Cross-Domain User Similarity without Overlapping Attributes via Optimal Transport Theory

Genki Kusano[1], Masafumi Oyamada[1]

[1]NEC Corporation, Japan

Abstract

Discovering similar users from different perspectives plays an essential role in marketing activities regarding understanding customers. In particular, user similarity based on attributes (e.g., preferences and behavioral tendencies) effectively captures user needs. However, such attributes are defined differently depending on the dataset, making comparing users accurately in a cross-domain setting challenging. Previous methods have focused on unifying attributes to calculate user similarity across multiple datasets. However, applicability is limited because they assume the existence of users or attributes that overlap in two datasets. In this paper, we propose *Attribute TransPortation* (ATP), a novel method based on optimal transport theory for calculating cross-domain user similarity without imposing the assumption of overlapping. In numerical experiments on six real-world datasets, ATP performed quantitatively better than related methods in two tasks, one for finding the same user to evaluate the similarity itself and the other for a cross-domain recommendation task to evaluate the effectiveness of utilizing the similarity.

Keywords

user similarity, optimal transport theory, cross-domain recommendation

1. Introduction

One-to-one marketing, which analyzes a single customer in detail, is becoming a powerful strategy for delivering better products and services. Then, it is essential to understand what customers are interested in outside the company in addition to the information obtained from the internal data that marketers own. For this reason, user data is increasingly being analyzed from the outside. However, it is generally difficult to identify target users in external data[1].

When analyzing users across internal and external data, common attributes are required to link users. A typical example of such attributes is a *demographic* attribute (e.g., age, gender, or occupation); however, analyzing users only with demographic attributes can cause stereotyping problems, leading to inappropriate marketing strategies. In this study, we focus on discovering similar users based on *psychographic* attributes, such as a user's preferences and behavioral tendencies. While psychographic attributes are more effective than demographic attributes in directly capturing user needs and avoiding stereotyping, internal and external data rarely contain the same attributes. Therefore, discovering similar users across different datasets is

eCom'23: ACM SIGIR Workshop on eCommerce, July 27, 2023, Taipei, Taiwan

g-kusano@nec.com (G. Kusano); oyamada@nec.com (M. Oyamada)

[1] External data is typically obtained by purchasing it from data vendors or forming a partnership between companies to share their user data, but personal information is often anonymized, and it is unknown whether external data contain target users.

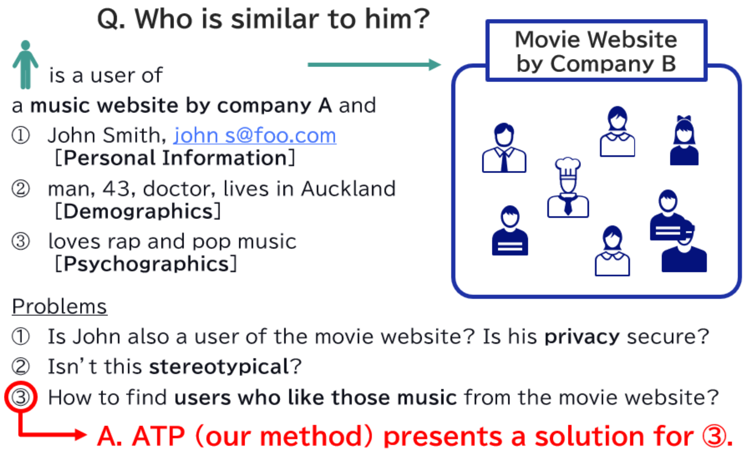

Q. Who is similar to him?
is a user of
a music website by company A and
① John Smith, john s@foo.com
[Personal Information]
② man, 43, doctor, lives in Auckland
[Demographics]
③ loves rap and pop music
[Psychographics]
Movie Website
by Company B
Problems
① Is John also a user of the movie website? Is his privacy secure?
② Isn't this stereotypical?
③ How to find users who like those music from the movie website?
A. ATP (our method) presents a solution for ③.

Figure 1: Problem overview. To discover a similar user across different user datasets, we can use (a) personal information, (b) demographic attributes, and (c) psychographic attributes, which typically leads to problems with (a) privacy risks, (b) stereotyping, and (c) difficulties in handling un-unified psychographic attributes. Our method ATP presents a solution for (c).

challenging because such psychographic attributes differ for both types of data (Fig. 1).

There have been various research in the context of cross-domain recommendation [1] which extract user interests from multiple datasets as user attributes and then utilize them for recommendation tasks. The methods in this vein [2, 3, 4] define cross-domain user similarity based on attributes; however, they assume the existence of users or attributes that overlap in two datasets, which makes them too restrictive for application to real-world situations[2].

To resolve the overlapping problem, this paper proposes *Attribute TransPortation* (ATP), a method for calculating cross-domain user similarity by connecting attributes. The core idea of ATP is to connect attributes across different domains by adopting *optimal transport* (OT) theory [5], a mathematical theory initially designed for solving the problem of transferring one probabilistic distribution to another while minimizing the total moving cost and preserving the total amount of each distribution. Regarding two sets of attributes of different domains as the two distributions with which OT deals, ATP takes a target user who is described with attributes in a target domain and re-describes him/her with attributes in the other domain. Since applying OT to two sets of attributes is interpreted as distributing one attribute in one domain to one attribute in another domain, it is possible to re-describe a user even if the two domains have no overlapping attributes. In this way, ATP enables the calculation of cross-domain user similarity by measuring the similarity between a user of the other domain and the re-described target user in the same feature space.

We investigated the effectiveness of ATP by conducting two experiments on six situations with real-world datasets containing millions of behavioral histories. In the first experiment, we evaluated ATP's effectiveness in matching accuracy for the same user. If the same users exist across domains, the cross-domain user similarity should be high; thus, we evaluated the matching

[2] Even if we obtain external data (e.g., movie data crawled from review sites), there is no guarantee that users and attributes in the external data will also appear in the internal data (e.g., music data that we already own).

Table 1
List of main symbols.

Notation	Description
$\mathbb{R}$	Set of real numbers
$\mathbf{X} \in \mathbb{R}^{m \times n}$	Matrix
$\mathbf{x} \in \mathbb{R}^{n}$	Vector
$\|\mathbf{x}\|_p := (\sum_j \|\mathbf{x}_j\|^p)^{1/p}$	L^p-norm
$z \in \{s, t\}$	Domain
U^z	Set of users
A^z	Set of attributes
$u^z \in U^z, a^z \in A^z$	user and attribute
$\phi^z : U^z \to \mathbb{R}^{\|A^z\|}$	User feature map
$\phi^{s \to t} : U^s \to \mathbb{R}^{\|A^t\|}$	Cross-domain user feature map
$\mathrm{sim} : U^s \times U^t \to \mathbb{R}$	Cross-domain user similarity

accuracy of ATP for the same users. Comparing ATP with five other methods (including modifications of ATP) showed that it achieved the highest and second-highest accuracies in three and two situations, respectively. In the second experiment, we evaluated the effect of using ATP in the context of a cross-domain recommendation task [1]. A recommendation task involving users who do not behave at all (known as cold-start users) would be challenging to recommend items to them properly because their preferences are unknown. However, if their behaviors in another domain are available, existing methods [6, 3] can make better recommendations by utilizing cross-domain user similarity. We confirmed that ATP achieved the highest and second-highest improvement rates in five and one situations, respectively, where the highest improvement rate was over 13%.

