

Toward a Ubiquitous Personalized Daily-Life Activity Recommendation Service with Contextual Information: A Services Science Perspective

Chen-Ya Wang^{1,2}, Yueh-Hsun Wu¹, Seng-cho T. Chou¹

¹Department of Information Management, National Taiwan University

²Department of Information Management, Lunghwa University of Science and Technology
d91003@im.ntu.edu.tw, b90042@im.ntu.edu.tw, chou@ntu.edu.tw

Abstract

In recent years Services Science has been an emerging discipline that aims to promote service innovation and increase service productivity by aligning scientific, management, and engineering perspectives. It emphasizes that service innovation should be able to create value for both services providers and consumers. To realize the core thinking of services science, that is, high value and high productivity, service design has to incorporate many factors into its consideration. Based on the ideas of this new research field, we develop a personalized daily-life activity recommendation service that includes information behavior, business value, and technology architecture as our service design considerations. Our service can be requested under a ubiquitous environment and include users' contextual information which is an important factor in information behavior. With regard to IT architecture, we use the service-oriented architecture (SOA) that provides the flexibility and extensiveness of technology as well as permit new innovative services to be easily added.

1. Introduction

The development and application of Internet and information technology grew rapidly during the past few decades. In the meanwhile, the openness and flexibility of IT capability are also growing quickly. The role of information technology has been transformed from a supporting position to a driving one and will continue to progress to the production of a new business model. This kind of progress provides a lot of opportunities for various services to spring up. Traditionally, most IT applications or services were designed from the technology perspective, without integrating other important viewpoints, such as human behavior, business values, culture, and so on [20, 22]. Nowadays, globalization and a highly competitive environment have pushed organizations to reconsider their business values from different aspects and to strive for service innovation in order to gain a new competitive advantage. Many organizations have

transformed their perspectives, and now view IT as services rather than as supporters. Take Software as a Service (SaaS) for example, it is a software application delivery model where companies provide their software functions through networks [1]. Customers pay for using these functions rather than owning them. Following this trend of innovation, Services Science has emerged from the demand for service innovation, and emphasizes the combination of social science, business management, and technical engineering concepts [6, 20, 22].

With the advent of the ubiquitous age, various kinds of communication devices, as well as the related application services, are developing with surprising speed. IT architecture can be designed as an open platform that can be accessed from any device, anytime, and from anywhere. The IT capability also now makes it possible to provide "personalized services" and not remain limited to the classical "mass services" [1, 18]. The aim of service is to deliver service experience and consumer satisfaction that require human involvement and are therefore highly related to human feelings. Therefore, the economic value of services is greatly dependent on human factors [13]. According to the information behavior theory, the context might influence the human perception of information; that is, information seeking behavior is affected by the human-in-context factor [11]. In marketing, research on customer decision making behavior has evinced a similar opinion that the same customer may have different preferences or make different decisions under different contexts [5, 14]. Consequently, the design of recommendation service should incorporate the human-in-context factor into its consideration. In this paper, we develop a personalized recommendation service called Daily-Life Activity Assistant (DLAA) from the services science perspective. Our recommendation service takes users' contextual information and flexible technology architecture into consideration and can be requested under a ubiquitous environment. By using the flexible concept hierarchy and dynamic clustering method, we provide a dynamic recommendation service highly related to the

users' information seeking context, based on the multidimensional recommendation model (MD model) proposed by Adomavicius and Tuzhilin [2]. We also design a peer group mechanism based on the collective concept of Web 2.0 so that users can construct their own peer groups. DLAA users can choose to get activity recommendations based on the users' ratings of their peer groups rather than using all users' ratings in the database.

2. Related Works

2.1 Services Science

The concept of services science arose from the meeting in 2002 of a research team consisting of IBM's Almaden Research Center and UC Berkeley's Professor Henry Chebrough, that discussed service issues from the aspect of social engineering systems. The importance of services science as the foundation for service innovation and economic growth were emphasized by IBM's CEO and chairman, Samuel Palmisano, who wrote an article in the U.S. Council on Competitiveness journal titled *Innovate America* in December 2004 [1, 13]. In 2005, services science was renamed as Services Sciences, Management, and Engineering (SSME) and its main goal was to promote service innovation and increase service productivity. SSME is an interdisciplinary approach involving computer science, operations research, industrial engineering, mathematics, management sciences, decision-making theory, social and cognitive sciences, and legal sciences [1, 6, 13, 22]. IBM is the primary leader of SSME and promotes the concept by cooperating with universities. Many business administration programs in European and U.S. schools have already provided courses related to SSME [1, 6, 7].

SSME has three objectives: (1) to provide a systematic way to analyze services scientifically, manage services effectively, and maximize services productivity, (2) to solve problems that arise from the intangibility, simultaneity, and heterogeneity characteristics of services, and (3) to explore a framework with which to develop innovation in a systematic way [13]. To improve productivity and promote service innovation, many endeavors are needed concerning the following research issues: management and measurement of service innovation, technology to improve service effectiveness, measurement of service productivity, methodologies and tools to improve service quality and efficiency, and so on [1, 21, 22]. Besides offering courses in universities, several influential academic research centers were established to facilitate SSME research. Many international conferences also accelerated the progress of SSME research [18]. In many academic

journals, SSME has gradually been viewed as a research topic worthy of deep exploration. Communication of ACM also published a special issue for SSME in July 2006 to provide a multidisciplinary viewpoint and proposed some future research issues.

