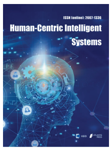
Human-Centric Intelligent Systems
Vol. **1(1–2)**; June (2021), *pp.* 43–53
DOI: https://doi.org/10.2991/hcis.k.210524.001; eISSN 2667-1336
https://www.atlantis-press.com/journals/hcis



Research Article

Context-Based User Typicality Collaborative Filtering Recommendation

Jinzhen Zhang*, Qinghua Zhang, Zhihua Ai, Xintai Li

Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China

ARTICLE INFO

Article History

Received 07 March 2021
Accepted 24 May 2021

Keywords

Knowledge granulation
contextual information
user typicality
recommendation
granular computing

ABSTRACT

Since contextual information significantly affecting users' decisions, it has attracted widespread attention. User typicality indicates the preference of user for different item types, which could reflect the preference of user at a higher abstraction level than the items rated by user, and can alleviate data sparsity. But it does not consider the impact of contextual information on user typicality. This paper proposes a novel context-based user typicality collaborative filtering recommendation algorithm (named CBUTCF), which combines contextual information with user typicality to alleviate the data sparsity of context-aware collaborative filtering, and extracts, measures and integrates contextual information. First, the items are clustered and classified into different item types. For different users, the significance of contextual information for different item types is defined and measured via knowledge granulation. Then, the contextual information is combined with user typicality to measure the context-based user typicality; subsequently, the 'neighbor' users are determined. Finally, the unknown ratings under a single context are predicted, and the unknown ratings under multi-context are predicted according to the weighted summation of the significance of contextual information. The experimental results demonstrate that CBUTCF can effectively improve the accuracy of recommendation and increase coverage.

1. INTRODUCTION

With the advent of Web 2.0, data on the internet have grown exponentially, and traditional information retrieval has failed to satisfy the users' needs. To gain insight about users' interests and present corresponding information to them, recommender systems have become the most potential tool.

Collaborative filtering (CF) is a classic and popular technology for recommender systems. It includes user-based CF [28] and item-based CF [29]. The idea of CF is to find 'neighbors' of all users (or items) according to the historical ratings, and recommend highly related items to the target user. Therefore, finding the 'neighbors' of users (or items) is an important step in CF. Currently, most CF algorithms measure the similarities between users (or items) based on user historical ratings to find 'neighbors'. However, CF is not suitable for users (or items) with few or no rating, called the data sparsity problem [14]. To address this problem, Ren et al. [27] used the rule extraction algorithm based on rough sets to extract the user & item attributes from the core value decision rules to fill in the rating table. Zhang et al. [36] integrated the social information of users with their rating information to fill in the expert ratings. Hawashin et al. [12] proposed a new efficient mixed similarity measure method for user interest-based recommender systems. Cai et al. [5] borrowed the idea of object typicality from cognitive psychology and proposed a typicality-based CF recommendation method. Cai's method, different from the matrix filling method, could solve the data sparsity problem. However, in some cases, it is not accurate to only consider the user's interaction with the item, and the contextual information may also have an impact on the user's decision-making.

In the field of recommender systems, Adomavicius et al. [2] believed that it is unreasonable to only rely on the user-item matrix to capture the preference relationships between users. Chen [6] believed that 'neighbors' of user have the same preference in similar environments. For example, some users originally like horror movies, but they watch cartoons with their children. Owing to the popularization of mobile network equipment, recommender systems can better obtain and collect more contextual information. Correspondingly, the researches on the context-aware recommendation theory have been further developed [7,9,17,20,22,33]. In previous researches, Yep et al. [35] combined the context attributes with Bayesian networks for prediction. Setten et al. [30] proposed the travel application COMPASS. Lee et al. [18] combined the contextual information with decision trees for a restaurant recommender systems. However, these algorithms considered all contextual factors together and did not distinguish the significance of different contextual information. Huang et al. [15] proposed a context-aware recommendation method using rough sets and CF to recommend suitable items in a specific context. This method considered the contextual dependencies of the items; however, it did not consider the differences in the dependencies of the items between users, and user-based CF method was inflicted the effects of data sparsity problem, which inspired our study.

[*] *Corresponding author. Email: 799232295@qq.com*

Peer review under responsibility of KEO (Henan) Education Technology Co. Ltd

Knowledge granulation which is a new uncertainty measure formula proposed by Liang [19]. It can be used to measure the significance of attributes in complex information. Zhang [37] proposed a utility function with both knowledge space granularity and approximation accuracy for automatically searching optimal knowledge space according to user's requirement. Jing [16] introduce incremental mechanisms to compute new knowledge granularity and developed the corresponding incremental algorithms for attribute reduction. Xu [34] considered relationship between knowledge granulation, knowledge entropy and knowledge uncertainty measure, and introduce definition of rough entropy of rough sets in ordered information systems, which is an application of knowledge granulation.

Therefore, for context-aware collaborative filtering affected by data sparsity, using user typicality instead of user ratings to express user preferences can effectively alleviate data sparsity. This paper combines user typicality with contextual information, which can not only consider user's preferences in contextual information, but also alleviate data sparsity. The original algorithm did not explicitly distinguish the significance of information in different contexts, which does not conform to human cognition. Regarding the significance of distinguishing different contextual information, this paper introduces knowledge granulation, which can distinguish the significance of different contextual information in complex information. On this basis, this paper proposes a context-based user typical collaborative filtering recommendation algorithm, and a large number of experimental results illustrate that the algorithm in this paper can achieve a better recommendation effect.

Therefore, the main innovations of this paper can be summarized as follows:

(1) Combine user typicality with contextual information, and consider user preferences at a higher level under contextual information.
(2) The introduction of knowledge granularity is used to measure the user's preference for item types under the context type.

The remainder of this paper is organized as follows. Section 2 introduces the related work and basic knowledge. Section 3 describes the formal definition and basic model of the context-based user typicality. Section 4 introduces CBUTCF in detail. The experimental results are presented in Section 5. Section 6 concludes the present work and outlines future research.

2. RELATED WORK

To further discuss the ideas of this study, this section reviews the relevant definitions of user typicality, context-aware recommender systems, and knowledge granulation. User typicality is a concept for describing preference of user. Context-aware recommendation is the integration of contextual information into recommender systems. Knowledge granulation describes the degree of subdivision of knowledge.

2.1. User Typicality

To solve the inaccurate measurement of similarities between users caused by data sparsity problem in traditional CF, Cai et al. [5] borrowed the concept of object typicality from cognitive psychology and proposed a typicality-based CF recommendation method named TyCo. First, items with similar attributes were grouped into a category called 'item group'. Second, for each item group, there existed a corresponding group called the 'user group'. The user group corresponding to the item group was considered as a fuzzy concept, i.e., 'users who like the items which in the item group'; additionally, all users hava different degrees of user typicality in each user group. Third, a user-typicality matrix was constructed and the similarities between users were measured according to the degree of user typicality in all user groups; subsequently, a set of 'neighbors' was determined for the target user. Finally, the unknown rating of the target user was predicted based on the ratings of the 'neighbors' on the same item. User typicality indicates the preference of the user for different item types. Assume that in a CF recommender systems, there are a set U of users, and a set O of items, and its official definition can be given as follows.

Definition 1. [Item group] [5] An item group t_i is defined as:

$$t_i = \{o_1^{\omega_{i,1}}, o_2^{\omega_{i,2}}, ..., o_n^{\omega_{i,n}}\}, \tag{1}$$

where n is the number of items in t_i, o_y denotes an item and $\omega_{i,y}$ denotes the membership of o_y in t_i.

Definition 2. [User group] [5] A user group g_i is defined as:

$$g_i = \{u_1^{\upsilon_{i,1}}, u_2^{\upsilon_{i,2}}, ..., u_m^{\upsilon_{i,m}}\}, \tag{2}$$

where m denotes the member of users in g_i, u_x denotes a user in g_i, and $\upsilon_{i,x}$ denotes the degree of user typicality for u_x in g_i (i.e., the preference of user u_x for the t_i).