Our contributions in this paper are summarized as follows.

- We propose *Attribute TransPortation* (ATP), a method for calculating cross-domain user similarity based on attributes.
- ATP is versatile because it requires no overlapping users or attributes.
- We conducted extensive experiments on six real-world situations and confirmed that ATP performed quantitatively better than related methods.

2. Proposed Method

Throughout this paper, we denote two domains as s and t (source and target), and z denotes either unless otherwise stated. The sets of users and attributes in domain z are denoted as U^z and A^z, respectively. All attributes appearing in this paper are set to be written in a natural language, e.g., $A^z = \{\text{dance, anime, ...}\}$. A user u^z is represented by a numerical vector $\phi^z(u^z) \in \mathbb{R}^{|A^z|}$, where a value $\phi^z(u^z)[a^z] \in \mathbb{R}$ means the degree of the user's interest in the attribute $a^z \in A^z$. We assume all user' features are given in this form throughout this paper (see Section 3 for how this feature is obtained). Table 1 summarizes the symbols frequently used in this paper.

Our objective is to find users of domain s who are similar to a target user of domain t based on attributes. Now that two users u^s and u^t of different domains are expressed as $\phi^s(u^s) \in \mathbb{R}^{|A^s|}$

and $\phi^t(u^t) \in \mathbb{R}^{|A^t|}$, respectively, the features belong to different feature spaces. Our approach consists of transforming $\phi^s(u^s)$ to a new feature $\phi^{s\to t}(u^s) \in \mathbb{R}^{|A^t|}$ in order to treat users of different domains in the same feature space. The next subsection explains how to calculate the transformed user feature $\phi^{s\to t}(u^s)$.

2.1. Observation of Semantic Similarity

When some matrix $\mathbf{W} \in \mathbb{R}^{|A^s|\times|A^t|}$ is given, a transformed feature vector is implemented as $\phi^{s\to t}(u^s) = \mathbf{W}^T\phi^s(u^s)$. Then, each element $\mathbf{W}[a^s, a^t]$ is interpreted as a contributing score in which an attribute a^s is reworded as another attribute a^t. If a^s and a^t are "similar" words in some sense, a user u^s who is interested in attribute a^s is likely to be interested in a^t as well; thus, $\mathbf{W}[a^s, a^t]$ is desired to take a higher value. To implement such $\mathbf{W}$, we use a semantic similarity which is calculated by adopting a word embedding model (e.g., fastText [7]) to attributes. Let $\mathbf{e}(a^z) \in \mathbb{R}^d$ be an embedding vector of an attribute a^z and $\cos(\mathbf{x}, \mathbf{y}) := \langle \mathbf{x}, \mathbf{y}\rangle/(\|\mathbf{x}\|\|\mathbf{y}\|)$ $(\mathbf{x}, \mathbf{y} \in \mathbb{R}^d)$ be cosine similarity for vectors. A candidate for such $\mathbf{W}$ is calculated by cosine similarity as $\mathbf{W}_{\cos}[a^s, a^t] := \cos(\mathbf{e}(a^s), \mathbf{e}(a^t))$.

The disadvantage of using $\mathbf{W}_{\cos}$ is that the relationship between two attributes does not reflect their domains. For example, the semantic similarity of "rap" and "musical" might be high because both terms appear in many music-related sentences. However, if "rap" and "musical" are attributes in the music and movie domains, respectively, they would represent different genres. This gap stems from the difference between determining one word from among the millions of words appearing in training data for the language model or from words appearing only in the two domains.

To overcome this disadvantage, we reflect on the relationship between two domains and propose our method on optimal transport (OT) theory [5] concerning the similarity of attributes considering domains. In the following subsection, we first briefly explain OT and then explain why it is suitable for reflecting the relationship between two domains as a transforming matrix.

2.2. Optimal Transport Theory

Let $P^s = \{(\mathbf{p}_i^s, \mu_i^s)\}_{i=1}^m$ and $P^t = \{(\mathbf{p}_j^t, \mu_j^t)\}_{j=1}^n$ be weighted point sets on $\mathbb{R}^d$. In OT, a matrix $\mathbf{W} \in [0,1]^{|P^s|\times|P^t|}$ is called a transportation plan when it determines the amount of the weight of a point in P^s to another point in P^t, and a set of transportation plans is denoted by

$$\Pi(P^s, P^t) := \left\{\mathbf{W} \geq 0 \;\middle|\; \textstyle\sum_j \mathbf{W}_{i,j} = \mu_i^s,\; \sum_i \mathbf{W}_{i,j} = \mu_j^t\right\}. \tag{1}$$

For example, when a point $\mathbf{p}_i^s$ has a weight μ_i^s, a transportation plan transfers the amount $\mathbf{W}_{i,j}$ of the weight of $\mathbf{p}_i^s$ to $\mathbf{p}_j^t$ (Fig. 2 and 3). When the cost of moving one point from $\mathbf{p}_i^s$ to $\mathbf{p}_j^t$ is $\mathbf{C}_{i,j} > 0$, the optimal transportation plan is defined as one minimizing the total moving cost $\sum_{i,j}\mathbf{W}_{i,j}\mathbf{C}_{i,j}$.

Among the many available OT methods, we used word rotator's distance (WRD) [8], OT specialized for natural language processing. For an attribute set A^z, WRD transforms it into a weighted point set $P(A^z) := \{(\mathbf{e}(a_k^z), \mu_k^z)\}_{a_k^z \in A^z}$ whose weight is calculated in accordance with the L^2-norm, as $\mu_k^z := \|\mathbf{e}(a_k^z)\|/\sum_{a\in A^z}\|\mathbf{e}(a^z)\|$. The optimal transportation plan by WRD is defined

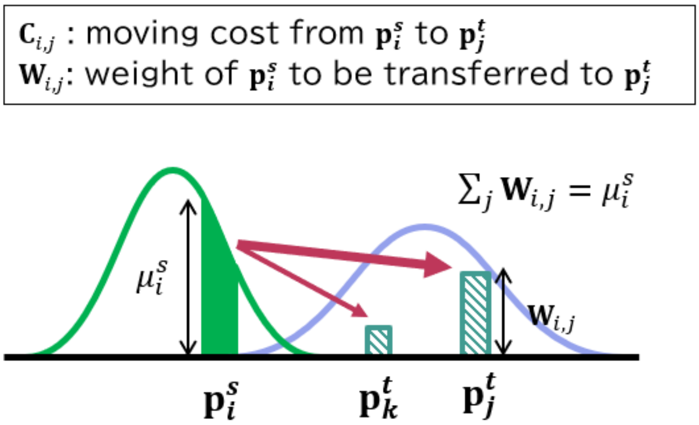

$\mathbf{C}_{i,j}$: moving cost from $\mathbf{p}_i^s$ to $\mathbf{p}_j^t$
$\mathbf{W}_{i,j}$: weight of $\mathbf{p}_i^s$ to be transferred to $\mathbf{p}_j^t$
$\sum_j \mathbf{W}_{i,j} = \mu_i^s$
μ_i^s
$\mathbf{W}_{i,j}$
$\mathbf{p}_i^s$
$\mathbf{p}_k^t$
$\mathbf{p}_j^t$

Figure 2: This figure shows that OT for 1-dimensional distributions transports a part of the green distribution into the blue distribution.