2.2 IT-Enabled Services

Technological development makes many new services possible and creates many innovative services, such as information services offered by companies like eBay and Google [16, 18]. Information technology not only helps in the production of goods but also advances services. Under this trend, the service research implies that every organization has to transform itself into a service organization, to use service to create competitive advantages and enhance revenues [18].

SSME is the study of service systems; it aims to create new service systems or improve existing service systems. Technology provides us with the ability to enable service innovation and boost service business. SaaS is a good example. Another example is the On-Demand Service business model. With the IT capabilities of simplification, flexibility, openness, and extensiveness, on-demand service/application/business can respond quickly to any customer demand, market opportunity, or external threat by integrating various business processes. To operate the on-demand model, organizations have to be able to utilize their resources flexibly and variably as well as to integrate people, processes, and internal and external information, to quickly sense and respond to the constantly changing environment. IT should provide such a platform that is characterized by standardization and componentization, which can be used in a heterogeneous, distributed, and a ubiquitous environment [1, 16]. From the IT-enabled services viewpoint, SOA (Service-Oriented Architecture) provides the infrastructure that makes on-demand service/application/business possible.

The design concept of modularization also increases the flexibility and capability required for services innovation and IT-enabled services. Organizations can identify various components of specific product/service. By the reuse and recombination of these components, modularization provides the ability for businesses to offer quick, customized product/service without totally destroying old product/service designs [16]. This kind of concept is not limited to use in technological application or IT architecture design. It is also a powerful concept that can be applied in all aspects of a business, such as: products/services design, business process, organization structure, and so on.

2.3 Information Behavior

During the past decade, information science communities have gradually paid greater attention to information behavior when designing application systems. Traditional application system designs are usually system-centered and exclude the information behavior factor. Until recently, more and more applications have transformed into user-centered approaches to match the users' behavior model. The major conceptual developments in information behavior have made a similar paradigm shift from system-centered to user-centered, since the 1980s. The increase of conceptual development and research in information behavior can not be ignored. The concepts developed by Dervin and Wilson are the ones frequently discussed and in use [11].

Dervin proposed the sense-making approach in 1976; the central idea of sense-making is determining "how people make sense of their world". The core elements of sense-making are: situation, gap/bridge, and outcome. *Situation* defines the context in which information needs arise; i.e., sense-making and sense-unmaking by humans involve a constant journey through time and space. *Gap* identifies the difference between the contextual situation and the desired situation. *Outcome* is the result of the person's experience of the world, i.e., the consequences of the sense-making process. *Bridge* is the medium that decreases the gap between situation and outcome. To some extent, sense-making can be defined as the process of creating situation awareness in situations of uncertainty [11].

Wilson developed an information seeking behavior model in 1981 that included the influence of an individual's physiological, cognitive and effective needs. He then proposed an integrative model of information need, information seeking behavior and information use in 1994. This model also claims that human's information seeking behavior is also influenced by the context of an information need arising out of a situation related to the person's environment, social roles and individual characteristics. The context might affect human perception of barriers to information seeking. Wilson modified the model in 1999 and placed Ellis' list of characteristics into his information seeking behavior model [11].

From the information behavior aspects mentioned above, we can find that the human perceptions of information behavior are affected by the context where they are situated [10, 11, 17]. This kind of perception might influence users' satisfaction of information service quality. In our ubiquitous personalized daily-life activity recommendation service, we take context as an important service design factor.

2.4 Personalized Recommendation Service

Personalized recommendation service aims to provide products, content, and services tailored to individuals, satisfying their needs in a given context based on knowledge of their preferences and behavior [4]. The personalized services are usually realized by the form of recommender systems. Recommender systems appeared as an independent research field in the mid-1990s [3]. They help users deal with information overload by providing personalized recommendations related to products, content, and services, usually accomplished by the use of personal profile information and item attributes. In the past decade, most works focused on modifying algorithms for greater effectiveness and correct recommendations [2]. They used methods from disciplines such as human-computer interaction, statistics, data mining, machine learning, and information retrieval [3, 4]. Recommender systems can be classified into three types according to how recommendations are made [3, 4]:

(1) Content-based Recommendation

It recommends items to users that are similar to those they preferred previously. The analysis of similarity is based on the items' attributes.

(2) Collaborative Recommendation

It recommends items to users according to the item ratings of other people who have characteristics similar to their own. The analysis of similarity is based on the users' tastes and preferences.

(3) Hybrid Recommendation

It is a combination of content-based and collaborative recommendations.

Traditionally, recommender systems usually compute the similarity using two-dimensional user-item information under an e-commerce environment. They failed to take into consideration contextual information which might affect decision making behavior, such as time, location, companions, weather, and so on. According to information behavior theory, the context of information needs is an important factor of information seeking behavior [10, 11, 17]. Including human-in-context information as one system design factor is necessary for producing more accurate recommendations.

Herlocker and Konstan successfully introduced additional information in the traditional recommendation methods [2]. They included task information in recommender system design and argued that including users' task information can lead to better recommendations. However, their approach was still focused on two-dimensional user-item information. Adomavicius and Tuzhilin proposed a multidimensional approach to incorporate contextual

information into the design of recommender systems [2]. They also proposed a multidimensional rating estimation method based on the reduction-based approach, and tested their methods on a movie recommendation application that took time, place, and companion contextual information into consideration.