The measurement method of $v_{i,x}$ is as follows:

$$v_{i,x} = \tau_{g_i}(u_x) = \frac{s^x_{g_i,r} + s^x_{g_i,f}}{2}, \tag{3}$$

where the degree of user typicality $v_{i,x}$ is measured by combination function $\tau_{g_i}(u_x)$ of $s^x_{g_i,r}$ and $s^x_{g_i,f}$. The higher the values of $s^x_{g_i,r}$ and $s^x_{g_i,f}$, the more degree of user typicality, thus, the user is as a member in g_i.

$s^x_{g_i,r}$ denotes the weighted summation of all ratings of the corresponding t_i by u_x; it can be measured as follows:

$$s^x_{g_i,r} = \frac{\sum_{y=1}^{n} \omega_{i,y} \cdot R_{x,y}}{n \cdot R^{\max}}, \tag{4}$$

where n denotes the number of items rated by u_x in t_i, $R_{x,y}$ denotes the rating of u_x on o_y, $\omega_{i,y}$ denotes the degree to which o_y belongs to t_i, and $R^{\max}$ denotes the maximum rating value.

$s^x_{g_i,f}$ denotes the frequency items rated by u_x in t_i; it can be measured as follows:

$$s^x_{g_i,f} = \frac{N_{i,x}}{N_x}, \tag{5}$$

where $N_{i,x}$ denotes the number of items rated by u_x in t_i and N_x denotes the number of items rated by u_x in all item groups.

Definition 3. [User typicality vector] [5] A user typicality vector $\vec{u}_x$ of u_x is a vector of real numbers in the closed interval [0, 1]; it is defined as follows:

$$\vec{u}_x = (\upsilon_{1,x}, \upsilon_{2,x}, ..., \upsilon_{n,x}), \tag{6}$$

where n denotes the number of user group, and $\upsilon_{i,x}$ denotes the degree of user typicality for u_x in g_i.

Definition 4. [User-typicality matrix] [5] A user-typicality matrix, denoted by M_T, is composed of user typicality vector of each user, and defined as follows:

$$M_T = \left\{ \begin{array}{c} \vec{u}_1 \\ \dots \\ \vec{u}_m \end{array} \right\} = \left\{ \begin{array}{c} v_{1,1}, v_{2,1}, \dots, v_{n,1} \\ \dots \\ v_{1,m}, v_{2,m}, \dots, v_{n,m} \end{array} \right\}, \quad (7)$$

where m denotes the number of user typicality vectors, n denotes the number of user groups, and $\vec{u}_x$ denotes the user typicality vector of u_x.

2.2. Context-Aware Recommender Systems

In previous research, Adomavicius et al. [2] believed that combining contextual information with recommender systems could help in improving the accuracy of prediction, and proposed the concept of 'context-aware recommender systems' [3], referred to as CARS [1]. They extended the 'user-item' two-dimensional model, $User \times Item \rightarrow rating$, to a multi-dimensional model containing contextual information, $User \times Context \times Item \rightarrow rating$. Currently, there is no standard definition of 'context'. Dey [10] provided a relatively normative definition of context; context refers to the environment itself and the information expressed by the entities present in the environment. Entities denote users, geographic locations, or related objects that interact with users and applications.

Definition 5. The formal definition of contextual information [3] is as follows:

$$\begin{aligned} C &= (C_1, C_2, \dots, C_w), \\ C_h &= (C_1, C_2, \dots) \end{aligned} \quad (8)$$

where $C_h = C(h = 1, 2, \dots, w)$ denotes the context attribute, such as the location, time, and weather; $C_{h,k}$ denote the k-th attribute value of the h-th context attribute, named context state, such as *morning*, *noon*, *evening* in *Time*. Examples of contextual information are as follows:

$$C_1 : Time(morning, noon, evening),$$
$$C_2 : Location(home, theater).$$

The context of the user is defined as context combination c, such as a user is at night and at home at the same time. And the task of CARS [3] is to predict the user ratings under context combination c.

$$\hat{R}_{u,c,o} = f(u, c, o), \quad (9)$$

where $f(\cdot)$ denotes the prediction model, and $\hat{R}_{u,c,o}$ denotes the prediction rating for user u interacting with item o under context combination c.

2.3. Rough Sets

The basic idea of the rough set theory is to form concepts and rules through the classification and induction of a relational database, and discover knowledge through the classification of indistinct relationships and approximation of targets [21]. In this chapter, the theory of rough sets and the measurement of knowledge granulation are briefly introduced.

Definition 6. [Decision information system] [25] A decision information system can be expressed as $S = (I, A, V, f)$, where I is the complete set of objects, also known as universe. $A = C \cup D$ is the complete set of attributes, and subsets C and D are called the conditional attribute set and decision attribute set, respectively. $V = \bigcup_{r \in A} V_r$ is the set of attribute values. $f : I \times A \rightarrow V$ is an information function that specifies the attribute value of each object x in I.

Definition 7. [Indiscernibility relation] [25] Given a decision information system $S = (I, A, V, f)$, $A = C \cup D$, for each attribute subset $B \subseteq A$, an indiscernibility relation $IND(B)$ on universe I can be defined as:

$$IND(B) = \{(x, y) \left| (x, y) \in I^2, \forall_{b \in B}(b(x) = b(y)) \right. \}, \quad (10)$$

Obviously, the indiscernibility relation satisfies reflexivity, symmetry and transitivity, so $INB(B)$ is an equivalence relation in I. The equivalence relation $INB(B)$ induces a partition of I. An object $x \in I$ is described by its equivalence class of $I/IND(B)$:$[x]_{INB(B)}$, or simply $[x]$ and $[x]_B$. The pair $(I, IND(B))$ is called an approximation space.

Definition 8. [Knowledge granulation] [19,37] Given a decision information system $S = (I, A, V, f)$, $A = C \cup D$, $I/IND(B) = \{X_1, X_2, \dots, X_n\}$, the knowledge granulation of A, denoted by $G(B)$, can be described as follows:

$$G(B) = \frac{1}{|I|^2} \sum_{i=1}^{n} |X_i|^2, \quad (11)$$

where $\sum_{i=1}^{n} |X_i|^2$ denotes the cardinality of the equivalence relationship determined by $\bigcup_{i=1}^{n} (X_i \times X_i)$. Knowledge granulation can describe the distinguishing ability of knowledge; it should be noted that the distinguishing ability is stronger, when the knowledge granulation is smaller.

3. CONTEXT-BASED USER TYPICALITY

Some formal definitions of CBUTCF are introduced in Section 3.1, the mechanism of CBUTCF is described in Section 3.2, and the correlation between knowledge granulation and context attributes is discussed in Section 3.3.

3.1. Preliminaries

Assume that $O = \{o_1, o_2, \dots, o_n\}$ is a set of items, $U = \{u_1, u_2, \dots, u_m\}$ is a set of users, $C = \{c_1, c_2, \dots, c_w\}$ is a set of context attributes, and $T = \{t_1, t_2, \dots, t_z\}$ is a set of item groups. The items can be classified into different item groups; the items in a item group have similar attributes, and an item can only be classified into one item group. For example, all movie can be classified into horror, action, comedy movies, and so on. These items are classified into different groups with K-means [23], so the degree of one item belonging to one item group is only 1 or 0. Because the clustering method is not included in the scope of this study, it will not be discussed here. These formal definitions can be described as follows.