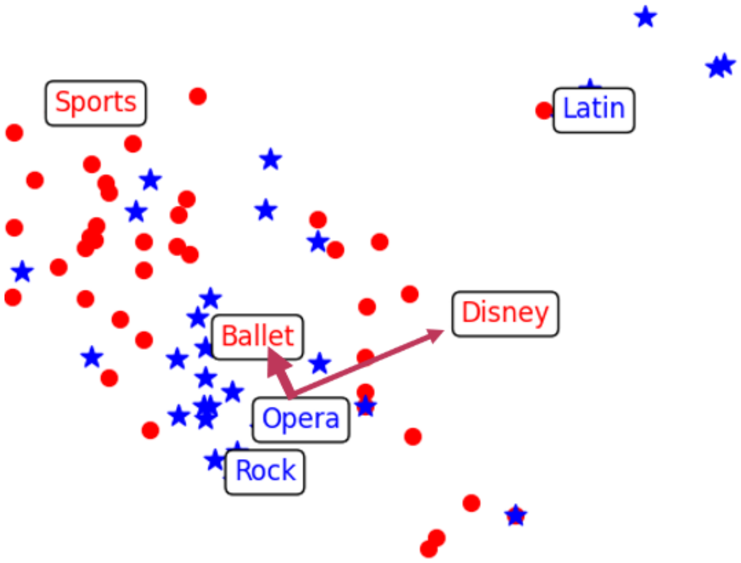


Figure 3: This figure indicates the 2-dimensional PCA plot of $\{\mathbf{e}(a)\}$ of music (blue dots) and movie (red stars) attributes. When we consider distributing the "opera" attribute in the music domain to movie attributes, OT proposes proportions to distribute the attributes.

as

$$\mathbf{W}_{\mathrm{OT}} := \underset{\mathbf{W}\in\Pi(P(A^s),P(A^t))}{\arg\min} \sum_{i,j} \mathbf{W}_{i,j}\mathbf{C}_{i,j} + \lambda\Omega(\mathbf{W}), \tag{2}$$

where $\mathbf{C}_{i,j} := 1 - \cos(\mathbf{e}(a_i^s), \mathbf{e}(a_j^t))$ is a cost defined by cosine similarity, $\lambda > 0$, and $\Omega(\mathbf{W}) :=$

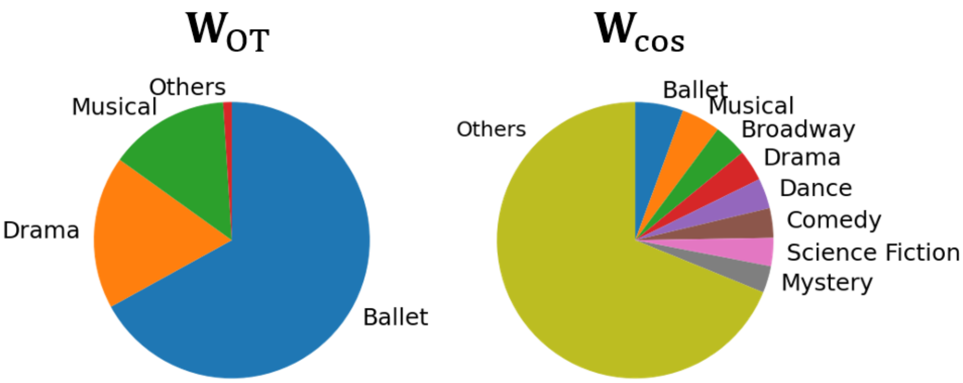


Figure 4: Visualization of the pie charts of $\mathbf{W}_*[\text{Opera}, a^{\text{movie}}]$ for $* \in \{\text{OT}, \cos\}$ and each attribute $a^{\text{movie}} \in A^{\text{movie}}$. The "Opera" attribute is similar in cosine similarity to various movie attributes (right figure). On the other hand, some movie attributes (e.g., "Broadway" and "Dance") are more similar to other music attributes than the "Opera" attribute. As the left figure shows, the transportation plan modifies the similarity relevant to the two domains.

$\sum_{i,j} \mathbf{W}_{i,j} \log(\mathbf{W}_{i,j})$ [3]. To solve it numerically, we added the regularization term $\Omega(\mathbf{W})$ from the Sinkhorn algorithm [10] [4].

OT reflects the relationship of two domains to the semantic similarity as follows. Let a^t_j be the attribute most similar to a^s_i, i.e., the moving cost from a^s_i to a^t_j is the minimum compared to the other attributes in domain t. In this case, a transportation plan will typically transfer most of the weights of a^s_i to a^t_j, but if there is another attribute $a^s_{i'}$ that is most similar to a^t_j, it is also going to transfer most of the weights of a^t_j to $a^s_{i'}$ (Figure 3). OT balances this transferring of weights by satisfying the summation assumption of $\Pi(P(A^s), P(A^t))$ and then calculates an attribute similarity for items that are not only semantic but also relevant to the two domains.

2.3. Cross-Domain User Similarity

Although the resulting matrix $\mathbf{W}_{\text{OT}}$ re-describes a user feature $\phi^s(u^s)$ into that in domain t as $\phi^{s\to t}(u^s) = (\mathbf{W}_{\text{OT}})^T \phi^s(u^s)$, there remains a problem regarding the treatment of unrelated attributes. For example, we assume that a user u^s is intensely interested in the "instrumental" attribute in the music domain, and there is no similar movie attribute to the music attribute. Although the music attribute is unrelated to any movie attributes, the construction $(\mathbf{W}_{\text{OT}})^T \phi^s(u^s)$ cannot avoid fully transferring the degree of this user's preference for the unrelated music attribute "instrumental" to the transformed feature.

To deal with this issue, we reduce the influence of the unrelated attribute by formulating

[3] As the original paper [8] discussed, WRD has significance compared to word mover's distance [9], which is a traditional method of introducing OT into word sets, by changing the cost matrix from the Euclidean distance $\mathbf{C}_{i,j} = \|\mathbf{e}(a^s_i) - \mathbf{e}(a^t_j)\|_2$ to the distance matrix from the cosine similarity and the mass on each word from a uniform distribution $\mu^z_k \equiv 1/|A^z|$ to a weighted distribution of which the mass is proportional to the norm of the embedding vector.

