

Article

OurPlaces: Cross-Cultural Crowdsourcing Platform for Location Recommendation Services

Luong Vuong Nguyen ¹ , Jason J. Jung ¹  and Myunggwon Hwang ^{2,*} 

¹ Department of Computer Engineering, Chung-Ang University, Seoul 156-756, Korea; vuongnguyen@cau.ac.kr (L.V.N.); jjung@cau.ac.kr (J.J.J.)

² Korea Institute of Science and Technology Information, 245 Daehak-ro, Yuseong-gu, Daejeon 136-791, Korea

* Correspondence: mgh@kisti.re.kr

Received: 14 October 2020; Accepted: 13 November 2020; Published: 27 November 2020



Abstract: This paper presents a cross-cultural crowdsourcing platform, called *OurPlaces*, where people from different cultures can share their spatial experiences. We built a three-layered architecture composed of: (i) places (locations where people have visited); (ii) cognition (how people have experienced these places); and (iii) users (those who have visited these places). Notably, cognition is represented as a pairing of two similar places from different cultures (e.g., Versailles and Gyeongbokgung in France and Korea, respectively). As a case study, we applied the *OurPlaces* platform to a cross-cultural tourism recommendation system and conducted a simulation using a dataset collected from TripAdvisor. The tourist places were classified into four types (i.e., hotels, restaurants, shopping malls, and attractions). In addition, user feedback (e.g., ratings, rankings, and reviews) from various nationalities (assumed to be equivalent to cultures) was exploited to measure the similarities between tourism places and to generate a cognition layer on the platform. To demonstrate the effectiveness of the *OurPlaces*-based system, we compared it with a Pearson correlation-based system as a baseline. The experimental results show that the proposed system outperforms the baseline by 2.5% and 4.1% in the best case in terms of MAE and RMSE, respectively.

Keywords: recommendation systems; crowdsourcing platform; cognitive similarity; similar places

1. Introduction

In recent years, the tourism industry has grown tremendously owing to the rapid development of global information, communications technology, and the widespread use of the Internet. Therefore, the process of accessing large amounts of global data from potential customers (tourists) on their points of interest, travel plans, and destinations [1–3] needs to be simplified. Given the rising demand of the tourism industry through information technology, recommendation systems have been included in tourism websites such as Expedia (<https://www.expedia.com/>), Booking (<https://www.booking.com/>), and TripAdvisor (<https://www.tripadvisor.com/>). Clearly, the most common feature of these recommendation systems (RSs) is to understand what users are thinking and expecting. Therefore, such systems used on tourism websites focus on an effective method to collect as many user preferences as possible to allow the system to provide personal recommendations to each user. In general, most RSs collect reviews, ratings, and rankings from users for each place they are aware of or have visited. Then, based on the dataset collected from active and other users, the system makes recommendations to active users by relying on the dataset of their neighbors. The described system is called a collaborative filtering

(CF) recommendation system and is widely used in many different domains beyond the tourism industry. The CF approach consists of two major approaches: item-based and user-based methods. An item-based approach uses the similarity between items to define whether users like those items. A user-based approach is to define users who have a similar pattern as an active user, and then provide the items these similar users are interested in to the active user. However, because the CF approach focuses on past interactions between users and items, for new users or items with few interactions, the CF presents some typical problems, such as cold-start, sparsity, and early rate problems [4].

To address these problems, we propose an approach that alleviates the limitations of CF, through the user-based CF approach. In this regard, we focus on understanding the user cognition pattern that helps find users similar to the active user. The cognition pattern used in this study is defined by applying the extracted features of the tourism places. This approach aims to use similar tourism places (calculated by our system based on extracted features such as their rating, ranking, and a number of reviews) to request the cognitive feedback from users. We then use this cognitive feedback to analyze and define the similarities of the users. This approach is called a cognitive similarity-based approach [5,6], in which the cognitive similarity between users can be defined based on the priority in selecting the similar tourism places of users. In particular, we deploy a crowdsourcing platform [7], called *OurPlaces* (<http://recsys.cau.ac.kr:8084/ourplaces>), which collects the cognitive feedback from users regarding similar tourism places of different countries. To maximize the benefit of collecting cognitive feedback, our system is designed to intelligently interact with the users, i.e., the users are provided the simplest and fastest process for selecting similar places. In addition, to determine the cognitive similarity between users, we propose a three-layered architecture. The structure of this architecture is comprised of three superposed networks, which are strongly linked: (i) user-relating network based on explicit information extracted from the cognition network; (ii) cognition network for relating the cognitive similarities among users based on their similar interest in tourism place; and (iii) a place network consisting of the relevant places based on the measured similarities among the extracted features. We applied the proposed platform to a cross-cultural recommendation system as a case study and demonstrated its performance experimentally in term of MAE and RMSE. The main contributions of this paper are as follows:

- (a) We propose the *OurPlaces* crowdsourcing platform to collect cognitive feedback from users regarding similar tourism places.
- (b) We built the three-layered architecture to extract the cognitive similarities among users.
- (c) We deployed a recommendation system based on the cognitive similarities among users (*k*-nearest neighbors) extracted from the three-layered architecture.

The remainder of this paper is organized as follows. Section 2 discusses the background and previous related studies on crowdsourcing platforms. In Section 3, we present the cognitive similarity measurements and propose a three-layer architecture to extract cognitive similarities. In Section 4, we present a case study of a cross-cultural recommendation system. In Section 5, we describe experiences of using the *OurPlaces* platform, including an overview, the available functions, and a data analysis. Finally, in Section 6, we provide some concluding remarks and present areas of future study.

2. Related Work

The most popular approach to building recommendation systems is based on the CF method [4,8,9]. CF has been commonly used as a sort of personalized recommendation technique in several domains [10]. In general, CF models are based on collecting and calculating large amounts of information on user's historic behavior, activities, or preferences and predicting what users prefer on the basis of their similarity to other users (neighborhood). The critical advantage of CF is that it does not rely on machine analyzable

content. Therefore, it can accurately recommend complex items without any understanding of the items themselves [4,11]. However, CF still has a few limitations that significantly reduce the user experience such as cold start problems, data sparsity, and scalability.