From the aspect of recommendation service provided by recommender systems, most works focused on products, content, and service recommendations based on web e-commerce [8]. However, it is not sufficient to pass over contextual information and just consider users and items in many applications and services, especially when making recommendations under a ubiquitous environment. For example, a user might like to read entertainment or movie news on weekend mornings, rather than read the financial or political news on the web. In addition, a user might prefer to get an activity suggestion that takes time, place, companion, and weather into consideration in a ubiquitous environment. Consequently, excluding the context factor from the consideration of recommendation service design, the quality of the recommendations comes under question. Contextual information should be incorporated into recommendation service implementation consideration [2, 9]. In this paper, we use service-oriented architecture and provide a daily-life activity recommendation service that includes contextual information in a ubiquitous environment.

3. Ubiquitous Daily-Life Activity Recommendation Service

Our design goal is to implement a ubiquitous Daily-Life Activity Assistant (DLAA), including contextual information. Most existing recommender systems are not suitable for providing service in a ubiquitous environment because of the need for plentiful database computing. In DLAA, we adopt Service-Oriented Architecture (SOA) to implement our system. Users can request our web services for activity recommendations by providing their personal profile data and contextual information through access devices. We use name, age, gender, single/married, location, and some registered information as users' original profile data and dynamically cluster users based on contextual information, before making activity recommendations to users. At the same time, users can rate the recommendations and help us to modify the accuracy of our recommendations. In DLAA, we also design a peer group mechanism related to the collective concept of Web 2.0 whereby users can construct their own peer groups. DLAA users can choose to get activity recommendations from the users' ratings of their peer groups rather than using all users' ratings.

We will describe our recommendation service design in more detail in the following sections.

3.1 DLAA Serviced-Oriented Architecture

SOA is an architecture that evolved from distributed computing based on a request/reply design paradigm. The business processes or application functions are modularized and presented as services for users. The service interface is independent of the implementation and services can be accessed without knowledge of their underlying platform implementation. With its loosely coupled and interoperable nature, SOA delivers greater business efficiency and agility to respond more quickly and with greater cost-effectiveness, to business changes. The business processes are no longer constrained by the limitations of the underlying infrastructure. In an SOA environment, business can flexibly plug into or compose new services under existing IT infrastructure and safeguard their investments. SOA is the next wave of application and service development architecture.

We designed DLAA as an SOA-based recommendation service using web services technology. The basic unit of communication in web services is a message rather than an operation, and is often referred to as message-oriented services. Figure 1 shows the DLAA SOA environment.

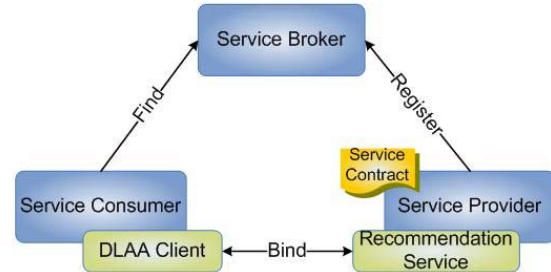


Figure 1. DLAA Service-Oriented Architecture

The Service provider offers the recommendation service through well-defined service contracts and has to publish the service contracts for its services in the registry. A service contract represents an agreement by the joining parties and binds the users with the service provider. It is a contract recorded as a metadata that defines the rules for interacting with a specified service. The service contract is the key to loose coupling. It provides the ability to use as little information as possible that is necessary for governing the relationships between interacting parties. In the meanwhile, service brokers maintain the service registry that acts as a service directory listing. Service consumers can look up the services in the registry and invoke the service by sending messages that meet the service contract format.

SOA can construct loosely coupled relationships for the interacting parties so that service providers can independently create and control each component of the IT environment. In our design, DLAA is a daily-life activity recommendation service that can provide service under a ubiquitous environment. It is an assistant/agent of many possible kinds of services that can be used any time and anywhere by any access device. Any new innovative service for a ubiquitous environment can be implemented as another assistant/agent and integrated into our architecture to provide services.

3.2 Multiple Dimensions and Concept Hierarchy

In DLAA, we use the multidimensional model proposed by Adomavicius and Tuzhilin to store the information related to user, item, and context factors, where each factor can be represented as a concept hierarchy. Generally speaking, dimensions represent the recommendation space in a recommender system. Most classical recommender systems operate in the two-dimensional $User \times Item$ space and the rating value of each user-item pair can be expressed as a rating function: $R : User \times Item \rightarrow Ratings$. According to the multidimensional recommendation model (MD model), the recommendation space can not be limited to only two dimensions. By extending the concept of data warehousing and OLAP application in databases, the multidimensional method provides us with the ability to incorporate more factors into consideration while making recommendations, such as time, location, companions, and so on [2]. In DLAA, we use time, location, weather, and companions as our contextual information dimensions, and the recommendation space is defined as:

$$S = User \times Activity \times Time \times Location \times Weather \times Companion$$

In the MD model, a dimension D_i is the Cartesian product of attributes and can be expressed as $D_i \subseteq A_{i1} \times A_{i2} \times \dots \times A_{ij}$. Each attribute A_{ij} defines a set of attribute values of one particular dimension. For example, the *User* dimension is defined as: $User \subseteq Name \times Age \times Gender \times IsMarried$. Similarly, the *Location* dimension is defined as: $Location \subseteq Country \times City \times Place$. The attributes describing a dimension D_i sometimes is called a profile, such as user profile, time profile, and location profile. For each dimension, the attributes can be represented as a concept hierarchy which consists of several levels of concepts. The top-down view of a concept hierarchy is organized from generalization to specialization; i.e., the higher the layer, the more generalized the layer. Take *Time* for example, its concept hierarchy can be expressed as Figure 2.

