Large Scale Evaluation of Multi-Mode Recommender System Using Predicted Contexts with Mobile Phone Users

Takashi Shiraki
Information and Media Processing Laboratories, NEC Corporation
t-shiraki@bu.jp.nec.com

Chihiro Ito
Information and Media Processing Laboratories, NEC Corporation
c-ito@az.jp.nec.com

Takeo Ohno
Service Platforms Research Laboratories, NEC Corporation
takeo@aj.jp.nec.com

1753 Shimonumabe, Nakahara-ku, Kawasaki, Kanagawa, 211-8666, Japan

ABSTRACT
Context-aware recommender systems can improve user satisfaction with recommended information when user preferences change depending on user contexts (e.g. location, time, and weather). However, the effect of each context on various user preferences has yet to be fully elucidated. In addition, few examples address this challenge in large-scale real-world experiments. Therefore, relationships between mobile phone users' contexts and their preferences have been evaluated. Our system calculates the intensity of various user preferences (e.g. favorite area and type of content) and recommends content accordingly. It was applied to a restaurant recommendation service with 2,762 mobile phone users over a two-month period. The evaluation results indicate quantitative evidence that user preferences depend on context information. In particular, location information strongly correlates with users' interest in recommending different types of restaurants. Finally, predicted context data is shown to be more effective for recommendation than raw data.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Information filtering, Relevance feedback, Retrieval models, Selection process.*

General Terms
Algorithms, Experimentation, Human Factors

Keywords
Recommender System, Context-awareness, Mobile Systems

1. INTRODUCTION
Due to the recent information overload, recommendation schemes that adapt information to changing user needs are becoming more important. Recommender systems are increasingly being used for services on the web and on devices (e.g. video recorders). For example, Amazon.com, iTunes, and TiVo mainly recommend books, music, and TV programs, respectively. Recommender systems score items using user profiles, content data, user feedback such as purchase logs, and user contexts.

Many personalized recommender systems using profiles, content data, and feedbacks have already been proposed. They are suitable for services for providing users with items that match their preferences. Amazon.com recommends products by using item-based collaborative filtering, which finds similar items from user feedback [6]. The Netflix Grand Prize solution [5, 10, 13] had a root mean square error (RMSE) 10% better than that of the Netflix's legacy algorithm Cinematch.

However, many challenges still remain in predicting various user needs, which change depending on context. Cyberguide [1], GUIDE [3], and COMPASS [11] are mobile tour guide systems that provide information such as tourist resorts and exhibits using the time of day and user current locations as contexts. Magitti [2] is a mobile leisure guide system that predicts a user's current and future activity. It is based on extensive fieldwork in which an online survey and many interviews were conducted. Reference [9] also predicts a user's current activity from past / current / future contexts of all users. Though all these systems take into account contexts that influence users' interests, these contexts have rarely been evaluated precisely in large-scale experiments, because of the difficulty in obtaining both user contexts (e.g. location) and user feedback in the real world.

To overcome this, we have developed a Multi-Mode Recommender System (MMRS), which can adapt to various contextual conditions and evaluate relationships between contexts and user interests automatically. We applied it to a trial recommendation service of about 28,000 restaurants for 2,762 mobile phone users with the largest Japanese mobile communications operator NTT DOCOMO. As a result, we clarified the following issues based on quantitative evaluations in a large-scale experiment.

- Users' favorite items depend on user contexts.
- The MMRS can find effective profile/context.
- Predicted contexts can be more effective for recommendation.

2. MMRS
Figure 1 shows the overall architecture of the MMRS. The MMRS consists of four components: model learner, user-mode estimator, sub-recommender systems (SRSes), and item selector. Model learner constructs user-mode learning models that are calculated from past user behavioral logs. The user-mode estimator predicts the current user-mode from the user-mode learning models when a user requests recommendations. Each SRS makes an item list by using each algorithm such as content-based filtering and collaborative filtering. The item selector retrieves results from item lists of SRSes. We describe those algorithms in detail in the subsections 2.1-2.4.

The MMRS has four types of input: profile data, context data, feedback logs, and item data. Profile data are static-feature user data, such as age, gender, and whether she/he smokes. Context data are dynamic-feature user data, such as time, location, and pulse rate. Feedback logs are user positive feedback data in recommendation services, such as purchases, clicks, and

----
CARS-2011, October 23, 2011, Chicago, Illinois, USA.

bookmarks. Item data are content data that should be retrieved in recommendation services.

The MMRS has two types of output: an item list and user-modes. An item list is set of recommended items that the MMRS retrieves as results. User-modes are defined as user interests in selecting information such as location of users' favorite contents, user preference, and user's favorite method for retrieving items.

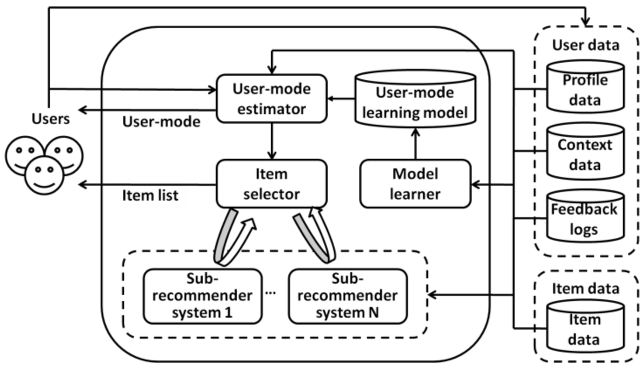

Figure 1. Overall architecture of MMRS

2.1 Model learner

The model learner is used for offline-processing that produces user-mode learning model from profile data, past context data, feedback logs, and item data. User-mode learning model is the set of intensities of user-modes for all profile/context combinations. The intensity is based on the probability of category j_i of user-mode i in each profile/context combination. It is given by

$$P\left(M_{ij_i}\middle|C_{1k_1,}C_{2k_2,}\dots,C_{Hk_H}\right)=\frac{P\left(M_{ij_i},C_{1k_1,}C_{2k_2,}\dots,C_{Hk_H}\right)}{\sum_{j_i}P\left(M_{ij_i},C_{1k_1,}C_{2k_2,}\dots,C_{Hk_H}\right)}\quad(1)$$

