF
分類器	SVM
	C=10
正解率	**.315**

表 4　未既婚推定の最適結果

ウェブホスト列処理手法	集合化
重み付け手法	w2vave
	size=128
	window=5
分類器	ロジスティック回帰
	C=1
未既婚の割合	未婚 30%，既婚 70%
正解率	**.741**

表 5　未既婚推定の最適結果の適合率，再現率，F 値

	適合率	再現率	F 値
未婚	0.68	0.27	0.38
既婚	0.75	0.95	0.84

表 6　子供の有無推定の最適結果

ウェブホスト列処理手法	集合化
重み付け手法	w2vave
	size=128
	window=3
分類器	ロジスティック回帰
	C=1
子供の有無の割合	子供あり 38%，子供なし 62%
正解率	**.694**

表 7　子供の有無推定の最適結果の適合率，再現率，F 値

	適合率	再現率	F 値
子供なし	0.68	0.39	0.49
子供あり	0.70	0.89	0.78

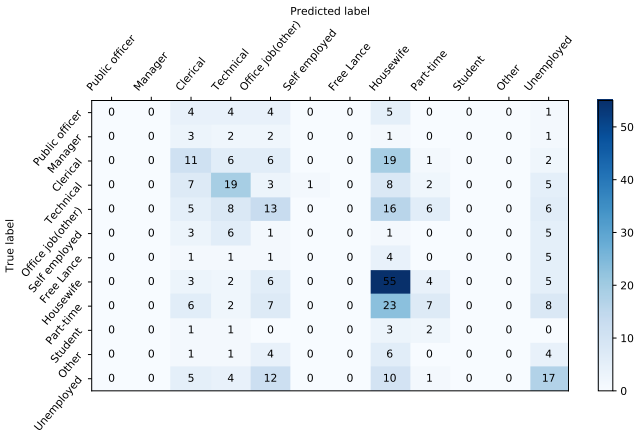

図 4　職業カテゴリ推定の最適結果の混同行列

表 8　職業カテゴリ推定の最適結果

ウェブホスト列処理手法	集合化
重み付け手法	w2vave
	size=512
	window=3
分類器	ロジスティック回帰
	C=1
正解率	**.305**

5.3　考　　察

性別の推定は，正解率 0.844 で行うことが可能なモデルを構築することができた．この正解率は非常に高いわけではないが，ランダムで性別を推定するよりも十分に高い正解率で性別を推定することができているといえる．よって，ウェブ広告閲覧履歴は性別の推定に貢献するということができる．

年齢の推定では，図 3 の年齢カテゴリ予測の混同行列を見ると，ある程度対角線上に大きな値が現れていることがわかる．年齢カテゴリ予測の正解率は 0.315 であったが，おおむねの年齢の推定という意味では，ある程度の予測が可能であると言える．

未既婚の推定は，正解率が 0.741 であった．また，子供の有無の推定は，正解率が 0.694 であった．これらの正解率は，性別の推定よりも正解率が低くなってしまっていた．この原因の一つは，性別の推定においてはラベルごとのデータの偏りがあまりなかったのに対して，未既婚の推定と子供の有無の推定では，ラベルごとのデータの偏りがあったことがあげられる．表 4 に記載したとおり，今回のデータセットにおいて既婚の割合は 70%と多く，既婚かどうかの推定では F 値が 0.84 とより良くなっている．子供の有無の推定では，表 6 に示すとおり，子供なしの割合が 62%と多かったが，こちらについては逆に，より少ないラベルである子供ありの推定で F 値が 0.78 とより

良くなっている．クラスラベルの偏りを考慮した学習を行うなど，今後，より精緻な実験を行いたい．

これらのユーザ属性の推定において，様々なユーザ特徴ベクトルの作成方法を試した．その中で，安定して良い結果をもたらしたのは，ウェブホスト列の前処理手法として**集合化**を用いて，重み付け手法として **w2vave** を用いるという手法であった．また，分類器は**ロジスティック回帰**を用いた場合に高い正解率となった．ウェブホスト列の前処理手法としては**集合化**が良かったことから，ユーザ属性を推定するためには，ユーザがあるウェブホストを閲覧したかどうかという程度まで情報を抽象化してしまうのが良いと考えられる．また，重み付け手法としては **w2vave** が良かったことから，類似するウェブホストはある程度集約して扱うことがユーザ属性の推定に必要であるということが考えられる．いずれも，ウェブ広告閲覧履歴をある程度抽象化してユーザ特徴ベクトルを作成することが良いということが示されたといえる．