User-based CF is one of the most commonly used recommendation approaches [9,12]. It recommends items according to the opinions of users who have similarities with an active user. The critical point of user-based CF is to measure the similarities among users. In traditional user-based CF, to determine the similarities between two users, items rated by both users are typically used. The prediction for active user depends on the ratings for an item given by their neighboring users, as shown in Figure 1. For instance, KARS [13] is a user-based CF technique that forms the recommendations based on user preferences. This system analyzes the reviews of previous users and stores them in the database. It then extracts the keywords from these reviews which serve as a factor for recommendations given to new users. In [14], to reduce the overall prediction error, a combination of user-based and item-based methods, which combine item-based and user-based predictions through multiple linear regression and support vector regression, was proposed. Recent research has considered alleviating data sparsity by merging user-based and item-based collaborative filtering [15]. For this purpose, a bi-clustering technique is leveraged that helps cluster all users/items into several groups (reducing the dimensionality) and then measure the user-item similarities based on the similarity of these groups. Many studies applying a user-based approach or a combination of user-based and item-based collaborative filtering has achieved high performance in terms of user recommendations. However, these approaches still have problems because they rely on the user ratings. This means that, when users only rate a few common items, the similarities between users may be inaccurate. The results lead to false recommendations by the system. To address this problem, our approach focuses on finding the neighbor of an active user by considering the cognitive similarities between users instead of their items ratings. The key point of this approach is to collect cognitive feedback from users regarding similar places. Therefore, we deploy a crowdsourcing platform that collect the cognitive feedback of users to analyze and determine the cognitive similarity between them.

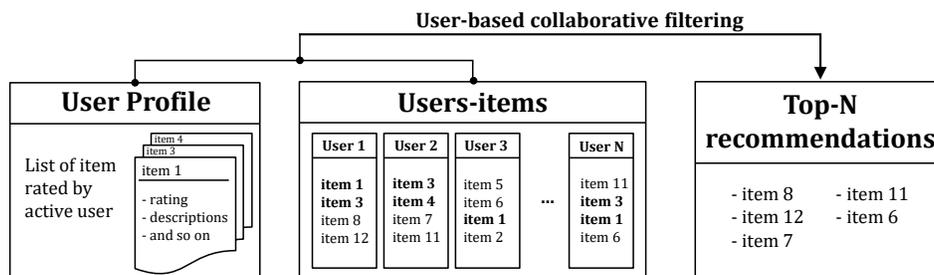


Figure 1. User-based collaborative filtering.

Crowdsourcing is defined as a strong framework for the delivery of knowledge, and has been applied under several different names, including collective intelligence, and social computation or human computation [16]. In particular, the requester separates the main task into several small tasks and post to the crowd so that the workers accomplish questions for intrinsic or extrinsic reasons [17]. Managing and analyzing data on crowdsourcing platforms has recently become widespread, leading to an upsurge of research activities [18,19]. Therefore, many crowdsourcing platforms have been proposed, including CloudCrowd (used to write and edit projects) and ResearchGate (allowing researchers to connect and share their research). One of the most popular ones is Amazon Mechanical Turk (MTurk), which is a crowdsourcing platform that allows individuals or business corporations (called requesters) to share several tasks such as clustering, labeling, and creative design. Workers (called Turkers) choose human

intelligence tasks and earn a monetary incentive. In [20], the authors provided an overview of MTurk as an academic research platform and guidelines for data collection. Facebook is another popular platform that can be used for crowdsourcing. Compared to Twitter, Facebook has more information sources, including blogs and photographs. Therefore, Facebook can perform some sophisticated tasks such as character analysis, financial analysis [21], activity planning [22], and product repository generation [23]. In addition, Facebook builds various applications for individuals to design their own crowdsourcing tasks. In [5], the authors proposed a crowdsourcing platform for collecting the cognitive feedback from users and deployed a movie recommendation system as a case study.

This study focused on deploying a crowdsourcing platform (called *OurPlaces*), which provides the tasks to users, including feedback regarding the similarity between tourism place and request confirmation from active users after the system recommends other similar places. Based on the cognitive user feedback collected, we conducted an analysis and defined the cognition pattern of users in selecting pairs of similar tourism places. Then, by applying a three-layered architecture network, we determined and extracted the cognitive similarity between users. Based on CF, we could define the k -nearest neighbors of an active user based on the cognitive similarities between both users, allowing our platform to form suitable recommendations to the active user. In this study, we deployed a cross-cultural recommendation system based on the cognitive similarity between users instead of traditional CF.

3. Crowdsourcing Platform for Measuring Cognitive Similarity

3.1. Cognitive Similarity Measurements

As the basic idea of calculating the similarity between two tourism places p_i and p_j , the tourism place based on the extracted features related to both places is first presented. In the next step, we apply a calculation to determine the similarity of each extracted feature from these tourism places. Finally, we combine all similarity scores between the extracted features and obtain a similarity score between two tourism places. In this paper, the soft cosine similarity is the metric for measuring the similarity between extracted features. The cosine similarity is defined as the cosine of the angle between two non-zero vectors of an inner product space. Applying the suggestion function in *OurPlaces*, when the users select known tourism place p_i , the system measures the similarity between p_i and all tourism places p_j remaining in the database. Then, the four tourism places that have the highest similarity with p_i are suggested for the users, and the system requests their feedback regarding these similar places. The formulation for scoring the similarity between two tourism places p_i and p_j is described as follows:

$$Sim(p_i, p_j) \equiv \langle R_{ij}, K_{ij}, N_{ij} \rangle, \quad (1)$$

where R_{ij} , K_{ij} , and N_{ij} represent the similarity measurement of feature ranking, rating, and number of reviews between tourism places p_i and p_j , respectively. The similarity measurement of each feature between p_i and p_j is calculated using the cosine similarity. In particular, considering the rating feature, when using the cosine similarity, the similarity measurement of the rating features between tourism places p_i and p_j is described as follows:

$$Sim_R(p_i, p_j) = \frac{\sum_{h=1}^n R_{i,h} R_{j,h}}{\sqrt{\sum_{h=1}^n (R_{i,h})^2} \sqrt{\sum_{h=1}^n (R_{j,h})^2}}. \quad (2)$$