[4] In experiments, we used Python Optimal Transport [11] and set $\lambda = 0.01$.

its degrees of influence for an attribute a^s to the attribute set A^t as *attribute-domain similarity*, which is defined by

$$\mathbf{s}_{A^t}[a^s] := \frac{1}{2}\left(1 + \max_{a^t \in A^t} \cos(\mathbf{e}(a^s), \mathbf{e}(a^t))\right). \tag{3}$$

We can then define a cross-domain user feature of u^s as

$$\phi^{s\to t}(u^s) := (\mathbf{W}_{\mathrm{OT}})^T(\mathbf{s}_{A^t} \odot \phi^s(u^s)) \in \mathbb{R}^{|A^t|}, \tag{4}$$

where $\odot$ is the Hadamard (element-wise) product for vectors.

As a mathematical remark, users who are similar in terms of user features in the original domain are also similar in terms of the transformed user features; in other words, the mapping from $\phi^s(u^s)$ to $\phi^{s\to t}(u^s)$ is Lipschitz continuous. To be precise, for any $1 \leq p < \infty$, there is a constant $C_p > 0$ such that

$$\|\phi^{s\to t}(u_1^s) - \phi^{s\to t}(u_2^s)\|_p \leq C_p\|\phi^s(u_1^s) - \phi^s(u_2^s)\|_p \tag{5}$$

for any users $u_1^s, u_2^s \in U^s$. This is proven as a consequence of the facts that (1) the operation of the matrix product is a linear map, i.e., for any matrix $\mathbf{M}$, there exists a constant $C > 0$ such that $\|\mathbf{Mx} - \mathbf{My}\|_2 \leq C\|\mathbf{x} - \mathbf{y}\|_2$ for any vectors $\mathbf{x}, \mathbf{y}$, (2) the Hadamard product can be seen as a diagonal matrix with $\mathbf{c} \odot \mathbf{x} = \mathrm{diag}(\mathbf{c})\mathbf{x}$, and (3) the 2-norm and the p-norm ($1 \leq p < \infty$) are equivalent norms, i.e., there exist constants $C_1, C_2 > 0$ such that $C_1\|\mathbf{x}\|_p \leq \|x\|_2 \leq C_2\|\mathbf{x}\|_p$ for any vectors $\mathbf{x}$.

Finally, we define the cross-domain user similarity in the same feature space $\mathbb{R}^{|A^t|}$ by comparing $\phi^{s\to t}(u^s)$ and $\phi^t(u^t)$. Note that $\phi^t(u^t)[a_j^t]$ can have a higher value for some attribute a_j^t, while none of the transformed user features can have higher values for the attribute because $\mathbf{s}_{A^s}[a_j^t]$ is low. We also adjust the user feature $\phi^t(u^t)$ on the basis of attribute-domain similarity as

$$\mathrm{sim}_t(u^s, u^t) := \cos\left(\phi^{s\to t}(u^s), \mathbf{s}_{A^s} \odot \phi^t(u^t)\right). \tag{6}$$

In addition, since u^s and u^t can also be compared in A^s through $\phi^{t\to s}(u^t)$ and $\phi^s(u^s)$, we define the *cross-domain user similarity of ATP*, as

$$\mathrm{sim}_{\mathrm{ATP}}(u^s, u^t) := \frac{1}{2}\left(\mathrm{sim}_t(u^s, u^t) + \mathrm{sim}_s(u^t, u^s)\right). \tag{7}$$

3. Experiments

We applied ATP to real-world datasets and compared its performance to other methods in defining cross-domain user similarities for linkage of the same users and cross-domain recommendation tasks.

3.1. Dataset

We used the Amazon Review Dataset [12][5] for our experiments, which contain millions of user evaluations on items of various domains along with attribute information. Regarding

[5] Data source: https://nijianmo.github.io/amazon/index.html. We regarded the Music (CDs_and_Vinyl), Movie (Movies_and_TV), Book (Books), Kindle (Kindle_Store), and Clothes (Clothing_Shoes_and_Jewelry) datasets as different domains.

Table 2
The number of all reviews, users, items, attributes, and sparsity of each domain.

	$\lvert D\rvert$	$\lvert U\rvert$	$\lvert I\rvert$	$\lvert A\rvert$	$\lvert D\rvert/(\lvert U\rvert\lvert I\rvert)$
Movie	581987	5172	52108	381	0.00216
Music	341089	2805	65083	216	0.00187
Kindle	751749	6830	88659	76	0.00124
Clothes	327417	4631	138971	917	0.00051
Book	7883742	62404	610031	472	0.00021

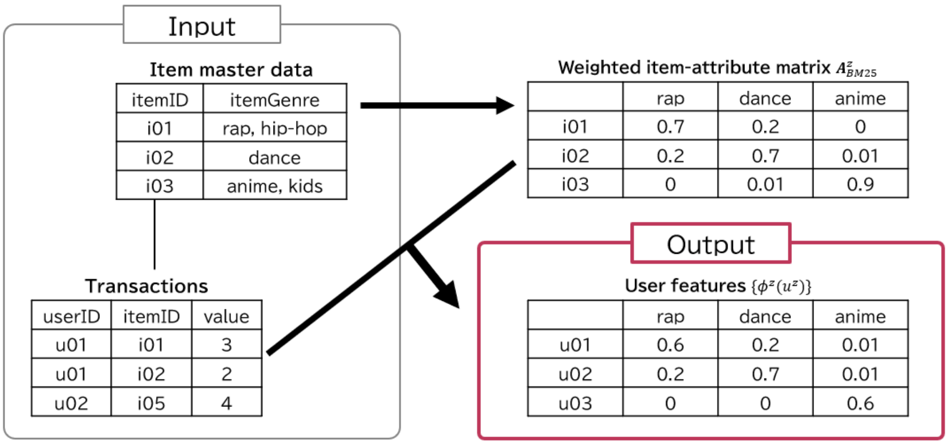
Table 3
The number of intersections of pairwise datasets, where each value inside the parentheses is the Jaccard similarity.

	s	t	$\lvert U^s \cap U^t\rvert$		$\lvert A^s \cap A^t\rvert$	
Case 1	Music	Movie	516	(0.069)	38	(0.068)
Case 2	Book	Music	495	(0.008)	50	(0.078)
Case 3	Clothes	Movie	106	(0.011)	48	(0.038)
Case 4	Clothes	Music	19	(0.003)	34	(0.031)
Case 5	Book	Movie	1318	(0.02)	61	(0.077)
Case 6	Kindle	Book	6313	(0.1)	59	(0.121)

users, we limited the user set U^z to those who had evaluated more than 50 items, as it would be challenging to capture meaningful user features for users who gave evaluations on only a few items. Regarding attributes, we separated all original attributes into words by morphological analysis, cleaned them to their lemmata (canonical forms) because some shared common components (e.g., “rock” in “classic rock”, “folk rock”, and “hard rock”)[6], and set the separated and cleaned words as the attribute set A^z. We also removed attributes that were not given to 5 or more items from A^z.

Let I^z and $D^z \subset U^z \times I^z$ be sets of items and transaction logs that users reviewed items, respectively. Tables 2 and 3 summarize the statistics for the datasets and intersections of the pairwise datasets, respectively. We set the cross-domain situation by choosing pairwise datasets so that common users appeared in both datasets, denoted as $U_{\text{linked}} := U^s \cap U^t \neq \emptyset$. For Table 3, we omitted the item column because all pairs of item sets had no intersections, i.e., $|I^s \cap I^t| = 0$. As indicated by the lower number of intersections and the Jaccard similarity[7], users and attributes seldom overlapped across domains, even though each dataset had a large number of users and attributes. This observation supports the necessity of this study, which aims to obtain new insights from two datasets with slight overlapping.