Given dimensions: D_1, D_2, \dots, D_n , ratings are the rating domain which represents the set of all possible rating values under the recommendation space $D_1 \times D_2 \times \dots \times D_n$. The rating function R is defined as: $R : D_1 \times D_2 \times \dots \times D_n \rightarrow Ratings$. Based on the recommendation space:

$User \times Activity \times Time \times Location \times Weather \times Companion$ our rating prediction function $R(u, a, t, l, w, c)$ specifies how much user u likes activity a , accompanied by c at location l and time t under weather w , where $u \in User$, $a \in Activity$, $t \in Time$, $l \in Location$, $w \in Weather$, and $c \in Companion$. The ratings are stored in a multidimensional cube and the recommendation problem is to select the maximum or top-N ratings of $R(u, a, t, l, w, c)$.

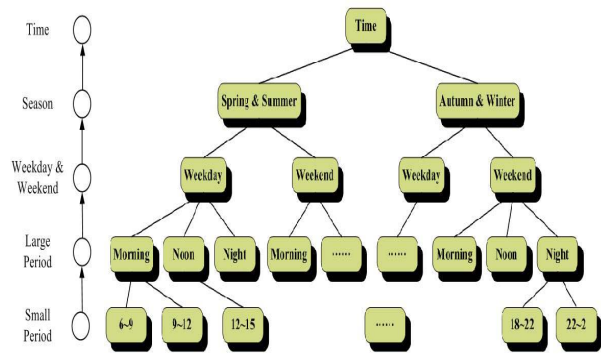


Figure 2. Time Concept Hierarchy

3.3 Reduction-Based Approach and Rating Estimation

The computation of recommendations grows exponentially with the number of dimensions. The reduction-based approach can reduce the multidimensional recommendation space to the traditional two-dimensional recommendation space by fixing the values of context dimensions, and improve the scalability problem [2, 9]. Assume that $R_{User \times Activity \times Time}^D : U \times A \times T \rightarrow rating$ is a three-dimensional rating estimation function supporting time and D contains the user-specified rating records (*user, activity, time, rating*). It can be expressed as a two-dimensional rating estimation function: $\forall (u, a, t) \in U \times A \times T, R_{User \times Activity \times Time}^D(u, a, t) = R_{User \times Activity}^{D[Time=t]}(u, a)$, where $D[Time=t](User, Activity, Rating)$ is a set of rating records by selecting *Time* dimension which has value t and keeping the values of *User* and *Activity* dimensions.

Another problem is rating estimation that $D[Time=t](User, Activity, Rating)$ may not contain enough ratings for recommendation computation. The rating estimation process is particularly complex in a

multidimensional recommendation space. In DLAA, we use the rating aggregations of contextual segment S_t that expresses the superset of the context t when insufficient ratings are found in a given context value t . The rating of $R(u,a,t)$ is expressed as:

$$R_{User \times Activity \times Time}^D(u,a,t) = R_{User \times Activity}^D[Time \in S_t](User, Activity, AGGR(rating))(u,a)$$
, where $AGGR(rating)$ is the rating aggregations of contextual segment S_t . Thus, the rating prediction function for weekend afternoon might be presented as the formula:

$$R_{User \times Activity \times Time}^D(u,a,t) = R_{User \times Activity}^D[Time \in weekend](User, Activity, AGGR(rating))(u,a)$$
.

By using the reduction-based approach, the computing of multidimensional recommendation space can be simplified by fixing the values of context dimensions. Using the reduction-based approach in DLAA meets the task of our recommendation service which aims to provide activity recommendations according to a given context condition.

3.4 DLAA System Architecture

DLAA is a ubiquitous recommendation service including contextual information consideration. We implement DLAA using the SOA architecture and multidimensional recommendation model. Users can get different activity recommendation lists in different situations via their access devices. The system architecture of DLAA is shown in Figure 3.

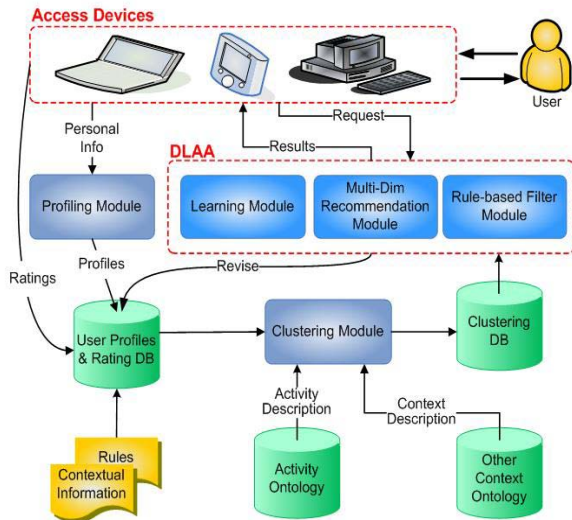


Figure 3. DLAA System Architecture

DLAA is composed of a profiling module, a clustering module, a learning module, a rule-based filter module, and a Multi-Dim recommendation module. The primary design and functions of these modules are described as follows.

● **Profiling Module**

The profiling module deals with and organizes profile data used in DLAA. There are six types of profile data for the recommendation space of DLAA.