We repeat Equation (2) with the remaining features such as the ranking (K_{ij}) and the number of reviews (N_{ij}). Finally, we obtain the similarity scores between two tourism places according to Equation (1). We loop Equation (1) for all the remaining tourism places in the system and obtain a set $\{Sim(p_i, p_{j+h}) | h \in$

$[1, \dots, n]$, where n is number of the tourism places in the database excluding the tourism places p_i and p_j . In addition, we consider the convenience of interacting with users. Therefore, we aim to achieve a strong performance in the first step of cognitive user feedback collection process, as described in Section 1. To achieve this, we determine the α parameter, which is the priority of users in selecting similar tourism places. The priority of users in selecting similar places is calculated and extracted from a cognition network of the three-layered architecture and is dynamically updated according to the user activities. In this way, we can display the tourism places in the first step of the cognitive feedback collection process close to the expectations of the users. The similarity measurement between tourism places is also re-calculated based on the priority in selecting similar places for users. This means that the formulation for calculating the similarity between tourism places changes dynamically according to the α parameter (the personal change according to the cognition pattern of the users). Equation (1) is therefore applied according to each user and is rewritten as follows:

$$Sim(p_i, p_j) = \frac{\sum_{k=1}^n \alpha_k \times Sim_k(p_i, p_j)}{\sum_{k=1}^n \alpha_k}, \quad (3)$$

where α denotes the priority of the user in selecting pairs of similar tourism places. k denotes the number of features extracted from a place, i.e., the rating, ranking, and number of reviews. $Sim_k(p_i, p_j)$ is a similarity measuring between places p_i and p_j , as described in Equation (1).

Following the cognitive feedback collection process, we obtain a dataset of similar tourism places from each user. The key point in these datasets is the priority in selecting pairs of similar tourism places for each user (which we call the cognition pattern of a user) and updating them according to the activities of each user. To determine and update this cognition pattern, we recalculate the similarity between the tourism places of each user after all user activities. We can then determine the similarity between users based on the similarity of these user cognition patterns. The cognitive similarity between user u and user v is defined as follows:

Definition 1. *The cognitive similarities between users u and v are the similarities among the priorities of these users in selecting similar tourism places (p_i, p_j) . The formulation for measuring such similarities is as follows:*

$$CS_{u,v} = \frac{\sum_{i,j}^N \rho_{u_i} \rho_{v_j}}{\sqrt{\sum_{i,j}^N \rho_{u_i, u_j}} \sqrt{\sum_{i,j}^N \rho_{v_i, v_j}}}, \quad (4)$$

where ρ_u and ρ_v represent the priorities of users u and v , respectively, to select similar tourism places.

For instance, coefficients i and k are the number of tourism places and number of features extracted from each tourism place, respectively. Then, each tourism place is represented as a vector $p_i = \{F_n | n \in [1, \dots, k]\}$. When user u has a new activity (a pair of similar tourism places p_i and p_j), the cognition pattern of user u in selecting similar tourism places is represented as $\rho_u = \{Sim_{F_n} | n \in [1, \dots, k]\}$, where Sim_{F_n} is the cosine similarity between features F_n of tourism places p_i and p_j . This cognition pattern of users is enriched by each of their activities (dynamic updates). This means the cognitive similarity between user u and other users is also a dynamic update according to their activities. This demonstrates that the proposed method is suitable for building the personalized recommendations.

3.2. Three-Layered Architecture for Cognitive Similarity

In this section, we introduce the three-layered architecture to extract the cognitive similarity between users, which includes: (i) a place network \mathbb{P} ; (ii) a cognition network \mathbb{C} ; and (iii) a user network \mathbb{U} .

Herein, we consider networks with several different relations between individuals. Therefore, a network is characterized as a set of objects (or nodes) and a set of relations. The definitions of a network and a distance network are presented below.

Definition 2 (Network). A network $\langle N, E^1, \dots, E^n \rangle$ is made of a set N of nodes and n sets of objects pairs $E^i \subseteq N \times N$ the set of relations between these nodes.

Definition 3 (Distance network). A distance network $\langle N, E^1, \dots, E^n \rangle$ is made of a set N of nodes and n sets of distance functions $E^i : N \times N \rightarrow [0,1]$ defining the distance between nodes (so satisfying symmetry, positiveness, minimality, and triangular inequality).

To uncover those cognitive similarities among individuals that can be found based on their cognition pattern, we propose a three-layered architecture for constructing the user network. Our aim is to determine the cognitive similarity between users such that they define the k -nearest neighbors. The cognitive similarity between users is defined as their priority in selecting a pair of similar tourism places. Based on the cognitive similarity between users, we classify and determine the k -nearest neighbors of active users. Our architecture goes from left to right in Figure 2. The networks are considered to have several different relations between individuals. This means that each network is characterized as a set of relations and a set of objects (nodes). The characteristics of each layer and the relationships between layers in the three-layered architecture are described below.

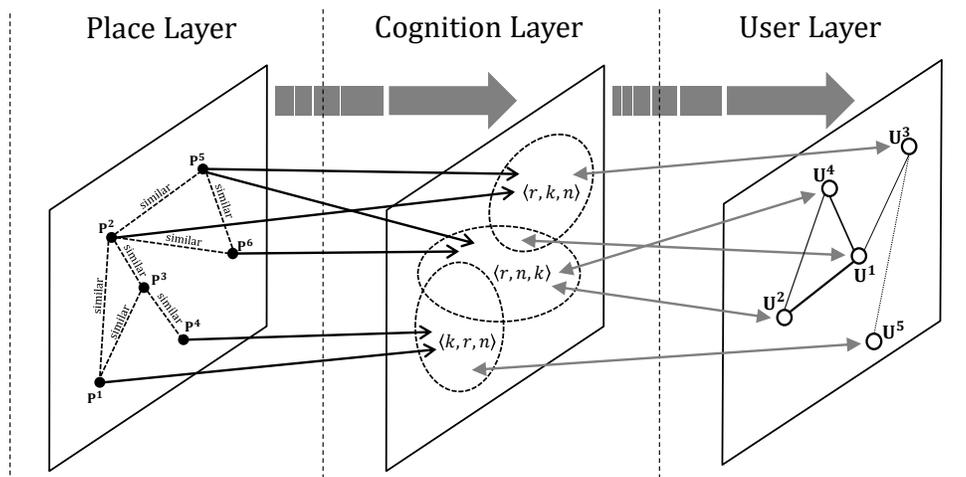


Figure 2. Three-layered architecture for cognitive similarities.