[6] We used spaCy (https://spacy.io/) to obtain the canonical forms of words.
[7] Jaccard similarity of two sets A and B is defined as $|A \cap B|/|A \cup B|$.



Item master data

itemID	itemGenre
i01	rap, hip-hop
i02	dance
i03	anime, kids

Transactions

userID	itemID	value
u01	i01	3
u01	i02	2
u02	i05	4

Weighted item-attribute matrix A^z_{BM25}

	rap	dance	anime
i01	0.7	0.2	0
i02	0.2	0.7	0.01
i03	0	0.01	0.9

User features $\{\phi^z(u^z)\}$

	rap	dance	anime
u01	0.6	0.2	0.01
u02	0.2	0.7	0.01
u03	0	0	0.6

Figure 5: The figure indicates the way to calculate user features from transaction data.

3.2. User Feature

We created a user feature $\phi^z(u^z)$ from user transactions. Let I^z be a set of items in domain z, $I^z[u^z] \subset I^z$ be a set of items that user u^z evaluated, and $r(u^z, i^z)$ be a rating score for i^z by u^z. Let $\mathbf{A}^z$ be an item-attribute matrix of A^z, where each element is given by $\mathbf{A}^z[i^z, a^z] = 1$ if an item attribute $a^z \in A^z$ is attached to an item $i^z \in I^z$; otherwise, 0. A simple approach to obtaining a user feature is just to take a weighted sum of the attribute vectors in which the user was interested, that is,

$$\phi^z_{\text{simple}}(u^z) := \sum_{i^z \in I^z[u^z]} r(u^z, i^z)\mathbf{A}^z[i^z] \in \mathbb{R}^{|A^z|}. \tag{8}$$

To properly adjust the degree of an attribute following its domain, we change the item-attribute matrix $\mathbf{A}^z$ by its transformed matrix using BM25 [13], which is denoted by $\mathbf{A}^z_{\text{BM25}}$[8]. To fairly compare all users in one domain, we also normalize a user feature so that the sum of all elements of any user features is equal to 1. To sum up, the resulting user feature of u^z (Figure 5) is obtained as

$$\phi^z(u^z) := \frac{1}{Z} \sum_{i^z \in I^z[u^z]} r(u^z, i^z)\mathbf{A}^z_{\text{BM25}}[i^z] \tag{9}$$

where Z is a normalization term to make $\|\phi^z(u^z)\|_1 = 1$. Then, each element in a user feature can be interpreted as a percentage of the degree of the user's interest.

3.3. Baselines

We compared our method with the following two baselines **Agg** and **SCT** for calculating cross-domain user similarity. As an ablation study, we also compared **Agg** and **SCT** with **ATP-cos**

[8] For example, assume that a song has both "rap" and "young" attributes. If we describe this song with one attribute, "rap" would be appropriate because it is more representative than "young" in the music domain. BM25 is known in the information retrieval field as an appropriate method for adjusting the importance of each attribute in this way.

and **ATP-ads**, which are partially changed components of **ATP**.

ATP (Proposed approach) : The embedding model $\mathbf{e}$ to compute WRD was set as fastText [7] because all attributes are words (not sentences). It can handle words that are not registered to its corpus[9].

ATP-cos : To determine the effectiveness of OT, we replaced the optimal transportation plan $\mathbf{W}_{\mathrm{OT}}$ in Eq. (4) with $\mathbf{W}_{\cos}$.

ATP-ads : To determine the effectiveness of attribute-domain similarity, we cancelled it, that is, we set $\mathbf{s}_{A^t}[a^s] \equiv 1$ and $\mathbf{s}_{A^s}[a^t] \equiv 1$ for all attributes.

Agg [6] : This method expresses a user feature by setting the aggregated attribute set $A^s \cup A^t$ and calculating BM25 for a user-attribute matrix on $A^s \cup A^t$. With our symbols, a user feature with **Agg** can be written as

$$\phi_{\mathrm{Agg}}(u^z) := \frac{1}{Z} \sum_{i^z \in I^z[u^z]} r(u^z, i^z) \mathbf{A}^{\mathrm{outer}}_{\mathrm{BM25}}[i^z] \tag{10}$$

where Z is a normalization term so as to make $\|\phi_{\mathrm{Agg}}(u^z)\|_1 = 1$, $\mathbf{A}^{\mathrm{outer}}$ is the outer join of $\mathbf{A}^s$ and $\mathbf{A}^t$, and $\mathbf{A}^{\mathrm{outer}}_{\mathrm{BM25}}$ is the transformed matrix of $\mathbf{A}^{\mathrm{outer}}$ by BM25.

SCT-*K* [3] : For a set $\{\mathbf{e}(a) \mid a \in A^s \cup A^t\}$ of embedding vectors of all attributes, Semantic Correlation in Tagging (SCT) first runs the K-means clustering algorithm, and then expresses the user feature as

$$\phi_{\mathrm{SCT}}(u^z)[c_k] = \frac{N(c_k|u^z)}{\sum_{\ell=1}^{K} N(c_\ell|u^z)}, \tag{11}$$

where $N(c_k|u^z)$ is the number of attributes on which the user u^z evaluated that belong to a cluster c_k. In the experiments, we investigated $K = 50$ and 100.

On the basis of the above construction, $\phi_{\mathrm{Agg}}(u^z)$ and $\phi_{\mathrm{SCT}}(u^z)$ of either domain z belong to the same spaces $\mathbb{R}^{|A^s \cup A^t|}$ and $\mathbb{R}^K$, respectively. Here, we calculated the cross-domain user similarity by cosine similarity, as $\mathrm{sim}_*(u^s, u^t) := \cos(\phi_*(u^s), \phi_*(u^t))$, where $* \in \{\mathrm{Agg}, \mathrm{SCT}\}$.

3.4. Matching Users

Since all overlapping users were active on both domains, any $u \in U_{\mathrm{linked}} = U^s \cap U^t$ should show a higher cross-domain user similarity $\mathrm{sim}(u, u)$ than $\mathrm{sim}(u, u^t)$ of the other users $u^t \in U^t \setminus \{u\}$. We, therefore, evaluated the effectiveness of ATP by matching accuracy for the same user.

[9] We used the fastText model released in https://dl.fbaipublicfiles.com/fasttext/vectors-crawl/cc.en.300.bin.gz without any fine-tuning to ensure a fair comparison.

Table 4
Results of experiments for matching users in terms of MRR (left) and top@10% (right, in percentage), respectively, where bold and underlined values represent best and second-best methods in each column.