Definition 9. [Context-based decision information table] Construct a context-based decision information table: $RS =< R, C, T, V, f, \phi, \theta >$ using the historical ratings of a user, where

- RS denotes a non-empty finite set of ratings, R denotes one rating;

Table 1 Examples of context-aware recommendation

Ratings	User	Time	Location	Companion	Item type (Item)	Rating value
R_1	u_1	Weekend	Home	Friend	$3(I_1)$	5
R_2	u_1	Weekend	Home	Friend	$3(I_{18})$	3
R_3	u_1	Weekend	Theater	Friend	$3(I_{100})$	5
R_4	u_1	Weekend	Theater	Alone	$3(I_{56})$	4
R_5	u_1	Weekday	Theater	Family	$4(I_{30})$	4
R_6	u_1	Weekday	Home	Friend	$1(I_{13})$	3

- C denotes a non-empty finite set of context attributes (set of conditional attributes), $C = (C_1, C_2, ..., C_w)$, where C_h denotes a context attribute, $C_{h,k}$ denotes a variable of the context attribute (named context state);
- D denotes a non-empty finite set of item type (set of decision attributes), t_i denotes a item type;
- $\phi : R \to D$ denotes a function of the corresponding item type for rating R;
- $\theta : R \to D$ denotes a function of the context attribute of rating R.

For example, Table 1 presents a context-aware recommendation in the form of a decision information table, wherein the rows contain a set of ratings $RS\{R_1, R_2, ..., R_6\}$. The columns consist of user u_1, set of context attributes C (time, location, and companion), item type t_i, and rating value. Context state $C_{h,k}$ refers to the variable of a context attribute, such as 'Weekday' or 'Weekend' in time. The rating value of a user depends on these context states; correspondingly, the rating predictions depend on other user ratings under the same context state.

These ratings can be divided into different decision classes according to different item type. For example, for ratings of item type 3, the decision class $\{R_1, R_2, R_3, R_4\}$ can be obtained, and the decision class can be divided into 3 equivalence classes $\{R_1, R_2\}$, $\{R_3\}$, and $\{R_4\}$ according to the context attributes. This indicates that ratings R_1 and R_2 are indistinguishable, while other ratings can be uniquely identified using the context states.

Under the same context state, users with similar interests in a item group could form a community, called a context-based user group. The context-based user typicality indicates the preference of a user for a specific item group under a context state. Under the same context state, users can have different context-based user typicality in different item groups; additionally, in different context states, users can have different context-based user typicality in the same item group.

Definition 10. [Context-based user group] A context-based user group $g_i^{C_{h,k}}$ can be defined as:

$$g_i^{C_{h,k}} = \{u_{v_{i,1}}^{C_{h,k}}, u_{v_{i,2}}^{C_{h,k}}, ..., u_{v_{i,m}}^{C_{h,k}}\}, \tag{12}$$

where m denotes the number of users in $g_i^{C_{h,k}}$, and $u_{v_{i,x}}^{C_{h,k}}$ denotes the degree of context-based user typicality for u_x in g_i in $C_{h,k}$ (i.e., the preference of u_x for t_i in $C_{h,k}$).

The degree of context-based user typicality for u_x in g_i in $C_{h,k}$ is measured by combination function $s_{x,g_{i,r}}^{C_{h,k}}$ and $s_{x,g_{i,f}}^{C_{h,k}}$.

$$u_{v_{i,x}}^{C_{h,k}} = \tau_{g_i}^{C_{h,k}}(u_x) = \frac{s_{x,g_{i,r}}^{C_{h,k}} + s_{x,g_{i,f}}^{C_{h,k}}}{2}, \tag{13}$$

where $s_{x,g_{i,r}}^{C_{h,k}}$ denotes the weighted summation of all ratings of u_x in t_i in $C_{h,k}$. It can be measured as:

$$s_{x,g_{i,r}}^{C_{h,k}} = \frac{\sum_{y=1}^{n} R_{x,y}^{C_{h,k}}}{n \cdot R^{\max}}, \tag{14}$$

where n denotes the number of items rated by u_x in t_i in $C_{h,k}$, $R_{x,y}^{C_{h,k}}$ denotes the rating of u_x on o_y in $C_{h,k}$, and $R^{\max}$ denotes the maximum value of the rating.

$s_{x,g_{i,f}}^{C_{h,k}}$ denotes the frequency of items rated by u_x in t_i in $C_{h,k}$. It can be measured as follows:

$$s_{x,g_{i,f}}^{C_{h,k}} = \frac{N_{i,x}^{C_{h,k}}}{N_x^{C_{h,k}}}, \tag{15}$$

where $N_{i,x}^{C_{h,k}}$ denotes the number of items rated by u_x in t_i in $C_{h,k}$, as well as the number of items rated by u_x in all item groups in $C_{h,k}$.

Definition 11. [Context-based user typicality vector] In $C_{h,k}$, users can have different user typicality in different context-based user groups. The preference of a user can be denoted by a context-based user typicality vector. The context-based user typicality vector $\vec{u}_x^{C_{h,k}}$ of u_x in $C_{h,k}$ can be defined in the closed interval [0, 1] as follows:

$$\vec{u}_x^{C_{k,h}} = (v_{1,x}^{C_{h,k}}, v_{2,x}^{C_{h,k}}, ..., v_{n,x}^{C_{h,k}}), \tag{16}$$

where n denotes the number of context-based user groups, and $v_{i,x}^{C_{h,k}}$ denotes the degree of context-based user typicality for u_x in $g_i^{C_{h,k}}$.

Definition 12. [Context-based user-typicality matrix] In $C_{h,k}$, the context-based user-typicality matrix can be obtained for all users, defined as $M_T^{C_{h,k}}$, where each row of $M_T^{C_{h,k}}$ is composed of the context-based user typicality vector of each user; it can be defined as:

$$M_T^{C_{h,k}} = \left\{ \begin{array}{c} \vec{u}_1^{C_{h,k}} \\ ... \\ \vec{u}_m^{C_{h,k}} \end{array} \right\} = \left\{ \begin{array}{c} v_{1,1}^{C_{h,k}}, v_{2,1}^{C_{h,k}}, ..., v_{n,1}^{C_{h,k}} \\ ... \\ v_{1,m}^{C_{h,k}}, v_{2,m}^{C_{h,k}}, ..., v_{n,m}^{C_{h,k}} \end{array} \right\}, \tag{17}$$

where m denotes the number of users, n denotes the number of context-based user groups, and $\vec{u}_x^{C_{h,k}}$ denotes the context-based user typicality vector of u_x.

3.2. Knowledge Granulation of Context Attribute for Item Type

Knowledge granulation is used to measure the uncertainty between objects. When the knowledge granulation is larger, it indicates that the user makes more decisions in the current context attribute; thus, it can be considered as the significance of the context attribute. Therefore, the greater the knowledge granulation, the greater the significance of the context attribute. This algorithm aims to determine the correlation between the context attributes of different users and item types in context-aware recommendation. If C_h uniquely determines the item type, t_i depends entirely on C_h. The significance of C_h for t_i can be defined as:

$$G(t_i|C_h) = \frac{1}{\left|[R]_{t_i}\right|^2} \sum_{i=1}^{n} |R_i|^2, \tag{18}$$

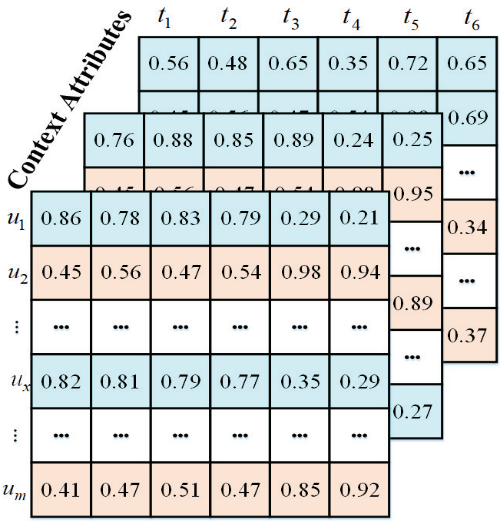
where $[R]_{t_i} = \{R \in RS\,|\phi(R) = t_i\}$, $R_i = \{R \in RS|\phi(R) = t_i, \theta(R) = C_h\}$, n denotes the number of R_i. For this user, if $G(t_i|C_h) = 1$, it means that the knowledge granulation of C_h for t_i is 1. If $0 < G(t_i|C_h) < 1$, it means that C_h is partially important for t_i. If $G(t_i|C_h) = 0$, then the knowledge granulation of C_h for t_i is 0.