- **Place Layer:** In the place network \mathbb{P} of place layer, the nodes and edges represent tourist places and their relations, respectively. The relations between the nodes are the similarity between tourist places. A place network \mathbb{P} is a directed graph $\langle N_{\mathbb{P}}, E_{\mathbb{P}}^{similarity} \rangle$, where $N_{\mathbb{P}}$ is the set of tourist places and $E_{\mathbb{P}}^{similarity} \subseteq N_{\mathbb{P}} \times N_{\mathbb{P}}$ is the set of relations between these tourist places. In this study, the relationship between tourism places was measured by using cosine similarity metric following Equation (1).
- **Cognition Layer:** In this layer, the cognition network \mathbb{C} is determined as a network $\langle N_{\mathbb{C}}, E_{\mathbb{C}}^i \rangle$ where $N_{\mathbb{C}}$ and $E_{\mathbb{C}}^i \subseteq N_{\mathbb{C}} \times N_{\mathbb{C}}$ are the set of the cognition pattern from groups of users and the relationship between these groups, respectively. These groups are determined and classified based on the cognition pattern of each user as mentioned in Section 3.1. The objective relationship between the \mathbb{S} and \mathbb{O}

defined through the priority in selecting similar pairs of tourism places by users. These relationship are expressed by a relation: $Selections \subseteq N_S \times N_C$.

- **User Layer:** In this layer, the user network \mathbb{U} consists of nodes and relations, which are users and numerous kinds of relationships, respectively. Therefore, the user network \mathbb{U} is defined as a network $\langle N_{\mathbb{U}}, E_{\mathbb{U}}^i \rangle$, in which $N_{\mathbb{U}}$ is a set of entity of a user and $E_{\mathbb{U}}^i \subseteq N_{\mathbb{U}} \times N_{\mathbb{U}}$ is the relationship between these entities. These relations are extracted through the objective relationship from \mathbb{C} to \mathbb{U} based on the extraction of user groups who have cognition pattern similarity. They can be expressed by a relation: $Extracts \subseteq N_C \times N_{\mathbb{U}}$.

4. Cross-Cultural Recommendation: A Case Study

In this section, we present a cross-cultural recommendation system and focus on tourism places in different countries including hotels, restaurants, shopping malls, and attractions. Notably, our system follows the CF approach and is designed as a personal recommendation system [24] based on the cognitive similarities among users. The cognition pattern of users regarding the similarity of the items is defined as the priority of the users in selecting similar tourism places, and the measurements used to calculate the cognitive similarities among users are described in Section 3.1. To develop the cognitive similarities between users, the similarities of the tourism places need to be determined during the preprocessing. Then, based on the previous similar tourism places that were browsed and selected, cognitive similarities between users occur. In general, following the basis of the CF principle, the process of recommending tourism places can be split into the following three steps:

- Representing the information based on the history of user activities: The priority in selecting similar tourism places of users needs to be analyzed and modeled (user cognition pattern).
- Generating the neighbors of the active user: The cognitive similarities between users can be extracted from the three-layer architecture according to the collected datasets from users and the collaborative filtering algorithm.
- Generating tourism place recommendations: The top-N tourism places are recommended to the active user according to the activity history of the neighbors.

According to these steps, all activities of every user in our database were used to calculate and construct the list of neighbors and were stored in the corresponding records in our database. When an active user signs in, the system shows the recommendations based on the neighbors have a high similarity to the active user. In addition, every response from an active user regarding a similar tourism places is used to update the cognitive similarities among users. This means that the cognitive similarities among users are dynamically updated according to all user activities (cognitive feedback on the similarities of tourism places). Figure 3 presents the recommendation process for our cross-cultural recommendation system.

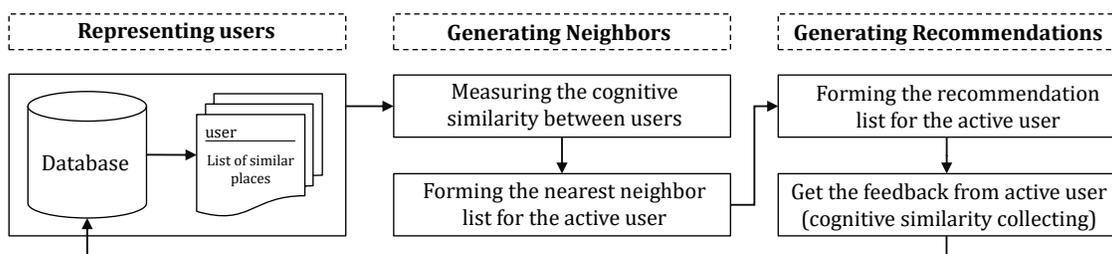


Figure 3. Recommendation process.

4.1. Users Representation

Our dataset consists of four types of tourism places: hotels, restaurants, shopping malls, and attractions. Therefore, we decided to use several features that overlap in these four types of places: the ratings, the rankings in an area, and the number of reviews. These features are considered to represent tourism places so that we can measure the similarities among tourism places. When an active user selects a pair of similar tourism places, the system calculates the similarities among features extracted from the two tourism places and combines them with the cognition pattern stored in our database of active users to obtain a set of features representing the current user cognition pattern. The set of features has three scores ordered from highest to smallest: the ratings, rankings, and number of reviews. The cognition pattern calculation is repeated and continuously updated when an active user has any new activities. For instance, users U_i have a list of tourism places with similarity $\{(P_a, P_b) | a, b \in [\text{list of tourism places in database}]\}$. The cognition pattern of the users is described as an ordered vector:

$$U_i = \{R_{a,b}^i; K_{a,b}^i; N_{a,b}^i\}, \quad (5)$$

where $R_{a,b}^i$, $K_{a,b}^i$, and $N_{a,b}^i$ represent the similarities of the feature ratings, rankings, and numbers of reviews, respectively. The ordering of these features depends on the similarity scores in comparing each pair of extracted features from the tourism places P_a and P_b . The similarity score of each feature is calculated according to Equation (2).