	Case 1		Case 2		Case 3		Case 4		Case 5		Case 6	
ATP	**0.014**	**25.8**	**0.005**	38.4	**0.006**	**22.6**	0.0010	**15.8**	0.002	33.6	0.016	39.5
Agg	0.007	24.4	0.001	15.6	0.003	7.6	0.0009	10.5	**0.004**	30.8	**0.026**	**46.1**
SCT-50	0.004	15.1	0.001	15.4	0.005	**22.6**	0.0007	0.0	0.001	33.1	0.012	36.2
SCT-100	0.004	17.6	0.002	**40.2**	0.002	6.6	0.0006	0.0	0.001	30.5	0.016	38.0
ATP-cos	0.012	25.2	0.002	24.2	0.002	8.5	0.0008	10.5	0.002	25.0	0.015	39.2
ATP-ads	0.013	21.9	0.003	40.0	0.004	20.8	**0.0014**	10.5	0.002	**34.8**	0.015	38.8

3.4.1. Evaluation

We utilized the mean reciprocal rank (MRR) and top-k accuracy to evaluate the matching accuracy. When we calculate $\{\mathrm{sim}(u, u^t) \mid u^t \in U^t\}$ for $u \in U_{\mathrm{linked}}$, if the similarity of u is located at the top ℓ-th position, we denote the number ℓ by $\mathrm{rank}(u)$. Then, the MRR is defined by the mean of the inverse of the rank, i.e.,

$$\mathrm{MRR} := \frac{1}{|U_{\mathrm{linked}}|} \sum_{u \in U_{\mathrm{linked}}} \frac{1}{\mathrm{rank}(u)}, \tag{12}$$

and top-k accuracy is defined by the probability that the rank is less than or equal to k, i.e.,

$$\mathrm{top@}k := \frac{|\{u \in U_{\mathrm{linked}} \mid \mathrm{rank}(u) \leq k\}|}{|U_{\mathrm{linked}}|}. \tag{13}$$

The higher MRR and top@k are, the better the cross-domain user similarity is.

Note that as the number of users to look for $|U^t|$ increases, MRR and top@k deteriorate. For example, it is more difficult to obtain a result for $\mathrm{top@}10 = 0.1$ when $|U^t| = 10^5$ than when $|U^t| = 10^3$. To reduce the effect from $|U^t|$, we determined whether the rank was in the top-10% of U^t and wrote top@$\lceil 0.1|U^t| \rceil$ as top@10%, where $\lceil \cdot \rceil$ is the ceiling function.

3.4.2. Results

The results shown in Table 4 indicate that **ATP** exhibited the highest MRR or top@10% in Cases 1–4 and the second-highest scores in Cases 5 and 6. One of the reasons **ATP** was defeated by **Agg** in these latter two is that there were many common attributes across domains (please refer to Table 3) and **Agg** utilized them to match users. We observed that the ratios of $\mathrm{sim}_{\mathrm{Agg}}(u, u) = 0$ for $u \in U_{\mathrm{linked}}$, which indicates that u did not show any interest in common attributes across domains, were less than 0.5% in Cases 5 and 6, and over 3%; otherwise. In other words, in Cases 1–4, where many overlapping users did not show interest in common attributes, **ATP** successfully found the same users with higher accuracy. **SCT**-K, which also essentially utilizes common attributes in clusters, showed the highest top@10% in Case 2 by $K = 50$; however, the results depended on the selections of K, and there remains the difficulty in determining the proper K. As an ablation study, we confirmed the necessity of OT compared to the results by **ATP-cos**. While **ATP-ads** outperformed **ATP** for Case 4 and 5, the differences were slight, and

ATP-ads was defeated in the other situations; hence, we also confirmed the necessity of using attribute-domain similarity.

3.5. Cross-Domain Recommendation

In this section, we evaluated the effect of using ATP in the context of a recommendation task that predicts which rating score a user will give to an item based on his past rating history. However, if a user did not evaluate at all (i.e., a *cold-start user*), it would be difficult to make a better recommendation because his preferences are unknown. If a cold-start user in domain t behaved in another domain s, methods of cross-domain recommendation [1] could then make an appropriate recommendation by utilizing information from the other domain. Existing methods [2, 3, 14] have been able to predict the rating scores of cold-start users by using cross-domain user similarity, and here, we investigated the effectiveness of ATP in comparison with these methods.

3.5.1. Evaluation

Let $\mathbf{R}^z \in \mathbb{R}^{|U^z|\times|I^z|}$ be a rating matrix in which $\mathbf{R}^z[u^z, i^z]$ is the rating score for an item i^z by a user u^z, where $\mathbf{R}^z[u^z, i^z] = 0$ if u^z did not rate i^z. We divided $\mathbf{R}^t$ into $\mathbf{R}^t_* \in \mathbb{R}^{|U^t_*|\times|I^t|}$ ($* \in \{\text{train}, \text{test}\}$) where $U^t_{\text{test}} := U_{\text{linked}}$ and $U^t_{\text{train}} := U^t \setminus U_{\text{linked}}$, and set the recommendation task to predicting the rating scores of U^t_{test}. Each result is evaluated by the mean absolute error (MAE). Specifically, a predicted rating score for $i^t \in I^t$ by $u \in U^t_{\text{test}}$ is denoted by $\hat{r}(u, i^t)$, and the MAE is then defined by

$$\text{MAE} := \frac{1}{|\mathbf{R}^t_{\text{test}}|_0} \sum_{(u,i^t)\,:\,\mathbf{R}^t_{\text{test}}[u,i^t]\neq 0} |\mathbf{R}^t_{\text{test}}[u, i^t] - \hat{r}(u, i^t)|, \tag{14}$$

where $|\mathbf{R}|_0$ counts the number of nonzero elements of a matrix $\mathbf{R}$.

Note that methods of single-domain recommendation (e.g., matrix factorization (MF)) cannot be applied to this task because such methods learn user representations only from $\mathbf{R}^t_{\text{train}}$, which does not contain any information of cold-start users U^t_{test} (Figure 6).

The methods in [2, 3, 14] were proposed for dealing with this situation by minimizing the following loss function with the regularizer of cross-domain user similarity:

$$\begin{aligned}
\text{loss} = {} & \frac{1}{|\mathbf{R}^s|_0} \|\mathbf{R}^s - \mathbf{P}^s(\mathbf{Q}^s)^T\|^2 & (15)\\
& + \frac{1}{|\mathbf{R}^t_{\text{train}}|_0} \|\mathbf{R}^t_{\text{train}} - \mathbf{P}^t_{\text{train}}(\mathbf{Q}^t)^T\|^2 & (16)\\
& + \frac{\lambda}{|U_{\text{train}}|^2} \sum_{u,u' \in U_{\text{train}}} \text{sim}(u, u') \|\mathbf{P}[u] - \mathbf{P}[u']\|^2, & (17)
\end{aligned}$$

where $\mathbf{Q}^z \in \mathbb{R}^{|I^z|\times k}$, $\mathbf{P}^s \in \mathbb{R}^{|U^s|\times k}$, $\mathbf{P}^t_{\text{train}} \in \mathbb{R}^{|U^t_{\text{train}}|\times k}$, k is a positive integer, $\lambda > 0$, $U_{\text{train}} := U^s \cup U^t_{\text{train}}$, and $\mathbf{P} = [\mathbf{P}^s, \mathbf{P}^t_{\text{train}}]$ [10]. In contrast to the original loss function used in [2, 3, 14], we divided each term by the number of elements to reduce adverse effects of imbalanced data. Indeed,

[10] In experiments, we chose $k = 50$ and $\lambda = 1$ by 5-fold cross validation.