分類器では**ロジスティック回帰**が良かったが，このような結果になった原因は，学習データが十分ではなかったということが考えられる．たとえば，XGBoost は様々な分類問題において良い正解率を達成することが知られているが，今回の実験ではそのような結果にならなかった．これは，XGBoost に用いる訓練データの量が不十分であり，過学習が起こったということが考えられる．訓練データの量が少ないため，学習率を上げたところ，より悪い正解率となってしまい，最終的には学習率を下げざるを得なかった．このようなことからも，訓練データの量が不十分であったと考えられる．

6　まとめと今後の課題

本研究では，ウェブ広告閲覧履歴を用いてユーザの属性を推定する問題に取り組んだ．まず，ウェブ広告閲覧履歴から URL を取得し，それをウェブホストに変換する処理を行う．そして得られたウェブホスト列に対して，前処理を行う手法として，**オリジナル**，**連続部の集約**，**集合化**という 3 種類の手法を提案した．前処理を行ったウェブホスト列から，**Count**，**IUF**，**w2vave**，**d2v** という 4 種類の重み付け手法を提案し，ユーザ特徴ベクトルの生成を行った．

ウェブホスト列の前処理手法と，重み付け手法の組み合わせによって，様々なユーザ特徴ベクトルが生成される．それらを用いてユーザ属性の推定を行った．その結果，ウェブホスト列の前処理手法としては**集合化**を用いること，重み付け手法としては **w2vave** を用いる手法において，推定の精度が良いということが示された．

今後の課題としては，まず，データ量を増やすということが考えられる．本研究では，1,991 名のユーザを対象にして実験を行った．しかし，利用する分類器によってはデータ量が不十分であり，過学習が起こったと考えられる．その問題を回避するためには，データ量を増やすことが必要だと考えられる．他には，今回対象としたユーザ属性以外にも様々なユーザ属性が存在するため，より多くのユーザ属性を対象として研究を進めていきたいと考えている．

謝　　辞

本研究の一部は JSPS 科学研究費助成事業 JP16H02906，JP17H00762，JP18H03244，JP18H03243 による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] 星尚志, 秋山卓也, 木村塁, 黒柳茂, 南川敦宣. URL エンベディングを用いたライフイベント予測. 情報処理学会 研究報告データベースシステム, Vol. 167, No. 3, pp. 1–5, 2018.

[2] 工藤航, 鳥海不二夫. 新聞記事のアクセスログを用いたユーザ属性の逐次推定. 人工知能学会全国大会論文集 第 32 回全国大会, Vol. JSAI2018, pp. 1–4, 2018.

[3] Yukihiro Tagami, Hayato Kobayashi, Shingo Ono, and Akira Tajima. Representation learning for users' web browsing sequences. *IEICE Transactions on Information and Systems*, Vol. E101.D, No. 7, pp. 1870–1879, 2018.

[4] Rajaraman Kanagasabai, Anitha Veeramani, Hu Shangfeng, Kajanan Sangaralingam, and Giuseppe Manai. Classification of massive mobile web log urls for customer profiling analytics. In *Proceedings of 2016 IEEE International Conference on Big Data*, pp. 1609–1614, 2016.

[5] 伊藤淳, 西田京介, 星出高秀, 戸田浩之, 内山匡. Twitter と blog の共通ユーザプロフィールを利用した twitter ユーザ属性推定. 情報処理学会 研究報告情報基礎とアクセス技術, Vol. 2013, No. 4, pp. 1–8, 2013.

[6] 近藤聖也, 吉田孝志, 和泉潔, 山田健太. 位置情報付きツイートを用いたユーザ属性推定と地域クラスタリング. 人工知能学会全国大会論文集 第 30 回全国大会, Vol. JSAI2016, pp. 1–3, 2016.

[7] Zachary Miller, Brian Dickinson, and Wei Hu. Gender prediction on twitter using stream algorithms with n-gram character features. *International Journal of Intelligence Science*, Vol. 2, No. 4, pp. 143–148, 2012.

[8] John D Burger, John Henderson, George Kim, and Guido Zarrella. Discriminating gender on twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 1301–1309, 2011.

[9] 榊剛史, 松尾豊. ソーシャルメディアユーザの職業推定手法の提案. 日本知能情報ファジィ学会誌, Vol. 26, No. 4, pp. 773–780, 2014.

[10] Tomas Mikolov, Kai Chen, Greg Corrado, , and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proceedings of the 1st International Conference on Learning Representations*, pp. 1301–3781, 2013.

[11] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, pp. 1188–1196, 2014.