4.2. Generation of Neighbors

Our purpose is to automatically classify users into groups and determine the k-nearest neighbors of active users. To do so, we first represent each user based on their activities, as mentioned above. For instance, as shown in Table 1, the ordered vectors $\langle K = 2; R = 1.97; N = 1.83 \rangle$, $\langle K = 2.13; R = 2.12; N = 2.06 \rangle$, $\langle R = 1.38; K = 1.36; N = 1.1 \rangle$, $\langle K = 2.13; R = 2.12; N = 2.06 \rangle$, and $\langle K = 1.93; N = 1.82; R = 1.76 \rangle$ represent users U_1, U_2, U_3, U_4 , and U_5 , respectively. We then compared the similarities of every user in each group to obtain the k-nearest neighbors. The ordering of features is dynamically updated when the active user has new activities. This means that the k-nearest neighbors of the active users are also dynamically updated.

Table 1. The activities history of the users.

Users	Similar Tourism Place			
	$\langle P_1, P_2 \rangle$	$\langle P_3, P_4 \rangle$	$\langle P_5, P_6 \rangle$	$\langle P_7, P_8 \rangle$
U_1	R = 0.87; K = 0.78; N = 0.67	R = 0.59; K = 0.64; N = 0.73	0	R = 0.51; K = 0.58; N = 0.43
U_2	R = 0.87; K = 0.78; N = 0.67	R = 0.59; K = 0.64; N = 0.73	R = 0.66; K = 0.71; N = 0.66	0
U_3	R = 0.87; K = 0.78; N = 0.67	0	0	R = 0.51; K = 0.58; N = 0.43
U_4	R = 0.87; K = 0.78; N = 0.67	R = 0.59; K = 0.64; N = 0.73	R = 0.66; K = 0.71; N = 0.66	0
U_5	0	R = 0.59; K = 0.64; N = 0.73	R = 0.66; K = 0.71; N = 0.66	R = 0.51; K = 0.58; N = 0.43

The relation between users is determined based on the priority in selecting the similar tourism places of each user (cognition pattern of each user). In this paper, we call this relation the cognitive similarity among users. Therefore, as mentioned in Section 3.1, the cognitive similarities between users were calculated using Equation (4). As shown in Table 1, the priority in selecting similar tourism places of user U_1 is $\langle K = 2; R = 1.97; N = 1.83 \rangle$. Then, after combining all scores of such similar tourism places, the priorities of user U_2 and U_4 in selecting similar tourism places is $\langle K = 2.13; R = 2.12; N = 2.06 \rangle$, which is similar to user U_1 . Therefore, the k-nearest neighbors of U_1 are U_2 and U_4 .

4.3. Generation of Recommendations

The activity histories of the neighbors are used to compute the recommendations. Following the above calculation, we can define the neighbors of user U_1 as U_2 and U_4 . Therefore, we list all activity histories of users U_1 , U_2 , and U_4 including all similar tourism places of these users so that we can determine the most popular tourism places. As listed in Table 1, we can see that the maximal number of similar tourism places is $\langle P_1, P_2 \rangle$ and $\langle P_3, P_4 \rangle$. As a result, tourism places P_1 , P_2 , P_3 , and P_4 should be presented to user U_1 as recommendations. Because the priority in selecting similar tourism places of user U_1 is updated when there are any new activities, the neighbors of user U_1 are also updated, which means that the tourism places that are represented as recommendations dynamically change according to the cognitive similarities between user U_1 and the other users. Thus, the system can predict the most suitable tourism places for the active user and address not only the cold-start problem but also the sparsity of the dataset in the collaborative filtering recommendation system.

5. Experiments

5.1. Overview of OurPlaces Crowdsourcing Platform

To achieve a simple and user-friendly interface, we focus on two important points: *the characteristics of the graphical and web user interfaces* and *the user interface design process* [25]. Besides, we determined that flexibility is also an important feature. Therefore, in the *OurPlaces* platform, users can directly use the function provided instead of following a predefined sequence. In addition, the platform not only provides roll-back methods, but also allows the user to stop during a sudden situation (e.g., solving other problems). Users can go back to their work easily because the *OurPlaces* system has been designed very simply and clearly. The platform aims to clearly provide access to users with different skill levels. This means that even users with no skill or experience can still use our system.

Notably, *OurPlaces* is a crowdsourcing platform based on the web. Therefore, we have to address several problems such as the latency response, instructions given to perform the operation for the entire system, and queries with large data. To solve these problems, we follow the concept: “*showing the user what they need when they need it, and where they want it*”. The platform has descriptions of specific functions for all users during their activities and allows users to quickly access all of the pages. Besides, to avoid querying large data, we apply a pagination method to represent pages during long processing. We believe that pagination helps users interact with the platform more simply and clearly. Especially, we followed the golden rules in designing the user interface [26]. In particular, we aimed to ensure the consistency by applying one template for all pages. Therefore, users can easily recognize similar signs (e.g., buttons and functions) while operating the *OurPlaces* platform. In addition, we applied this idea to all action processes occurring in the system. This means that, with all of the different actions, the system guaranteed the same interfacial behavior.

The *OurPlaces* crowdsourcing platform is built based on Java and contains two services: web service and background service. On the web service side, we use Apache Tomcat and Java Server Pages. We decided to use the MySQL database on the background service side to store all datasets collected because it has high reliability and security and on-demand scalability. To handle multiple accesses and tasks with large numbers of data, we decided to use the model-view-controller model for our system. The architecture of the system is shown in Figure 4. The key features of the *OurPlaces* platform are data collection, processing of collecting feedback from users, and analyzing data to extract the cognitive similarity between users. Specifically, there are two methods of collecting data in our platform. The first is to automatically collect tourism places information from TripAdvisor and the second is added by users for

the initial dataset. Then, the *OurPlaces* platform uses these datasets of tourism places to perform cognitive feedback collection from users regarding their similarities.