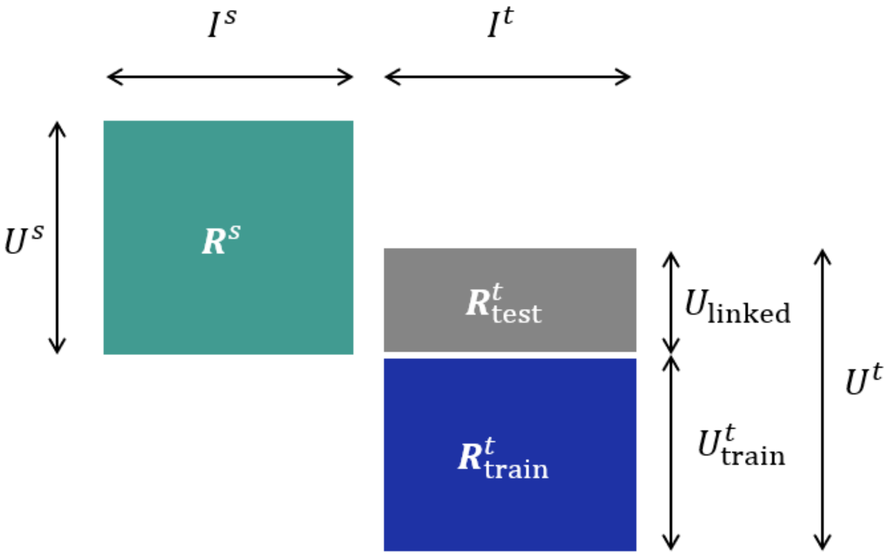


Figure 6: By removing $\mathbf{R}^t_{\mathrm{test}}$ from $\mathbf{R}^t$, both user and item sets of $\mathbf{R}^s$ and $\mathbf{R}^t_{\mathrm{train}}$ do not have any overlapping during the training stage, i.e., $U^s \cap U^t_{\mathrm{train}} = \emptyset$ and $I^s \cap I^t = \emptyset$. For this challenging situation, we address predicting ratings of cold-start users.

we observed that $|\mathbf{R}^s|_0 \approx 7 \cdot 10^6$ and $|\mathbf{R}^t_{\mathrm{train}}|_0 \approx 3 \cdot 10^5$ in Case 2. For ATP, we set $\mathrm{sim}(u, u') = \cos(\phi^z(u), \phi^z(u'))$ when u and u' belong to the same domain U^z; $\mathrm{sim}_{\mathrm{ATP}}(u, u')$, otherwise. After obtaining the trained embedding vectors, we predict a rating score as $\hat{r}(u, i^t) = \langle \mathbf{P}^s[u], \mathbf{Q}^t[i^t] \rangle$ for an item $i^t \in I^t$ by a cold-start user $u \in U^t_{\mathrm{test}}$.

As stated above, single-domain recommendation methods do not work for cold-start users, but we can apply a method that utilizes average ratings, called **AVE**, to this situation. **AVE** calculates the average of the rating scores attached to the item in the training dataset as $m_{i^t} = \mathrm{mean}\{\mathbf{R}^t_{\mathrm{train}}[u^t, i^t] \mid u^t \in U^t_{\mathrm{train}}\}$ and predicts a rating score for an item i^t by any user $u \in U^t$ as $\hat{r}(u, i^t) = m_{i^t}$. Since **AVE** was introduced as the minimum baseline, we calculate the improvement rate from **AVE** as

$$\text{Improvement rate (\%)} := 100 \times \frac{\mathrm{MAE}_{\mathrm{AVE}} - \mathrm{MAE}_*}{\mathrm{MAE}_{\mathrm{AVE}}} \tag{18}$$

where $* \in \{\mathrm{ATP}, \mathrm{Agg}, \mathrm{SCT}\}$ and $\mathrm{MAE}_{\mathrm{AVE}}$ is the MAE of **AVE**.

3.5.2. Results

The results shown in Table 5 indicate that **ATP** exhibited the highest MAE improvement rates for all cases except Case 4. Even for Case 4, **ATP** achieved the second-highest improvement rate and was only slightly worse than **SCT**-50. Unlike the results for Cases 5 and 6 in Section 3.4, **Agg** did not outperform **ATP**. One reason is that **Agg** could not give the degree of dissimilarity. Even if two users showed interest in similar (but not common) attributes, **Agg** uniformly treated them as entirely dissimilar users because their similarity score was zero. In contrast, **ATP** gives

Table 5
Results of our experiments for cross-domain recommendation in terms of MAE (left) and improvement rate compared with **AVE** (right, in percentage), where bold and underlined values represent the best and second-best methods in each column, respectively.

	Case 1		Case 2		Case 3	
ATP (Ours)	**0.759**	(+**8.36**%)	**0.726**	(+**1.4**%)	**0.731**	(+**3.09**%)
aggBM25	0.81	(+2.21%)	0.731	(+0.7%)	0.801	(< 0)
SCT-50	0.769	(+7.1%)	0.727	(+1.35%)	0.741	(+1.83%)
SCT-100	0.774	(+6.53%)	0.727	(+1.25%)	0.738	(+2.29%)
MAE of AVE	0.828		0.736		0.754	

	Case 4		Case 5		Case 6	
ATP (Ours)	0.62	(+9.68%)	**0.789**	(+**6.72**%)	**0.573**	(+**13.14**%)
aggBM25	3.271	(< 0)	0.793	(+6.22%)	0.641	(+2.73%)
SCT-50	**0.618**	(+**9.96**%)	0.789	(+6.68%)	0.578	(+12.29%)
SCT-100	0.623	(+9.3%)	0.79	(+6.62%)	0.601	(+8.89%)
MAE of AVE	0.686		0.845		0.659	

the degree of dissimilarity with a particular value, which is why it worked better than **Agg** and **SCT** in the recommendation task with cross-domain user similarity.

4. Related Work

Methods for calculating cross-domain user similarity and discovering the same user are known as methods of user identity linkage (UIL) [15, 16, 17, 18, 19, 20, 21, 6]. If users' real names are available in two datasets, the UIL problem can be solved using methods of named-entity linkage. Even if users' real names are unavailable, several UIL methods [16, 17] for social media have utilized users' screen names (e.g., Twitter ID and Instagram ID). Methods that do not use screen names use user behavior, such as a user's trajectory history [18, 19, 20, 21] or tag posting history [6]. However, most UIL methods focusing on trajectories are domain-specific, relying on the geographic coordinate system (latitude and longitude), grids, and zip codes. The UIL method proposed by Iofciu et al., [6] for a tagging system is the most relevant to this study in terms of focusing on a user's interests that are related to psychographic attributes. As mentioned in the experimental section (Section 3), their method relies on tags that appear in two domains as domain-bridging information.