User and *Activity* are two-dimensional factors considered by traditional recommender systems. *Time*, *Location*, *Weather*, and *Companion* are four contextual factors we incorporated into our system design. By using the snowflake schema concept of data warehouse, we implement the profiling module to organize and store these profiles. The structure of our profiles is shown in Figure 4.

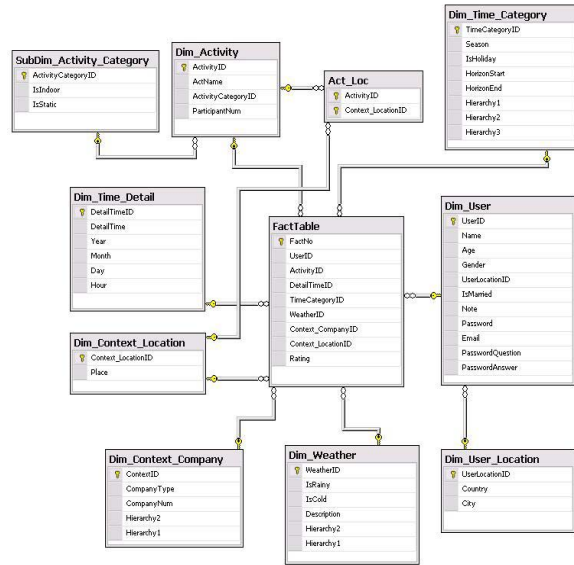


Figure 4. DLAA profiles structure

We record name, age, gender, single/married, location, and some register information as the user profile. Activity profile is classified as indoor/outdoor, static/dynamic. We keep the Time dimension data in Time_Category and Time_Detail. The time concept hierarchy is structured as season, weekday/weekend, large period, and small period. Weather dimension records the IsRainy/IsCold and status data. The information related to Companion Dimension is organized as parent/child, peer (lover, brother/sister, and friend/colleague/classmate), and oneself.

● **Clustering Module**

When a user gives the contextual information in which he/she is situated, the clustering module dynamically finds a group of users who are similar to the present user based on the same contextual information. The module first retrieves the historical data of specific contextual conditions from the database and then calculates the similarity of users by activity type. The reason why we use activity type to cluster users is to avoid the over-sparseness problem. We assume that users of similar activity type are somewhat similar, and estimate their similarity using the function: $sim(A,B) = \cos(A,B) = \frac{A \cdot B}{|A| \times |B|}$. Then, the

module produces a set of users similar to the present user, based on the center-based neighborhood manner.

- **Learning Module**

This module acts as a feedback learning mechanism. DLAA defines a rating threshold which is 3 and gradually modifies a user's profile by excluding the activity pattern that the user had given a rating below the threshold.

- **Rule-Based Filter Module**

In DLAA, we propose a warm up mechanism to avoid the cold-start problem. Via this mechanism, we collect users' data related to their profile and ratings. However, some users might provide unreasonable ratings to DLAA, such as eating breakfast at night. The rule-based filter module defines some rules to avoid inaccurate recommendations caused by 'dirty' data. This module will exclude the activities that should not be recommended in some specific context.

- **Multi-Dim Recommendation Module**

The recommendation module deals with the rating estimation according to the similar users set derived from the clustering module. If the number of users in the set is greater than 3, the module finds "Common rated activities" to re-compute the similarity of the users set. Otherwise, the module uses original similarity derived from the clustering module because of data sparseness. Then, we calculate target user u and similar user's (u') average rating of each type, where $\overline{r^D(u)} = 1/x \sum_{s=1}^x r(u, a_s)$ and $\overline{r^D(u')} = 1/x \sum_{s=1}^x r(u', a_s)$. Finally, the recommendation module predicts the target activity's (a) rating by adopting users' similarity as the weighting factor and excluding the users' differences by normalizing the rating. The rating function is defined as:

$$r^D(u, a) = \overline{r^D(u)} + y \sum_{u' \in U} \text{sim}(u, u') \times [r^D(u', a) - \overline{r^D(u)}], \text{ where } y = 1 / \sum_{u' \in U} |\text{sim}(u, u')|$$

is the normalizing factor.

The rating estimation method mentioned above is the default method we used in DLAA whereby the recommendations are made according to the whole data in our database. Based on the collective concept of Web 2.0, DLAA provides another rating estimation method that users can provide the constraint to include only some parts of data in the database as their recommendations base rather than the whole database. In DLAA, users can construct their own peer groups similar to the idea of web2.0 community. If users choose the demography or peer group method when searching for recommendation service in DLAA, they can set up the demographic characteristics or pick the peer group as the rule by which to filter the database. Then, DLAA gets a new similar user set as its rating estimation foundation. The Multi-Dim Recommendation Module will use the

same rating function defined above to predict ratings and give recommendations based on the limited data that the user had selected. This kind of choice for users provides some trustworthiness for our recommendations to some extent. The concept of this method is shown in Figure 5.