3.3. Mechanism of CBUTCF

The mechanism of context-based user typicality CF recommendation includes three steps, as follows:

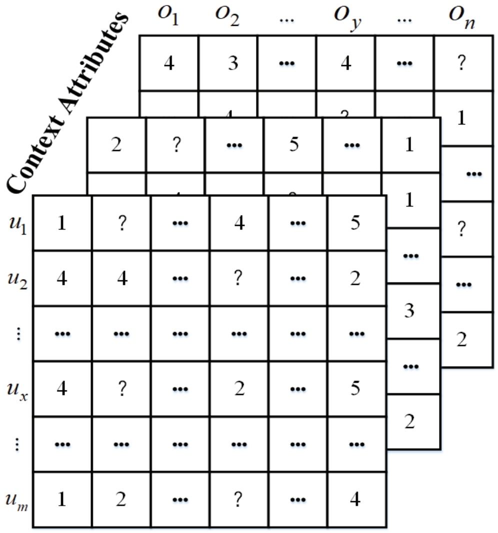
(1) The ratings of each user are separately constructed into a context-based decision information table, and each decision information table contains the ratings of only one user. First, the ratings are induced partitions based on t_i to obtain the rating decision classes $[o]_{t_i}$. Second, $[o]_{t_i}$ is divided according to C_h to obtain the rating equivalence classes; then, the knowledge granulation $G(t_i\,|C_h\,)$ is measured. Thus, the significance of context attributes for item type can be obtained.

(2) For each t_i, there exists a corresponding context-based user group $g_i^{C_{h,k}}$ in $C_{h,k}$. In $C_{h,k}$, users can have different user typicality for each user group $g_i^{C_{h,k}}$. In $C_{h,k}$, the user typicality of u_x in different user groups forms a context-based user typicality vector $\vec{u}_x^{C_{h,k}}$; then, the context-based user typicality vector of each user forms a context-based user-typicality matrix $M_T^{C_{h,k}}$. Therefore, in different $C_{h,k}$, there are different context-based user-typicality matrices $M_T^{C_{h,k}}$. The similarity between users can be measured according to the context-based user-typicality matrix; subsequently, a set of 'neighbors' with the largest similarity is determined for the target user. For example, according to Figure 1, because u_1 and u_x have similar typicality under the first context state, they are similar users under this context state.


Context Attributes
0.56
0.48
0.65
0.35
0.72
0.65
0.69
···
0.34
···
0.37
0.76
0.88
0.85
0.89
0.24
0.25
0.95
···
0.89
···
0.27

	t_1	t_2	t_3	t_4	t_5	t_6
u_1	0.86	0.78	0.83	0.79	0.29	0.21
u_2	0.45	0.56	0.47	0.54	0.98	0.94
⋮	···	···	···	···	···	···
u_x	0.82	0.81	0.79	0.77	0.35	0.29
⋮	···	···	···	···	···	···
u_m	0.41	0.47	0.51	0.47	0.85	0.92

Figure 1 Example of context-based user-typicality matrix.


Context Attributes
4
3
···
4
···
?
1
···
?
···
2
2
?
···
5
···
1
1
···
3
···
2

	o_1	o_2	...	o_y	...	o_n
u_1	1	?	···	4	···	5
u_2	4	4	···	?	···	2
⋮	···	···	···	···	···	···
u_x	4	?	···	2	···	5
⋮	···	···	···	···	···	···
u_m	1	2	···	?	···	4

Figure 2 Example of contextual user-item rating matrix.

In previous algorithms, the preference of users were inferred based on the contextual user-item rating matrix. Figure 2 illustrates a contextual user-item rating matrix. In CBUTCF, a user is denoted by a context-based user typicality vector, for whom each element can be considered as a feature of the user under the context state. This representation method can portray the preference of users under contextual information more accurately than the traditional context-aware CF algorithms.

(3) The unknown rating of the target user in $C_{h,k}$ is predicted based on the ratings of 'neighbors' of the target user on the same item. Subsequently, according to the significance of the context attributes for the item types, the predicted rating of the complete context state for a user can be obtained by weighted summation.

4. CONTEXT-BASED USER TYPICALITY RECOMMENDATION

To measure preference of users in contextual information, context-based user typicality recommendation algorithm was proposed, which combines contextual information with user typicality.

First, the importance of the context attribute to the item type is measured via knowledge granulation. Second, the similarity between users is measured according to context-based user typicality vector; subsequently, the 'neighbors' can be determined. Finally, the rating of item is predicted under context combination. And CBUTCF is described in Section 4.4.

4.1. Significance of Context Attribute for Item Type

Various types of contextual attribute appear in context-aware recommendation. However, some context attributes have a greater impact on users' decisions, while others have little or no impact. To recommend suitable items in context combination, the significance of context attributes for different item types must be determined for each user.

Definition 13. [Significance of context attribute for item type] Let $C, C_h \in C$, and t_i be a set of contextual information, context attribute, and item type, respectively. For u_x, a set of the significance of context attribute for item type α_i^x is defined as:

$$\alpha_i^x = \{\beta_{1,i}^x, \beta_{2,i}^x, ..., \beta_{w,i}^x\}, \tag{19}$$

where $\beta_{h,i}^x$ denotes the significance of C_h for t_i about u_x.

The significance of context attribute for item type refers to the degree of preference of a user for an item type under a context attribute. The measurement process of α_i^x is shown in Algorithm 1.

Algorithm 1 Significance of context attribute for item type

```
Input: a set of ratings $R$, a set of context attributes $C$, a set of item
       types $D$
Output: the corresponding set of significance $\alpha_i^x$ for each $t_i$
for each $t_i \in D$ do
    $h = 1; w = |C|; \alpha_i^x = \emptyset$;
    for h from 1 to w do
        Compute $G(t_i|C_h)$;
    end
    Compute $\beta_{h,i}^x = \frac{G(t_i|C_h)}{\sum_{h=1}^{w} G(t_i|C_h)}$ and $\alpha_i^x = \alpha_i^x \cup \{\beta_{h,i}^x\}$;
end
```

4.2. Similarity Measure and Neighbor Selection

A set of 'neighbors' of u_x in $C_{h,k}$ is denoted by $\vec{n}_x^{C_{h,k}}$, i.e.,

$$\vec{n}_x^{C_{h,k}} = \{u_j \,|Max_k \,(Sim^{C_{h,k}}(u_x, u_j))\}, \tag{20}$$

where $Sim^{C_{h,k}}(u_x, u_j)$ is the similarity of u_x and u_j in $C_{h,k}$, and Max_k denotes the K users with the *K-highest* similarity.

If a candidate user u_j has the *K-highest* similarity of target user u_x, they will be selected as $\vec{n}_x^{C_{h,k}}$. A user in $C_{h,k}$ is denoted by a context-based user typicality vector. Various measurement formulas can be utilized to measure the similarity. In this study, the similarity of users u_x and u_j is measured using the cosine similarity [29] between these two context-based user typicality vectors,

$$Sim^{C_{h,k}}(u_x, u_j) = \frac{\vec{u}_x^{C_{h,k}} \cdot \vec{u}_j^{C_{h,k}}}{\left\| \vec{u}_x^{C_{h,k}} \right\| \times \left\| \vec{u}_j^{C_{h,k}} \right\|}, \tag{21}$$

where '$\cdot$' denotes the dot-product operator of two vectors, $\vec{u}_x^{C_{h,k}}$ denotes the context-based user typicality vector of a user in $C_{h,k}$, where $\|\bullet\|$ denotes the norm operator.