From the viewpoint of user similarity, recommendation systems can be regarded as trying to extract user features that summarize their degrees of interest in items from their behaviors (e.g., purchase and evaluation histories for items). When users or items are registered to multiple datasets, standard (single-domain) recommendation methods provide each dataset's corresponding user features in different dimensions. To provide user features in a common dimension, methods in cross-domain recommendation (CDR) [1] have been developed; however, many impose the assumption that there are common users or items, i.e., $U^s \cap U^t \neq \emptyset$ or $I^s \cap I^t \neq \emptyset$, to connect heterogeneous datasets. If $U^s \cap U^t \neq \emptyset$, a method by Man et al., [22], trains the relationship between two user features of the same user in $U^s \cap U^t$ with supervised machine learning to obtain a feature for a user who has not yet been registered in one dataset from his/her feature in another dataset via a regression model. However, as with the UIL problem setting, we address a situation where user sets have no intersection. CDR methods [23, 24] handle situations in which neither user sets nor item sets have intersections to express user features of different domains in a common dimension. However, their resulting features are embedding vectors due to the matrix decomposition; hence, it is difficult to interpret and understand the features.

Tag-based CDR methods [14, 2, 3] utilizing additional information for bridging domains have been proposed. Methods of using tags differ in terms of whether tag sets have non-empty intersections. Several methods [14, 2] rely on tags that appear in both datasets, but this is only sometimes the case, and the number of such overlapping tags can be the method by Zhang et al., [3] handles the situation in which sets of users, items, and tags do not have intersections, which is the same as ATP. As mentioned in the experimental section (Section 3), however, their method cannot reflect tags in one domain that are not related to another, which is considered to harm calculating cross-domain user similarity. ATP overcomes this problem by introducing optimal transport theory.

5. Conclusion

In this paper, we proposed Attribute TransPortation (ATP), a novel method for calculating cross-domain user similarity without requiring assumptions for overlapping users or attributes. The core idea of ATP is to use optimal transport theory to provide attribute similarity for items that are not only semantic but also relevant to the two domains on which we focus. ATP transforms a target user into a feature vector of the other domain and then enables cross-domain user similarity in the same feature space to be calculated. The results of experiments using linkage of the same users and cross-domain recommendation tasks demonstrated the effectiveness of ATP compared to related methods.

References

[1] M. M. Khan, R. Ibrahim, I. Ghani, Cross domain recommender systems: A systematic literature review, ACM Comput. Surv. 50 (2017) 36:1–36:34.

[2] Y. Shi, M. A. Larson, A. Hanjalic, Tags as bridges between domains: Improving recommendation with tag-induced cross-domain collaborative filtering, in: UMAP, volume 6787 of *Lecture Notes in Computer Science*, Springer, 2011, pp. 305–316.

[3] Q. Zhang, P. Hao, J. Lu, G. Zhang, Cross-domain recommendation with semantic correlation in tagging systems, in: IJCNN, IEEE, 2019, pp. 1–8.

[4] Y. Zhu, Z. Tang, Y. Liu, F. Zhuang, R. Xie, X. Zhang, L. Lin, Q. He, Personalized transfer of user preferences for cross-domain recommendation, in: WSDM, ACM, 2022, pp. 1507–1515.

[5] C. Villani, Optimal transport: old and new, volume 338, Springer Science & Business Media, 2008.

[6] T. Iofciu, P. Fankhauser, F. Abel, K. Bischoff, Identifying users across social tagging systems, in: ICWSM, The AAAI Press, 2011.

[7] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Trans. Assoc. Comput. Linguistics 5 (2017) 135–146.

[8] S. Yokoi, R. Takahashi, R. Akama, J. Suzuki, K. Inui, Word rotator's distance, in: EMNLP (1), Association for Computational Linguistics, 2020, pp. 2944–2960.

[9] M. J. Kusner, Y. Sun, N. I. Kolkin, K. Q. Weinberger, From word embeddings to document distances, in: ICML, volume 37 of *JMLR Workshop and Conference Proceedings*, JMLR.org, 2015, pp. 957–966.

[10] M. Cuturi, Sinkhorn distances: Lightspeed computation of optimal transport, in: NIPS, 2013, pp. 2292–2300.

[11] R. Flamary, N. Courty, A. Gramfort, M. Z. Alaya, A. Boisbunon, S. Chambon, L. Chapel, A. Corenflos, K. Fatras, N. Fournier, L. Gautheron, N. T. Gayraud, H. Janati, A. Rakotomamonjy, I. Redko, A. Rolet, A. Schutz, V. Seguy, D. J. Sutherland, R. Tavenard, A. Tong, T. Vayer, POT: Python optimal transport, Journal of Machine Learning Research 22 (2021) 1–8.

[12] J. Ni, J. Li, J. J. McAuley, Justifying recommendations using distantly-labeled reviews and fine-grained aspects, in: EMNLP/IJCNLP (1), Association for Computational Linguistics, 2019, pp. 188–197.

[13] K. S. Jones, S. Walker, S. E. Robertson, A probabilistic model of information retrieval: development and comparative experiments, Inf. Process. Manag. 36 (2000) 779–840.

[14] Y. Zhen, W. Li, D. Yeung, Tagicofi: tag informed collaborative filtering, in: RecSys, ACM, 2009, pp. 69–76.

[15] K. Shu, S. Wang, J. Tang, R. Zafarani, H. Liu, User identity linkage across online social networks: A review, SIGKDD Explorations 18 (2016) 5–17.

[16] R. Zafarani, H. Liu, Connecting corresponding identities across communities, in: ICWSM, The AAAI Press, 2009.

[17] R. Zafarani, H. Liu, Connecting users across social media sites: a behavioral-modeling approach, in: KDD, ACM, 2013, pp. 41–49.

[18] O. Goga, H. Lei, S. H. K. Parthasarathi, G. Friedland, R. Sommer, R. Teixeira, Exploiting innocuous activity for correlating users across sites, in: WWW, International World Wide Web Conferences Steering Committee / ACM, 2013, pp. 447–458.

[19] J. Feng, M. Zhang, H. Wang, Z. Yang, C. Zhang, Y. Li, D. Jin, DPLink: User identity linkage via deep neural network from heterogeneous mobility data, in: WWW, ACM, 2019, pp. 459–469.

[20] W. Cao, Z. Wu, D. Wang, J. Li, H. Wu, Automatic user identification method across heterogeneous mobility data sources, in: ICDE, IEEE Computer Society, 2016, pp. 978–989.

[21] W. Chen, H. Yin, W. Wang, L. Zhao, X. Zhou, Effective and efficient user account linkage across location based social networks, in: ICDE, IEEE Computer Society, 2018, pp. 1085–1096.

[22] T. Man, H. Shen, X. Jin, X. Cheng, Cross-domain recommendation: An embedding and mapping approach, in: IJCAI, ijcai.org, 2017, pp. 2464–2470.

[23] B. Li, Q. Yang, X. Xue, Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction, in: IJCAI, 2009, pp. 2052–2057.

[24] S. Gao, H. Luo, D. Chen, S. Li, P. Gallinari, J. Guo, Cross-domain recommendation via cluster-level latent factor model, in: ECML/PKDD (2), volume 8189 of *Lecture Notes in Computer Science*, Springer, 2013, pp. 161–176.