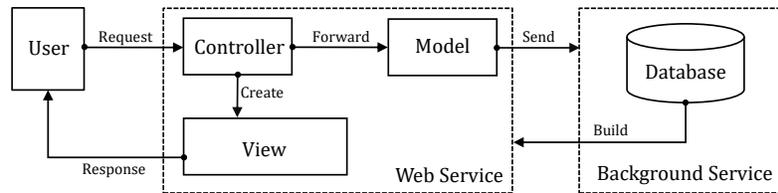


Figure 4. Architecture of *OurPlaces* crowdsourcing platform.

The process of selecting the similar tourism places from users is described in Figure 5, where the users, in turn, select tourism place that they know or have visited. Then, based on the four tourism places suggested by *OurPlaces*, users continue to choose similar tourism places as those previously selected. The *OurPlaces* automatically stores all user activities as well as their known tourism places similarities of such tourism places. Relying on this collected dataset, *OurPlaces* can define the cognition pattern of users and dynamically update it for each user depending on their activities. Thus, the cognitive similarities between users also dynamically change according to the user activities. For example, when an active user performs an activity (e.g., selects the tourism place they know or have similar tourism place pairs), the system automatically re-calculates the cognitive similarity between the active user with others and dynamically updates to each user. Therefore, *OurPlaces* obtains new knowledge and enriches the cognitive similarity between users so that it can provide more suitable personal recommendations. In summary, the process of cognitive feedback collection consists of three steps:

- (a) During the first step, the *OurPlaces* platform displays the four types of tourism places (hotels, restaurants, shopping malls, and attractions) based on the country information, and new users then select the tourism places that they are aware of and/or have been to.
- (b) During the second step, the platform suggests a list of four similar tourism places (dependent on the type of place and accompanied by an α that is initially set to 1 for new users parameter, which is the selecting trend of the user). The user then selects a tourism place they think is similar to the one they have chosen during the first step.
- (c) During the final step, the platform stores cognitive feedback from the users in the database and re-calculates the α parameter for the suggestion process when the user has a new selection (start a new loop).

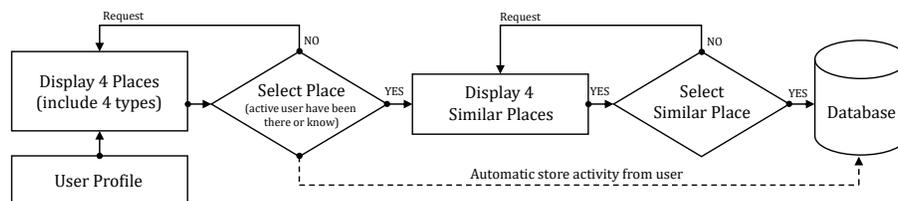


Figure 5. The process of collecting feedback from users in *OurPlaces* crowdsourcing platform.

The selection of similar tourism places from users becomes a loop with variable α parameters dynamically closer to the priority of users in selecting a pair of similar tourism places. Each loop finishes with the results is a pair of similar tourism places of the active user stored in the database. Based on this collected dataset, we conduct an analysis and define the cognition pattern of each user in selecting similar tourism places.

5.2. Statistics of Cognitive Feedbacks

There are many sources to gather information about various places and tourism data are provided online. We decided to collect data from TripAdvisor because it is quite suitable and has sufficient information for our purpose. We collected 18,286 popular places from 24 countries, including of 84 cities. The places are categorized into four types: hotels, restaurants, shopping malls, and attractions. Each place includes such features as the name, descriptions, rating, ranking, number of reviews, and overall price. However, the overlapping features extracted between tourism places include the rating, ranking, and number of reviews. Therefore, we used these overlapping features to calculate the similarity between tourism places in constructing the initial dataset for our platform. Detailed statistics of the tourist place data in the *OurPlaces* crowdsourcing platform are described in Table 2. The *OurPlaces* platform is now online and is continuing to collect cognitive feedback from users for the collected tourism places. At the time of writing, we have around 50 active users and approximately 2000 pieces of feedback from users regarding similar tourism places.

Table 2. The statistics of tourist places data in our *OurPlaces* platform.

#	Country	(#) Cities	(#) Places
1	South Korea	(7) Seoul; Busan; Daegu; Deajeon; Geoje; Incheon; Ulsan	1306
2	Vietnam	(3) Hanoi; Ho Chi Minh; Danang	670
3	Singapore	(1) Singapore	240
4	Thailand	(4) Bangkok; Phuket; Chiang Mai; Pattaya	750
5	India	(2) New Delhi; Mumbai	334
6	Japan	(4) Tokyo; Osaka; Kyoto; Fukuoka	889
7	China	(4) Beijing; Shanghai; Hong Kong; Macau	1012
8	England	(5) London; Manchester; Cambridge; Liverpool; Birmingham	1211
9	Spain	(5) Barcelona; Madrid; Seville; Malaga; Granada	1128
10	Germany	(3) Berlin; Munich; Hamburg	875
11	France	(3) Paris; Nice; Lyon	768
12	Greece	(2) Santorini; Athens	305
13	Austria	(2) Vienna; Langenfeld	264
14	Italy	(4) Rome; Milan; Venice; Turin	657
15	United States	(6) New York; Los Angeles; Miami; Chicago; Washington; San Francisco	2083
16	Brazil	(2) Rio de Janeiro; Sao Paulo	330
17	Argentina	(3) San Carlos de Bariloche; Pinamar; Buenos Aries	453
18	Colombia	(3) Bogota; Pereira; Salento	450
19	Uruguay	(2) Montevideo; La Paloma	321
20	Chile	(2) Santiago; Punta Arenas	304
21	Mexico	(2) Mexico City; Oaxaca	332
22	Canada	(4) Vancouver; Toronto; Montreal; Quebec City	750
23	Australia	(5) Melbourne; Sydney; Brisbane; Newcastle; Adelaide	1200
24	New Zealand	(6) Auckland; Queenstown; Wellington; Paihia; Wanaka; Hamilton	1654
Total		84 cities	18,286

The collected data from users have the following format: $(U_i, p_j, p_k, CS_{p_j, p_k}^{U_i}, \gamma_i)$ where U_i is the id of the user and p_j and p_k are a pair of the similar places. $CS_{p_j, p_k}^{U_i}$ is a vector representing the cognitive similarity of user U_i in selecting similar places p_j and p_k . γ_i is the number of times the user change a suggested tourism place when selecting a pair of similar tourism places. The collected data from users were used to conduct our experiment.