4.3. Prediction Based on Context

After obtaining these 'neighbors' of the target user in $C_{h,k}$, the rating of target user u_x on o_y is predicted in $C_{h,k}$ according to the ratings of other 'neighbors' $\vec{n}_x^{C_{h,k}}$ rated on the same item in $C_{h,k}$, denoted by $\hat{R}_{(x,c,y)}^{C_{h,k}}$. The formula is as follows:

$$\hat{R}_{(x,c,y)}^{C_{h,k}} = \frac{\sum\limits_{u_j \in \vec{n}_x^{C_{h,k}}} Sim^{C_{h,k}}(u_x, u_j) \bullet R(u_j, o_y)}{\sum\limits_{u_j \in \vec{n}_x^{C_{h,k}}} Sim^{C_{h,k}}(u_x, u_j)}, \tag{22}$$

where $\vec{n}_x^{C_{h,k}}$ denotes the 'neighbors' of u_x in $C_{h,k}$, $R(u_j, o_y)$ denotes the rating of u_j on o_y, and $Sim^{C_{h,k}}(u_x, u_j)$ denotes the similarity between u_x and u_j in $C_{h,k}$.

Then, the predicted rating $\hat{R}_{x,c,y}$ of target user u_x for o_y under context combination c can be denoted as

$$\hat{R}_{x,c,y} = \frac{\sum_{\beta_{h,i}^x \in \alpha_i^x} \beta_{h,i}^x \bullet R_{x,c,y}^{C_{h,k}}(u_x, o_y)}{\sum_{\beta_{h,i}^x \in \alpha_i^x} \beta_{h,i}^x}, \tag{23}$$

where $\beta_{h,i}^x$ denotes the significance of C_h for t_i that contained o_y for u_x.

4.4. Description of CBUTCF

To better present the logic and flow of the proposed algorithm in this study, the algorithm process of context-based user typicality CF recommendation can be described as follows (Algorithm 2), and the algorithm flow is illustrated in Figure 3.

Algorithm 2 Context-based user typicality CF recommendation

```
Input: user-context-item rating $R$, user set $U = \{u_1, u_2, ..., u_m\}$,
       item group $T = \{t_1, t_2, ..., t_z\}$, set of context attribute $C =
       \{C_1, C_2, ..., C_w\}$
Output: predicted rating $\hat{R}_{x,c,y}$
for h from 1 to w do
    for x from 1 to m do
        for i from 1 to z do
            Compute $u_{v_{i,x}}^{C_{h,k}}$ according to Eq. (13);
        end
        Construct $\vec{u}_x^{C_{k,h}}$ according to Eq. (16);
    end
    Construct $M_T^{C_{h,k}}$ according to Eq. (17);
end
Compute $\alpha_i^x$ according to Algorithm 1;
Compute user similarity according to Eq. (21) and select
  'neighbors';
Compute $\hat{R}_{(x,c,y)}^{C_{h,k}}$ according to Eq. (22);
Compute $\hat{R}_{x,c,y}$ according to Eq. (23);
```

5. EXPERIMENTS

To better demonstrate the effectiveness of CBUTCF, a series of experiments were conducted to compare CBUTCF with other CF algorithms. These experiments aimed to answer two questions: "an contextual information improve the accuracy?" and "an the proposed algorithm help in solving the data sparsity problem for recommendation?"

5.1. Data Set Description

In the experiments, three real data sets, LDOS-CoMoDa, Restaurant & Consumer and Filmtrust data sets, were used to evaluate performance of CBUTCF in this study. The LDOS-CoMoDa [4,24] contains 2296 ratings, assigned by 121 users on 1232 movies, and the ratings follow the numerical scale of 1 (bad) to 5 (excellent).

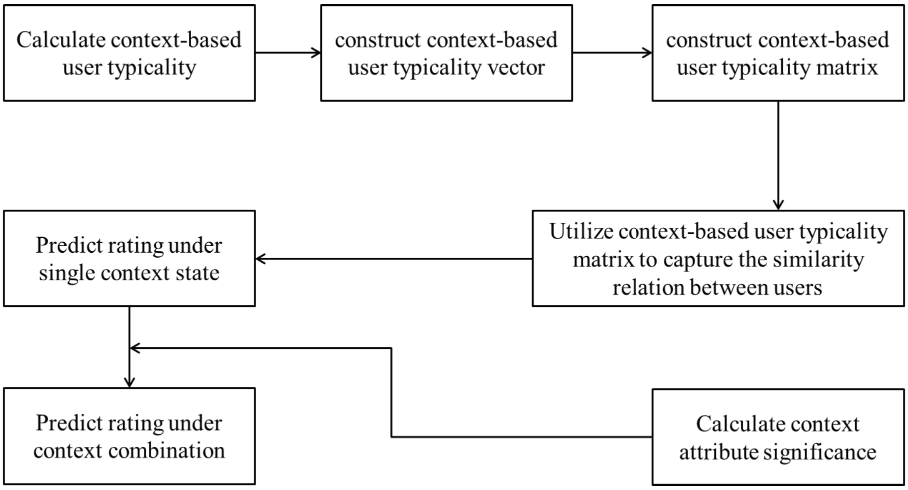

Calculate context-based user typicality
construct context-based user typicality vector
construct context-based user typicality matrix
Utilize context-based user typicality matrix to capture the similarity relation between users
Predict rating under single context state
Predict rating under context combination
Calculate context attribute significance

Figure 3 Flowchart of context-based user typicality CF recommendation.

The sparsity level of the data set is $1 - \frac{2296}{121\times1232}$, which is 0.9846. Restaurant & Consumer [8,11] contains 1161 ratings collected from 138 users on 130 restaurants, and the rating values are 0, 1, and 2, where 0,1,2 indicate that the user dislike, generally like, high preference for the restaurant, respectively. The sparsity level of the Restaurant & Consumer data set is $1 - \frac{1161}{138\times130}$, which is 0.9352. Filmtrust contains 35497 ratings, assigned by 1508 users on 2071 movies, and the ratings follow a digital scale from 1 (poor) to 5 (excellent). The sparsity of the data set is $1 - \frac{35497}{1508\times2071}$, which is 0.9896.

In this study, by adding reasonable context generation rules to the two data sets Restaurant & Consumer and Filmtrust, the simulated real data sets named C-R&C and C-Filmtrust were constructed, so that C-R&C and C-Filmtrust have the same contextual information with LDOS-CoMoDa.

The following contextual information was used in the experiments,

(1) Time: time period for watching movies (morning, noon, evening, early morning).
(2) Day type: types of days to watch movies (weekdays, weekends, holidays).
(3) Season: types of season to watch movies (spring, summer, autumn, winter).
(4) Location: location of watching movies (home, public places, friend home).

5.2. Metrics

To evaluate the performance of recommendation algorithm, the mean absolute error (MAE), root mean squared error (RMSE), and Coverage [13,31] were used as the evaluation indicators.

MAE and RMSE indicate the average error between the predicted and actual values. The lower these values, the better the recommendation effect of the recommendation algorithm. The MAE and RMSE are as follows:

$$MAE = \frac{\sum_{R_{x,y}\in R_{test}} \left|\hat{R}_{x,y} - R_{x,y}\right|}{|R_{test}|}, \tag{24}$$

$$RMSE = \sqrt{\frac{\sum_{R_{x,y}\in R_{test}} \left(\hat{R}_{x,y} - R_{x,y}\right)^2}{|R_{test}|}}, \tag{25}$$

where $\hat{R}_{x,y}$ denotes the rating predicted by user u_x for item o_y, $R_{x,y}$ denotes the actual rating of user u_x for item o_y, and $|R_{test}|$ denotes the number of ratings in the test set.

Coverage indicates the ratio of predictable items to all items that need to be predicted; it helps in evaluating the comprehensiveness of prediction. When the coverage is higher, the recommendation effect is more comprehensive. Coverage is as follows:

$$Coverage = \frac{|\cup_{u\in U} R(u)|}{|O|}, \tag{26}$$

where U represents the set of users, $R(u)$ a list of ratings for each user, and $|O|$ represents the number of items in the data set.