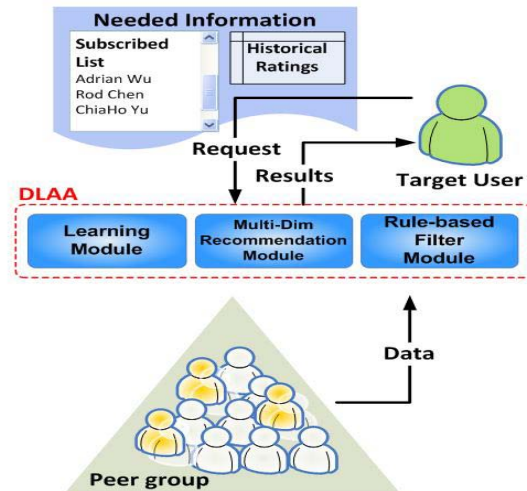


Figure 5. Peer group based rating estimation

4. Evaluation

The DLAA is a type of recommender system that includes the multidimensional concept under a ubiquitous environment. It can be evaluated using methods employed by other recommender systems. There are many types of accuracy metrics for evaluating the quality of a recommender system. In this paper, we use predictive accuracy metrics to examine the prediction accuracy of recommendations in DLAA. Predictive accuracy metrics are usually used to evaluate the system by comparing the recommender system's predicted ratings against the actual user ratings. Generally, Mean Absolute Error (MAE) is a frequently used measure for calculating the average absolute difference between a predicted rating and the actual rating. In addition, Normalized Mean Absolute Error (NMAE) represents the normalization of MAE which can balance the range of rating values, and can be used to compare the prediction results from different datasets. According to related research, the predictive accuracy of a recommender system will be acceptable when the value of NMAE is below 18% [8, 19]. In this paper, we use NMAE to evaluate DLAA and adopt the AllButOne method as our dataset selection strategy. The AllButOne strategy is widely used in many recommender systems [9]. This dataset's chosen strategy hides exactly one rating from the test dataset and predicts its rating based on other non-hidden ratings. Then we can compare the predicted data with the actual one.

In addition, the precision evaluation of recommender systems is mainly an extension of Information Retrieval to calculate the percentage of user-acceptable recommendations related to the total number recommendations from recommender systems. However, this type of data is not available in DLAA. Instead, we used the rate of adoption to evaluate the precision of DLAA. We also conducted the over-estimation/under-estimation Z-test of users' ratings to evaluate user satisfaction levels. With regard to the experimental dataset, many studies of recommender systems used publicly available data, from sites such as EachMovie, MovieLens, and Jester, for their experimental datasets [12]. Unlike traditional recommender systems, DLAA takes contextual information into consideration, and this type of data is not available in any public datasets. For this reason, we constructed an experimental website and invited end-users to use DLAA to get relevant contextual information and recorded their ratings for activities. In what follows, we will elucidate the experimental procedures and experimental results of DLAA.

4.1 Procedures

For a new domain recommender, it may be suitable to use synthetic data sets for the first evaluation [12]. DLAA is this type of recommender; however there are no existing available datasets. The experimental analysis of DLAA has three stages. Most recommender systems have a cold-start problem in the initial operation which indicates that they don't have enough ratings for recommendation computing. The first stage of our study aimed to generate pre-data for DLAA to avoid the cold-start problem. Following a three-step process, we generated pre-data of 100 users and 1319 ratings for DLAA. We first recruited 10 users of different ages, genders, and religions. They were asked to provide ratings based on randomized contextual information. This step generated 200 ratings (5%) of the pre-data. Then, we randomly generated 90 users who have similar backgrounds to the 10 initial users and fine tuned their contextual information. The ratings of these 90 users were randomly generated. This step simulated 760 (58%) ratings from the 90 users. In the third step, we found out which activities had fewer ratings than others and generated a further 359 (27%) rating records for them. In the first stage, we also validated the pre-data by analyzing the percentage of availability and the NMAE for the number of ratings. Figure 6 shows that the percent of availability reached 70% when the recommender system had 1300 ratings. This indicates that the users had a 70% probability of receiving this recommendation service using DLAA. Figure 7 shows the NMAE of our pre-data was 14.5%, which is an acceptable result.

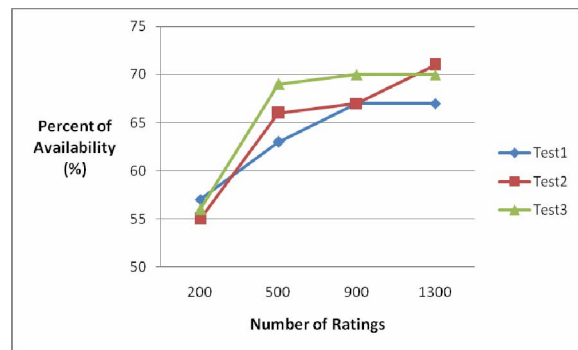


Figure 6. Percent of availability of pre-data

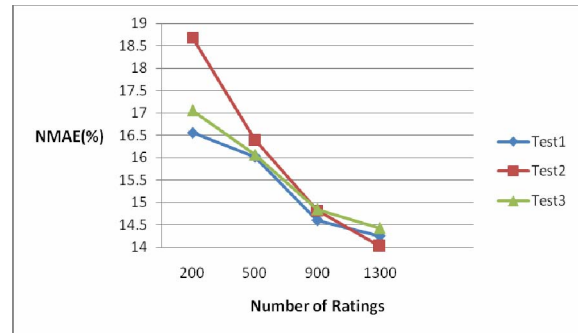


Figure 7. NMAE of pre-data

The second stage conducted an experiment using 47 university student participants. Each participant was asked to provide at least 10 pretest records to build their own profile as well as to complete 5 rounds of DLAA recommendations and give ratings. In this stage, we collected 730 ratings from 47 participants. There were 222 ratings which came from 33 participants who completed the experiment (i.e. 10 pretests and 5 rounds of recommendation service). These ratings included 6 unsatisfied recommendations. We then proceeded to the final stage of our experimental analysis by combining the data of the previous two stages. The NMAE was used to evaluate the minimum neighborhood size and the accumulated number of ratings. This stage also compared the results of the pre-data generated in stage 1 with the final data, which showed that inclusion of the real users' data did improve the system accuracy of DLAA. The precision of DLAA and the user satisfaction results were evaluated by the rate of adoption and the over-estimation/under-estimation Z-test of users' ratings.