5.3. Evaluation

To evaluate the proposed approach, we conducted five-fold cross-validation. In particular, our dataset was separated into five mutually exclusive folds, in which 80% of similar tourism places were used used

for training and the remaining 20% were removed for testing during each fold. The experiments were conducted for each fold and predicted the similar tourism places of the test set based on the tourism places of the training set. The performance of each fold was evaluated by comparing the predictions with the real value available for the test set. Finally, the result is the average across predicted tourism places. Several criteria can be used to evaluate the accuracy of the CF such as *Recall*, *Precision*, *F-measure*, Mean Absolute Error (MAE), Mean Square Error, and Root Mean Square Error (RMSE). We decided to use MAE and RMSE as the evaluation metrics for the comparative analysis. The calculation of MAE and RSME is described as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - y_i^p| \quad \text{and} \quad RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y_i^p)^2}, \quad (6)$$

where n denotes the number of tourism similarity of the test set. y_i and y_i^p denote the real and predicted values in the test set, respectively. The range of the MAE result is from 0 to infinity, where infinity is the maximum error according to the values scale of the measured.

To create the test set, we divided the items of each user according to the k number of sections/folds, where each fold was used as a test set. We set k to 5 because estimates of the yield test error rates do not suffer from an excessive bias or high variance. Specifically, the dataset was split into five folds. In the first interaction, the first fold is the test set and the remaining four folds are the training set. During the next interaction, the second fold is the test set, and this process is repeated until every fold has been used as the test set. We used the conventional *User-based Pearson Similarity* (UBPS) approach as the baseline for the comparative analysis of our proposed approach. We chose the neighborhood sizes of {5, 10, 20, 30, 50} during the experiments because the baseline is reported to achieve the best performance for approximately 50 neighbors of the active user. The results of the comparison between the proposed method and the baseline are shown in Table 3.

Table 3. Comparison between UBPS and proposed method with varying neighborhood sizes.

Number of Neighbors	MAE		RMSE	
	UBPS	Proposed Method	UBPS	Proposed Method
5	0.809	0.784 (+2.5%)	1.203	1.162 (+4.1%)
10	0.782	0.761 (+2.1%)	1.145	1.108 (+3.7%)
20	0.774	0.755 (+1.9%)	0.997	0.964 (+3.3%)
30	0.771	0.757 (+1.4%)	0.854	0.823 (+3.1%)
50	0.757	0.746 (+1.1%)	0.791	0.769 (+2.2%)

6. Conclusions and Future Work

In this paper, we present a novel approach for measuring the cognitive similarity between users to improve user-based CF. For this purpose, we propose a crowdsourcing platform (called *OurPlaces*) to request cognitive feedback from users regarding similar tourism places. This platform was designed to have good performance in terms of user interaction. In particular, the system applies the simplest process for collecting cognitive feedback from users. Such cognitive feedback from users is stored in our database to enhance the cognition pattern of users in selecting similarities in tourism places. We then measured the cognitive similarities among users. Then, based on cognitive similarities, our system determine the k -nearest neighbors of the active user by applying a three-layered architecture. According to this approach, we deployed a case study on a cross-cultural recommendation using the dataset collected from the *OurPlaces* crowdsourcing platform. Rather than the scalability testing of the system, we decided to evaluate the precision of the recommendation. To demonstrate the performance of our approach in improving the user-based CF, we designated the user-based Pearson correlation similarity as the baseline

for comparison with our method during the evaluation. The results demonstrate that the proposed method outperforms the baseline and achieves a 2.5% improvement in the best case with the MAE metric. In terms of the RMSE evaluation, the proposed method achieves a 4.1% improvement over the baseline in the best case.

The *OurPlaces* platform is now online and is continuing to collect more cognitive feedback on similar places from users. To date, the dataset has collected data from 50 users and consists of approximately 2000 pieces of cognitive feedback from these users. As a future study, we plan to have at least 5000 users and 20,000 responses to perform more accurate analyses of user cognition in choosing places similarities. Moreover, we will conduct an evaluation by comparing the proposed method with other similar approaches.