5.3. Experimental Results

The experiments extract four train set ratios, which 20%, 40%, 60%, 80% of the LDOS-CoMoDa as the train set, respectively, and the rest as the test set. In CBUTCF, the number of user groups depends on the number of item groups. To measure the effect of different numbers of user groups on the experimental results, the experiments were conduct on the number of user groups (i.e., i) from 5 to 25. Figures 4–6 illustrate the results for the MAE, RMSE, and Coverage in different train set ratios.

Figures 4–6 illustrate the influence of the number of different user groups in four train set ratios on the three recommendation metrics. The experiments illustrated that the number of user groups has little effect on the recommendation results under the same train set ratios. Therefore, the recommendation effect is considered to be stable under different user groups.

To illustrate the recommendation performance of CBUTCF, it was compared with the following three algorithms using the same similarity function (Cosine similarity),

(1) User-based CF algorithm (UBCF) [28]. The UBCF first finds out a set of nearest 'neighbors' (similar users) for each user, then

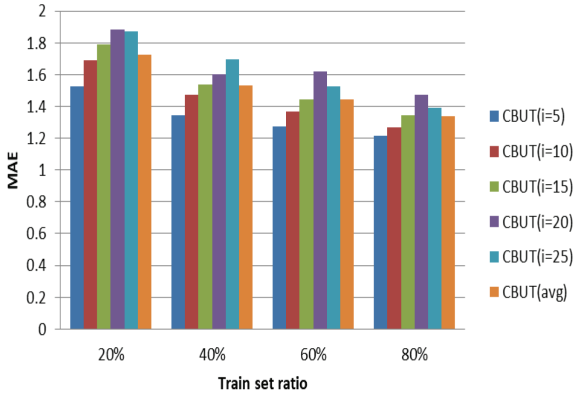


Figure 4 Effect of *i* on MAE in different train set ratios.

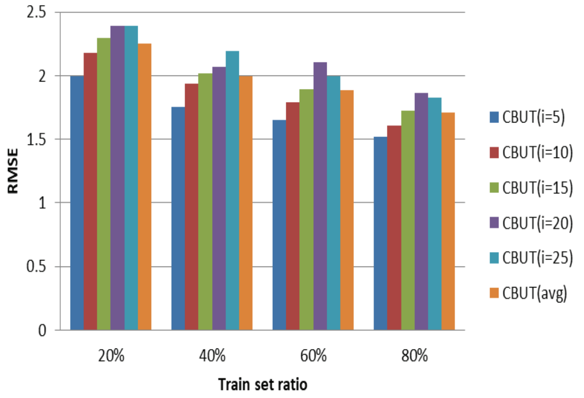


Figure 5 Effect of *i* on RMSE in different train set ratios.

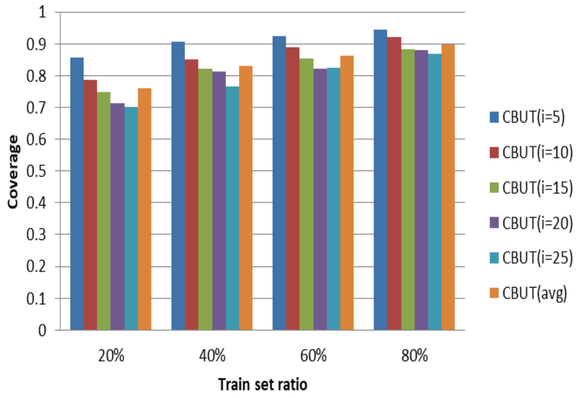


Figure 6 Effect of *i* on Coverage in different train set ratios.

the rating of a user on an unknown rating is predicted based on the ratings given by the target user 'neighbors' on the item.

(2) Context-aware recommendation using rough set model and CF (CARS-RSCF) [15]. The dependencies of items on the context attributes are measured according to attribute reduction; then, the similarities between users under the context are measured and IBCF is adopted to recommend appropriate items.

(3) Typicality-based CF recommendation (TyCo) [5]. A user is represented by a user typicality vector which could indicate the preference of user on each kind of items; then, the 'neighbors' of target user could be determined by measuring similarities between users based on their typicality degrees instead of co-rated items by users.

(4) CF hybrid filling algorithm for alleviating data sparsity (HFCF) [26]. From the project point of view, the sparse matrix is filled according to the rating information of similar items. At the same time, starting from the user's point of view, use the filled matrix to calculate the neighboring users of the target user. Select the item with the most common scores to further fill in the matrix.

(5) Mode Filling CF (MFCF) [32]. Calculate the user's average rating of the item and fill in the rating matrix.

(6) Mean value CF (MVCF) [32]. Fill the scoring matrix with the scores with the most user reviews.

Four train set ratios, i.e., 20%, 40%, 60%, and 80% were extracted from the three data sets in the experiments, and the aforementioned six algorithms were compared with CBUTCF (when the number of 'neighbors' was 20). UBCF, TyCo, HFCF, MFCF and MVCF did not consider the contextual information, and only used the user-item rating information from the three data sets. The experimental comparison results are shown in Figures 7–15.

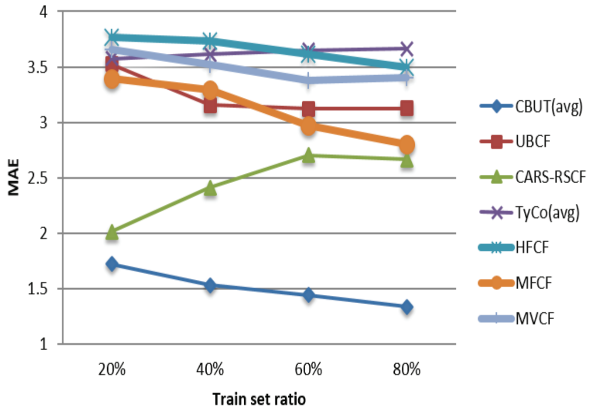


Figure 7 Comparison of MAE using LDOS-CoMoDa in different train set ratios.

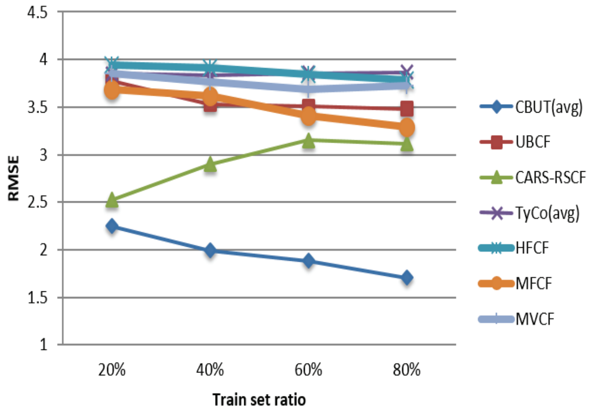


Figure 8 Comparison of RMSE using LDOS-CoMoDa in different train set ratios.

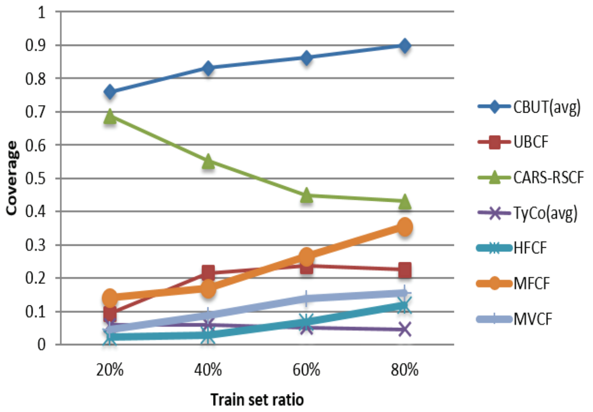


Figure 9 Comparison of Coverage using LDOS-CoMoDa in different train set ratios.