4.2 Results

The Minimum Neighborhood Size

Neighborhood size indicates a number of grouped users and the size is used in the rating prediction to make a collaborative recommendation. The setting of the neighborhood size affects the accuracy of the

rating prediction and of the system performance. A small neighborhood size will reduce the accuracy of the prediction. However, a large neighborhood size will lower the recommender system performance. We conducted two phase minimum neighborhood size experimental analyses using the pre-data from stage 1 and the experimental data from stage 3. Figure 8 shows the NMAE improved when neighborhood size increased. The NMAE has no significant improvement however when the neighborhood size is greater than 50. According to this finding, the neighborhood size of DLAA was set at 50. We also found that the NMAE of phase 2 greatly improved after including experimental data from real participants.

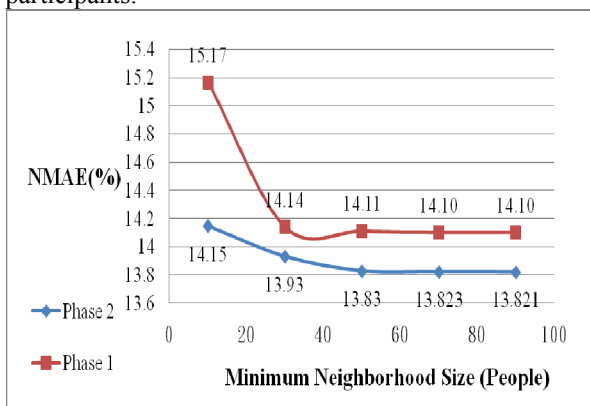


Figure 8. Minimum neighborhood size

The Accumulated Number of Ratings

Many recommender systems have a cold-start problem and it is reasonable to assume that the NMAE will reduce as the accumulated number of ratings increases. We analyzed the relationship between the accumulated number of ratings and NMAE. In stage 2 of our experiment, we got 222 ratings from 33 participants who completed the experiment. Excluding the 6 unsatisfied recommendations, we divided 216 ratings into four parts. Figure 9 shows the NMAE decreases gradually while the accumulated number of ratings increases. In the meanwhile, we find that the NMAE improved by 2% with outlier data removed.

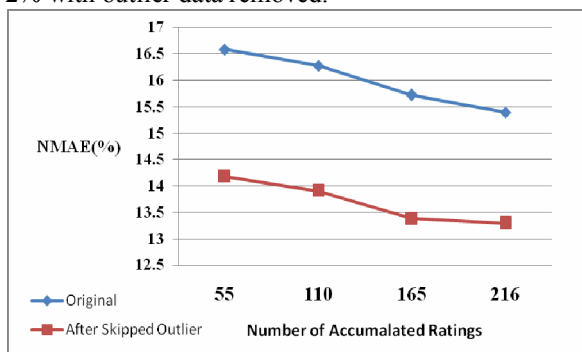


Figure 9. Accumulated number of ratings

The Rate of Adoption

In our experiment, participants rated recommendations provided by DLAA on a 1-10 scale. We divided the scale into three large groupings: 1-4 represents moderately acceptable, 5-7 represents acceptable, and 8-10 represents satisfied. The rate of adoption of our recommendations shows in Table 1. The percentage of acceptable recommendations for the DLAA is 92%.

Table 1. The rate of adoption of DLAA

Participant Ratings	Numbers	Percentages
Unsatisfied	6	3 %
Moderately Acceptable	12	5 %
Acceptable	72	32 %
Satisfied	132	60 %

User Satisfaction

A recommender system over-estimates the rating if the predicted rating is higher than the real rating provided by a user. Adversely, the recommender system under-estimates the rating when the predicted rating is lower than the real rating. Users will be satisfied with the recommendations while the over-estimation/under-estimation is not obviously significant. We conducted a Z-test using a 0.05 significance level to examine if the over-estimation/under-estimation was significant. The result of Z-value=1.36559 shows our null hypothesis can't be rejected, which means there are no significant differences between the predicted ratings and real ratings.

5. Conclusion and Future Works

“Services Science is an emerging discipline that focuses on fundamental science, models, theories and applications to drive innovation, competition, and quality of life through service(s)” [7]. SSME advocates service innovation and high service productivity to create organizational core competitive advantages. It claims that service design should be based on the dimensions of people, business, technology, and information [15, 20, 21]. Many organizations are facing challenges as they transform themselves into service organizations which have the characteristics of openness, flexibility, and agility.

From an IT-enabled services perspective, service innovation is supported by IT capability. IT-enabled service innovation still has to integrate viewpoints of people, business, technology, and information. Business agility is gained by adopting service structures for an organization’s IT-enabled services infrastructure, for example, service-oriented architecture. Flexible service structure also represents opportunities for quickly providing new services and reusing existing service components.