Author Contributions: Conceptualization, Luong Vuong Nguyen, Jason J. Jung, and Myunggwon Hwang; methodology, Luong Vuong Nguyen and Jason J. Jung; software, Luong Vuong Nguyen; validation, Jason J. Jung and Myunggwon Hwang; formal analysis, Luong Vuong Nguyen and Jason J. Jung; investigation, Jason J. Jung; resources, Luong Vuong Nguyen; data curation, Luong Vuong Nguyen, Jason J. Jung, and Myunggwon Hwang; writing—original draft preparation, Luong Vuong Nguyen and Jason J. Jung; writing—review and editing, Luong Vuong Nguyen, Jason J. Jung, and Myunggwon Hwang; visualization, Luong Vuong Nguyen; supervision, Jason J. Jung; project administration, Jason J. Jung; and funding acquisition, Jason J. Jung and Myunggwon Hwang. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Korea Institute of Science and Technology Information (KISTI). Also, this work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2018K1A3A1A09078981, NRF-2020R1A2B5B01002207).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sebastia, L.; Garcia, I.; Onaindia, E.; Guzman, C. e-Tourism: A Tourist Recommendation and Planning Application. *Int. J. Artif. Intell. Tools* **2009**, *18*, 717–738. [[CrossRef](#)]
2. Loh, S.; Lorenzi, F.; Salda na, R.; Liethnow, D. A Tourism Recommender System Based on Collaboration and Text Analysis. *Inf. Technol. Tour.* **2003**, *6*, 157–165. [[CrossRef](#)]
3. Borràs, J.; Moreno, A.; Valls, A. Intelligent Tourism Recommender Systems: A Survey. *Expert Syst. Appl.* **2014**, *41*, 7370–7389. [[CrossRef](#)]
4. Ekstrand, M.D.; Riedl, J.T.; Konstan, J.A. *Collaborative Filtering Recommender Systems*; Now Publishers Inc.: Boston, MA, USA, 2011. [[CrossRef](#)]
5. Nguyen, L.V.; Jung, J.J. Crowdsourcing Platform for Collecting Cognitive Feedbacks from Users: A Case Study on Movie Recommender System. In *Springer Series in Reliability Engineering*; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 139–150. [[CrossRef](#)]
6. Nguyen, L.V.; Hong, M.S.; Jung, J.J.; Sohn, B.S. Cognitive Similarity-Based Collaborative Filtering Recommendation System. *Appl. Sci.* **2020**, *10*, 4183. [[CrossRef](#)]
7. Van Pelt, C.; Sorokin, A. Designing a Scalable Crowdsourcing Platform. In Proceedings of the ACM SIGMOD International Conference on Management of Data, Scottsdale, AZ, USA, 20–24 May 2012; ACM Press: New York, NY, USA, 2012; pp. 765–766. [[CrossRef](#)]
8. Su, X.; Khoshgoftaar, T.M. A Survey of Collaborative Filtering Techniques. *Adv. Artif. Intell.* **2009**, *2009*, 421425. [[CrossRef](#)]
9. Zhao, Z.D.; Shang, M.S. User-based Collaborative-Filtering Recommendation Algorithms on Hadoop. In Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, Phuket, Thailand, 9–10 January 2010; pp. 478–481. [[CrossRef](#)]

10. Yu, X.; Ren, X.; Sun, Y.; Gu, Q.; Sturt, B.; Khandelwal, U.; Norick, B.; Han, J. Personalized Entity Recommendation: A Heterogeneous Information Network Approach. In Proceedings of the 7th ACM International Conference on Web Search and Data Mining, New York, NY, USA, 24–28 February 2014; ACM Press: New York, NY, USA, 2014; pp. 283–292. [[CrossRef](#)]
11. Xue, G.R.; Lin, C.; Yang, Q.; Xi, W.; Zeng, H.J.; Yu, Y.; Chen, Z. Scalable Collaborative Filtering Using Cluster-based Smoothing. In Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, 15–19 August 2005; ACM Press: New York, NY, USA, 2005; pp. 114–121. [[CrossRef](#)]
12. Wang, J.; De Vries, A.P.; Reinders, M.J. Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion. In Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, WA, USA, 6–11 August 2006; ACM Press: New York, NY, USA, 2006; pp. 501–508. [[CrossRef](#)]
13. Meng, S.; Dou, W.; Zhang, X.; Chen, J. KASR: A Keyword-aware Service Recommendation Method on Mapreduce for Big Data applications. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 3221–3231. [[CrossRef](#)]
14. Thakkar, P.; Varma, K.; Ukani, V.; Mankad, S.; Tanwar, S. Combining User-based and Item-based Collaborative Filtering using Machine Learning. *Inf. Commun. Technol. Intell. Syst.* **2019**, *15*, 173–180.
15. Kant, S.; Mahara, T. Merging User and Item based Collaborative Filtering to Alleviate Data Sparsity. *Int. J. Syst. Assur. Eng. Manag.* **2018**, *9*, 173–179. [[CrossRef](#)]
16. Quinn, A.J.; Bederson, B.B. Human Computation: A Survey and Taxonomy of a Growing Field. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI'11, New York, NY, USA, 7 May 2011; Association for Computing Machinery: New York, NY, USA, 2011; pp. 1403–1412. [[CrossRef](#)]
17. Von Ahn, L.; Dabbish, L. Designing Games with a Purpose. *Commun. ACM* **2008**, *51*, 58–67. [[CrossRef](#)]
18. Amsterdamer, Y.; Grossman, Y.; Milo, T.; Senellart, P. Crowdminder: Mining Association Rules from The Crowd. *Proc. VLDB Endow.* **2013**, *6*, 1250–1253. [[CrossRef](#)]
19. Amsterdamer, Y.; Grossman, Y.; Milo, T.; Senellart, P. Crowd Mining. In Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, New York, NY, USA, 22–27 June 2013; Association for Computing Machinery: New York, NY, USA, 2013; pp. 241–252. [[CrossRef](#)]
20. Sheehan, K.B. Crowdsourcing Research: Data Collection with Amazon's Mechanical Turk. *Commun. Monogr.* **2018**, *85*, 140–156. [[CrossRef](#)]
21. Libert, B.; Spector, J.; Tapscott, D. *We are Smarter than Me: How to Unleash the Power of Crowds in Your Business*; Pearson Prentice Hall: Upper Saddle River, NJ, USA, 2007.
22. Brabham, D.C.; Sanchez, T.W.; Bartholomew, K. Crowdsourcing Public Participation in Transit Planning: Preliminary Results from The Next Stop Design Case. In Proceeding of the Transportation Research Board 89th Annual Meeting, Washington, DC, USA, 10–14 January 2010; pp. 10–37.
23. Budde, A.; Michahelles, F. Towards An Open Product Repository Using Playful Crowdsourcing. In *Service Science—Neue Perspektiven für die Informatik. Band 1*; INFORMATIK: Lepizig, Germany, 2010; pp. 600–605.
24. Liu, H.; Hu, Z.; Mian, A.; Tian, H.; Zhu, X. A New User Similarity Model to Improve the Accuracy of Collaborative Filtering. *Knowl.-Based Syst.* **2014**, *56*, 156–166. [[CrossRef](#)]
25. Galitz, W.O. *The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques*; John Wiley & Sons: Hoboken, NJ, USA, 2007.
26. Mandel, T. *The Elements of User Interface Design Vol. 20*; Wiley: New York, NY, USA, 1997.

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).