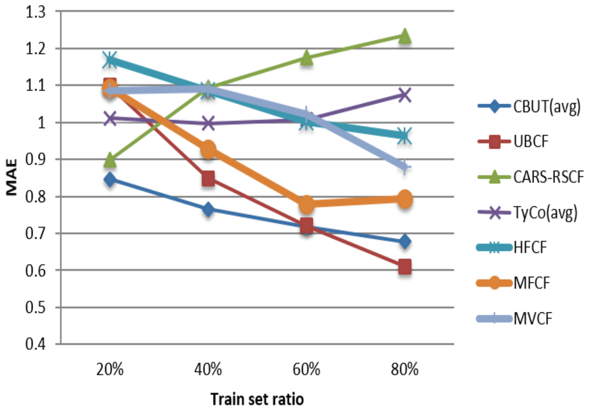


Figure 10 Comparison of MAE using C-R & C in different train set ratios.

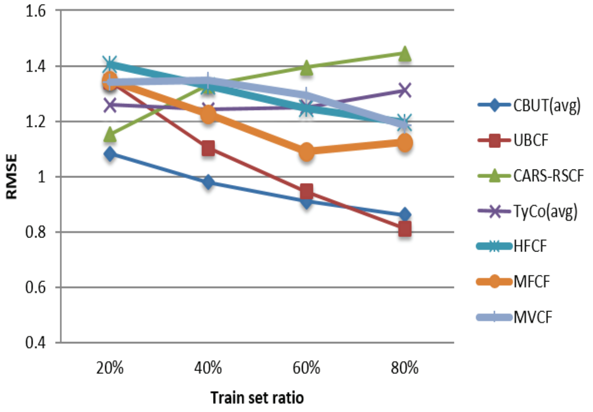


Figure 11 Comparison of RMSE using C-R & C in different train set ratios.

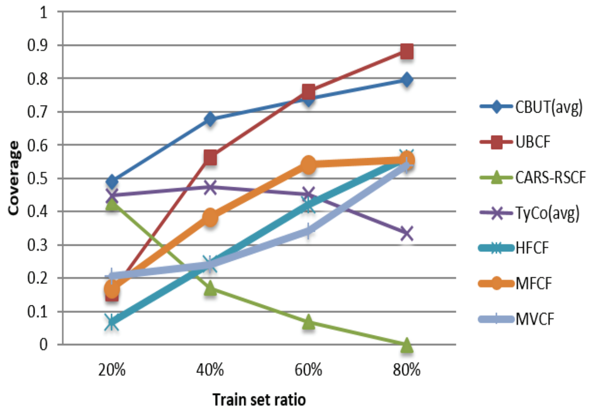


Figure 12 Comparison of Coverage using C-R & C in different train set ratios.

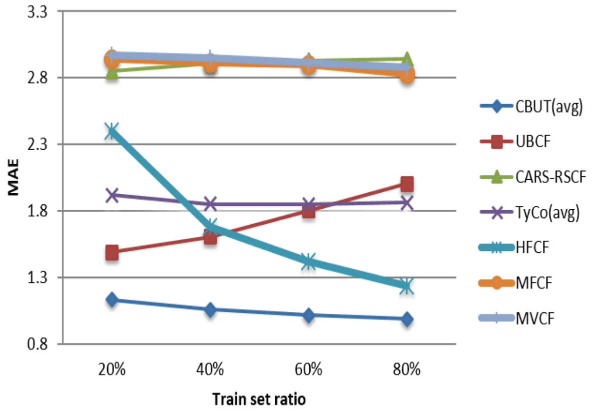


Figure 13 Comparison of MAE using C-Filmtrust in different train set ratios.

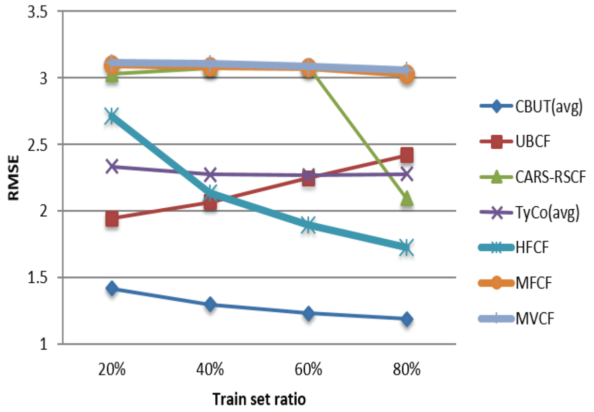


Figure 14 Comparison of RMSE using C-Filmtrust in different train set ratios.

Figures 7–15 illustrate the comparison results of MAE, RMSE, and Coverage for the seven algorithms in four train set ratios under the three data sets. First, the experimental results in Figures 7, 8, 13 and 14 validate that under the LDOS-CoMoDa and C-Filmtrust, the performance of CBUTCF is better than that of the other six algorithms in all train set ratios. Then, the experimental results in Figures 10 and 11, CBUTCF performs only slightly worse than UBCF in 80% train set ratio of the C-R & C; and better than other five algorithms in all train set ratios. This indicates that considering contextual information can effectively improve the recommendation effect, instead of solely relying on the user-item rating information. CBUTCF performs better than CARS-RSCF, which indicates that CBUTCF can effectively alleviate the impact of data sparsity problem. The results in Figures 9 and 15 validate that under

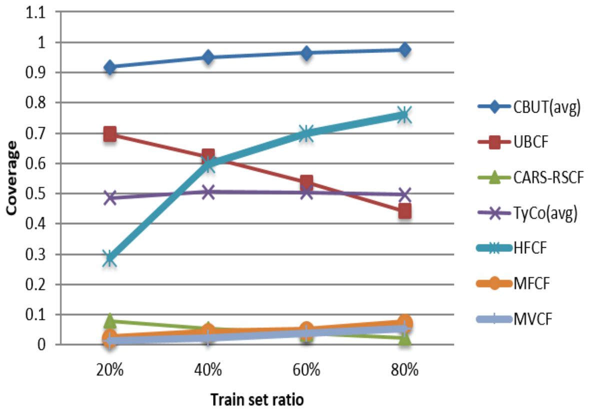


Figure 15 Comparison of Coverage using C-Filmtrust in different train set ratios.

the LDOS-CoMoDa and C-Flimtrust, the coverage of CBUTCF is superior to the other six algorithms in all train set ratios. The results in Figure 12 validate that the performance of CBUTCF is not much worse than that of UBCF in 60% and 80% train set ratios of the C-R & C; and in other train set ratios, its performance is better than other five algorithms in all train set ratios. This indicates that considering the contextual information can effectively improve the recommendation coverage. Additionally, CBUTCF is better than CARS-RSCF, which indicates that the combination of context information and user typicality can improve the coverage. The experimental results validate that CBUTCF has clear advantages in optimizing the prediction accuracy and improving item coverage, which confirms the effectiveness and better reliability of CBUTCF.

6. CONCLUSIONS

To consider the impact of contextual information on user typicality, this study proposed a context-based user typicality CF recommendation algorithm, named CBUTCF, considering the different preference of users for item types in different context environments. This algorithm combined user typicality with context to measure the preference of users within contextual information. Subsequently, the 'neighbors' of the target user can be determined based on context-based user typicality. Then, the significance of context attributes for item types can be to measure through knowledge granulation. Finally, the algorithm can predict the unknown ratings based on the context combination of the target user. The experimental results on two data sets demonstrated that CBUTCF could effectively improve the recommendation accuracy and coverage of sparse data.

For future work, we plan to incorporate users' social information and rating information to better determine 'neighbors' of target users, thereby improving recommendation performance by alleviating data sparsity problem. Additionally, it is also interesting to consider items that users hate in common, which may be different from similar preference of users.

CONFLICTS OF INTEREST

The authors declare they have no conflicts of interest.

AUTHORS' CONTRIBUTION

Jinzhen Zhang contributed in methodology, formal analysis, validation, data curation, coding, writing - original draft. Qinghua Zhang contributed in conceptualization, methodology, formal analysis, writing - review & editing. Zhihua Ai contributed in methodology, investigation and coding. Xintai Li contributed in methodology, investigation.