The ubiquitous personalized daily-life activity recommendation service we provided in this paper also takes several aspects into consideration. We adopted SOA as our technology architecture so that we can reuse our service components in the future and easily plug in any new innovative service quickly, flexibly, and at low-cost. The system design provides a more dynamic recommendation service highly related to the users' information seeking context by using the flexible concept hierarchy and the dynamic clustering method. The system architecture and model of DLAA can be used in the application of product recommendations, including contextual information, for example, for movies and restaurants. It can also integrate with the HomeCare application to provide activity recommendations for elderly people by incorporating their physical situations and schedules into consideration.

From a service design perspective, the more different the situation and application domain, the more different the information related to human-in-context factor consideration. Future research is required to understand what kinds of contextual information will affect information behavior and decision preferences in various application domains. Human emotional issues should be included in the service design, such as trust and privacy, and that will be the focus of our future work related to DLAA. In the end, from a services science perspective, gradually providing innovative services that can be easily added to our service structure is an issue for our future research.

Acknowledgement

This work was supported in part by the National Science Council Grants NSC95-2218-E-002-022/026, NSC96-3114-P-001-002-Y, and Intel Higher Education Research Grants on Digital Health (2006) and WiMAX (2007).

6. References

- [1] Abe, T. "What is Service Science?" *FRI Research Report No.246*, Fujitsu Research Institute, 2005.
- [2] Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. "Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach," *ACM Transactions on Information Systems* (23:1), 2005, pp. 103-145.
- [3] Adomavicius, G., and Tuzhilin, A. "Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," *IEEE Transactions on Knowledge and Data Engineering* (17:6), 2005, pp. 734-749.
- [4] Adomavicius, G., and Tuzhilin, A. "Personalization Technologies: A Process-Oriented Perspective," *Communications of the ACM* (48:10), 2005, pp. 83-90.
- [5] Bettman, J. R., Johnson, E. J., and Payne, J. W. "Consumer Decision Making," in *Handbook of Consumer Behavior*, T. Robertson and H. Kassarian eds., Englewood Cliffs, NJ: Prentice-Hall, 1991, pp. 50-84.
- [6] Bitner, M., J., and Brown, S., W. "The Evolution and Discovery of Services Science in Business Schools," *Communications of the ACM* (49:7), 2006, pp. 73-78.
- [7] Bitner, M., J., Brown, S., W., Goul, M., and Urban, S. "Services Science Journey: Foundations, Progress, Challenges," Center for Services Leadership, W. P. Carey School of Business at Arizona State University.
- [8] Brozovsky, L., Petricek, V. "Recommender System for Online Dating Service," eprint arXiv:cs/0703042, 2007. [Available at <http://aps.arxiv.org/abs/cs/0703042>]
- [9] Cho, S., Lee, M., Jang, C., and Choi, E. "Multidimensional Filtering Approach Based on Contextual Information," *International Conference on Hybrid Information Technology (ICHIT'06)*, 2006.
- [10] Choo, C., W., Detlor, B., and Turnbull, D. "A Behavioral Model of Information Seeking on the Web -- Preliminary Results of a Study of How Managers and IT Specialists use the Web," in *Proceedings of the 61st Annual Meeting of the American Society for Information Science*, Pittsburgh, PA, 1998, pp. 290-302.
- [11] Godbold, N. "Beyond Information Seeking: Towards a General Model of Information Behavior," *Information Research*, (11:4), 2006, paper 269.
- [12] Herlocker, J., L., Konstan, J., A., Terveen, L., G., and Riedl, J. T. "Evaluating Collaborative Filtering Recommender Systems," *ACM Transactions on Information Systems*, (22:1), 2004, pp. 5-53.
- [13] Hidaka, K. "Trends in Services Sciences in Japan and Abroad," *Science & Technology Trends: Quarterly Review* (19), 2006, pp. 35-47.
- [14] Klein, N. M., and Yadav, M. "Context effects on effort and accuracy in choice: An inquiry into adaptive decision making," *Journal of Consumer Research* (16), 1989, pp. 410-420.
- [15] Maglio, P., P., M., Srinivasan, S., Kreulen, J., T., and Spohrer, J. "Service Systems, Service Scientists, SSME, and Innovation," *CACM* (49:7), 2006, pp. 81-85.
- [16] Paulson, L., D. "Services Science: A New Field for Today's Economy." *IEEE Computer Magazine*, August 2006, pp. 18-21.
- [17] Pettigrew, K.E., Fidel, R., and Bruce, H. "Conceptual frameworks in information behavior," *Annual Review of Information Science and Technology (ARIST)* (35), 2001, pp. 43-78.
- [18] Rust, R., T., and Miu, C. "What Academic Research Tells Us about Service," *CACM* (49:7), 2006, pp. 49-54.
- [19] Sarwar, B., Karypis, G., Konstan, J., A., and Riedl, J. "Application of Dimensionality Reduction in Recommender Systems—a Case Study," in *Proceedings of the ACM WebKDD Workshop*, 2000.
- [20] Sheth, A., Verma, K., and Gomadam, K. "Semantics to Energize the Full Services Spectrum," *Communications of the ACM* (49:7), 2006, pp. 55-61.
- [21] Spohrer, J., and Maglio, P., P. "The Emergence of Service Science: Toward Systematic Service Innovations to Accelerate Co-Creation of Value," IBM Almaden Research Center.
- [22] Spohrer, J., and Riecken, D. "Services Science: Introduction," *CACM* (49:7), 2006, pp. 30-32.