ACKNOWLEDGMENTS

This work was supported by the National Key Research and Development Program of China (No. 2020YFC2003502), the National Natural Science Foundation of China (No. 61876201), and the Foundation for Innovative Research Groups of Natural Science Foundation of Chongqing (No. cstc2019jcyj-cxttX0002).

ETHICAL APPROVAL

This article does not contain any studies with human participants or animals performed by any of the authors.

REFERENCES

[1] G. Adomavicius, F. Ricci, RecSys'09 workshop 3: workshop on context-aware recommender systems (CARS-2009), Proceedings of the Third ACM Conference on Recommender Systems 2009, pp. 423–424.
[2] G. Adomavicius, R. Sankaranarayanan, S. Sen, A. Tuzhilin, Incorporating contextual information in recommender systems using a multidimensional approach, ACM Transactions on Information Systems 23 (2005), 103–145.
[3] G. Adomavicius, A. Tuzhilin, Context-aware recommender systems, in: F. Ricci, L. Rokach, B. Shapira, P. Kantor (Eds.), Recommender Systems Handbook, Springer, Boston, MA, 2011, pp. 217–253.
[4] M. Braunhofer, V. Codina, F. Ricci, Switching hybrid for cold-starting context-aware recommender systems, Proceedings of the 8th ACM Conference on Recommender Systems, Foster City, Silicon Valley, California, USA, 2014, pp. 349–352.
[5] Y. Cai, H.F. Leung, Q. Li, H.Q. Min, J. Tang, J.Z. Li, Typicality-based collaborative filtering recommendation, IEEE Transactions on Knowledge & Data Engineering 26 (2014), 766–779.
[6] A. Chen, Context-aware collaborative filtering system: predicting the user's preference in the ubiquitous computing, Extended Abstracts on Human Factors in Computing Systems, Oberpfaffenhofen, Germany, 2005, pp. 1110–1111.
[7] Y.S. Chen, C.H. Cheng, D.R. Chen, C.H. Lai, A mood- and situation-based model for developing intuitive pop music recommendation systems, Expert Systems 33 (2016), 77–91.
[8] L. Chen, M. Xia, A context-aware recommendation approach based on feature selection, Applied Intelligence 51 (2021), 865–875.
[9] J.Y. Choi, H.S. Song, S.H. Kim, MCORE: a context-sensitive recommendation system for the mobile Web, Expert Systems 24 (2007), 32–46.
[10] A.K. Dey, Understanding and using context, Personal and Ubiquitous Computing 5 (2001), 4–7.
[11] B. Vargas-Govea, G. González-Serna, R. Ponce-Medellín, Effects of relevant contextual features in the performance of a restaurant recommender system, Proceedings of the 3rd Workshop on Context-Aware Recommender Systems, Chicago, USA, 2011.
[12] B. Hawashin, M. Lafi, T. Kanan, A. Mansour, An efficient hybrid similarity measure based on user interests for recommender systems, Expert Systems 37 (2020), e12471.
[13] J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems, ACM Transactions on Information Systems 22 (2004), 5–53.

[14] Z. Huang, H. Chen, D.D. Zeng, Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering, ACM Transactions on Information Systems 22 (2004), 116–142.

[15] Z.X. Huang, X.D. Lu, H.L. Duan, Context-aware recommendation using rough set model and collaborative filtering, Artificial Intelligence Review 35 (2011), 85–99.

[16] Y.G. Jing, T.R. Li, C. Luo, S.J. Horng, G.Y. Wang, Z. Yu, An incremental approach for attribute reduction based on knowledge granularity, Knowledge-Based Systems 104 (2016), 24–38.

[17] J.H. Kim, D. Lee, K.Y. Chung, Item recommendation based on context-aware model for personalized u-healthcare service, Multimedia Tools and Applications 71 (2014), 855–872.

[18] B.H. Lee, H.N. Kim, J.G. Jung, G.S. Jo, Location-based service with context data for a restaurant recommendation, in: S. Bressan, J. Küng, R. Wagner (Eds.), Database and Expert Systems Applications, Springer, Berlin, Heidelberg, 2006, pp. 430–438.

[19] J.Y. Liang, Z.Z. Shi, The information entropy, rough entropy and knowledge granulation in rough set theory, International Journal of Uncertainty, Fuzziness & Knowledge-Based Systems 12 (2004), 37–46.

[20] S.C. Li, X. Cheng, S. Su, H.N. Sun, Exploiting organizer influence and geographical preference for new event recommendation, Expert Systems 34 (2017), e12190.

[21] W.T. Li, W.H. Xu, Double-quantitative decision-theoretic rough set, Information Sciences 316 (2015), 54–67.

[22] C.H. Liou, D.R. Liu, Hybrid recommendations for mobile commerce based on mobile phone features, Expert Systems 29 (2012), 108–123.

[23] J. Macqueen, Some methods for classification and analysis of multiVariate observations, Proceedings of the 5th Conference on Berkeley Symposium Mathematical Statistics and Probability 1967, pp. 281–297.

[24] A. Odić, M. Tkalčič, J.F. Tasič, A. Košir, Predicting and detecting the relevant contextual information in a Movie-recommender system, Interacting with Computers 25 (2013), 74–90.

[25] W. Pedrycz, Granular computing: an introduction, Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference, IEEE, Vancouver, BC, Canada, 2001, pp. 1349–1354.

[26] Y.G. Ren, S.Y. Wang, Z.P. Zhang, Collaborative filtering hybrid filling algorithm for alleviating data sparsity, J. Pattern Recognition and Artificial Intelligence 32 (2020), 166–175 (in Chinese).

[27] Y.G. Ren, Y.P. Zhang, Z.P. Zhang, Collaborative filtering recommendation algorithm based on rough set rule extraction, Journal on Communications 41 (2020), 76–83 (in Chinese).

[28] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, GroupLens: an open architecture for collaborative filtering of netnews, Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, Chapel Hill, North Carolina, USA, 1994, pp. 175–186.

[29] B.M. Sarwar, G. Karypis, J.A. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, Proceedings of the 10th International Conference on World Wide Web, Hong Kong, China, 2001, pp. 285–295.

[30] M.V. Setten, S. Pokraev, J. Koolwaaij, Context-aware recommendations in the mobile tourist application COMPASS, in: P.M.E. De Bra, W. Nejdl (Eds.), Adaptive Hypermedia and Adaptive Web-Based Systems, Springer, Berlin, Heidelberg, 2004, pp. 235–244.

[31] G. Shani, A. Gunawardana, Evaluating recommendation systems, in: F. Ricci, L. Rokach, B. Shapira, P. Kantor (Eds.), Recommender Systems Handbook, Springer, Boston, MA, 2011, pp. 257–297.

[32] X.H. Sun, Research of Sparsity and Cold Start Problem in Collaborative Filtering, Ph.D. Dissertation, Hangzhou, China: Zhejiang, 2005.

[33] X.Y. Tang, Y. Xu, S. Geva, Factorization-based primary dimension modelling for multidimensional data in recommender systems, International Journal of Machine Learning and Cybernetics 10 (2019), 2209–2228.

[34] X. Wei-hua, Z. Xiao-yan, Z. Wen-xiu, Knowledge granulation, knowledge entropy and knowledge uncertainty measure in ordered information systems, Applied Soft Computing 9 (2009), 1244–1251.

[35] G.E. Yap, A.H. Tan, H.H. Pang, Discovering and exploiting causal dependencies for robust mobile context-aware recommenders, IEEE Transactions on Knowledge & Data Engineering 19 (2007), 977–992.

[36] Z. Kaihan, L. Jiye, Z. Xingwang, W. Zhiqiang, A collaborative filtering recommendation algorithm based on information of community experts, Journal of Computer Research and Development 55 (2018), 968–976.

[37] Q.H. Zhang, K. Xu, G.Y. Wang, Fuzzy equivalence relation and its multigranulation spaces, Information Sciences 346–347 (2016), 44–57.