M_{ij_i}: User-mode j_i of user-mode category i

$(i=1,2,\dots,I,\ j_i=1,2,\dots,J_i)$

C_{hk_h}: Profile/context k_h of user profile/context category h

$(h=1,2,\dots,H,\ \ k_h=1,2,\dots,K_h).$

We use the Naïve Bayes algorithm to calculate Equation (1). Under the Naïve Bayes Assumption between profile/context sets, we have

$$P\left(M_{ij_i},C_{1k_1,}C_{2k_2,}\dots,C_{Hk_H}\right)=P\left(M_{ij_i}\right)\prod_{h=1}^{H}P\left(C_{hk_h}\middle|M_{ij_i}\right)\ .\quad(2)$$

The Naïve Bayes algorithm has the limitation in handling the dependency among profiles/contexts. However, we apply it to the MMRS because it has the following merits for the conditional independence assumption between profile/context sets.

- Space complexity can be reduced from $\prod_{h=1}^{H}K_h\in O(K^H)$ to $\sum_{h=1}^{H}K_h\in O(KH)$ where $K=\max_{h=1,2,\dots,H}K_h$
- We can evaluate each profile/context set in a fair manner
- It is easy to use in distributed systems

Additionally, $P\left(C_{hk_h}\middle|M_{ij_i}\right)$ in Equation (2) is calculated using frequency $\mathrm{Freq}\,(X)$, which is defined as the amount of user feedback in condition X during the experiment period, as below,

$$P\left(C_{hk_h}\middle|M_{ij_i}\right)=\frac{\mathrm{Freq}\,(M_{ij_i},C_{hk_h})}{\mathrm{Freq}\,(M_{ij_i})}.\quad(3)$$

To eliminate zeros from the denominator in Equation (3), we use Laplace smoothing [7], which simply adds one to each count $\left(\mathrm{Freq}\left(M_{ij_i},C_{hk_h}\right)=1,\ \mathrm{Freq}\left(M_{ij_i}\right)=K_h,\text{for all }i,j_i,h,k_h\right)$, as an initial condition. The model learner calculates $\mathrm{Freq}\left(M_{ij_i}\right)$ for all i,j_i and $\mathrm{Freq}\left(M_{ij_i},C_{hk_h}\right)$ for all i,j_i,h,k_h on a regular basis.

2.2 User-mode estimator

User-mode estimator is used for online-processing that predicts current user-mode intensities from user-mode learning models, profile data, and current context data. When the MMRS receives a query from a user, the user-mode estimator obtains probabilities in Equation (1) for a profile/context combination of the user.

Figure 2 shows a screenshot of the user-mode intensities. In the case of user-mode "Preference," the equalizer bars in the screen indicate that user-mode estimator predicts "Gourmet meal," which here means a relatively expensive meal, as the most interesting for the user. The user can also move the equalizer bars to control recommendation results interactively.

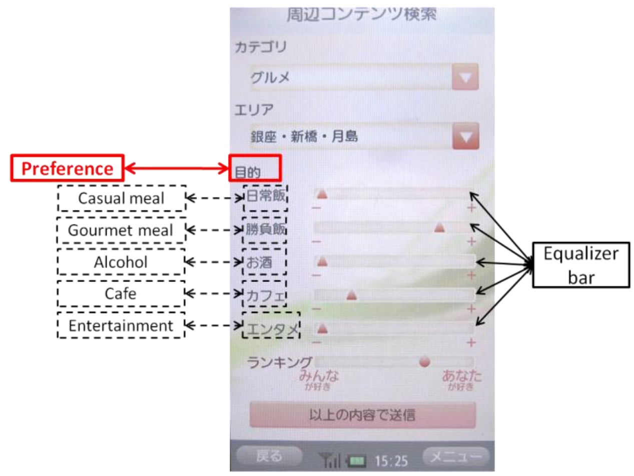

Figure 2. Screenshot showing user-modes

2.3 Sub-recommender system (SRS)

Each SRS produces an item list from user data and item data. Recommendation service providers can use any recommender systems as SRSes. Recommended item lists of SRSes generally differ from each other. The MMRS can evaluate each SRS's effect in context-aware recommendation. In our previous study [12], we evaluated the effect of five ranking methods (content-based / profile-based / item-to-item collaborative filtering / user-to-user collaborative filtering / profile-item matching) in personalized recommendation that did not use context data. We found that users' favorite methods depended on the person. For example, men in their twenties did not prefer ranking methods using user profile data as much as females and older males.

2.4 Item selector

Item selector is used for online-processing that produces a result item list from the user-mode intensities predicted by user-mode estimator and item lists of SRSes. User-mode estimator sets the probability in accordance with Equation (1) for each SRS. When a user requests recommendation, item selector randomly selects a SRS in accordance with the SRS's probability and collects one item at a time repeatedly from its SRSes until the item selector obtains the requested number of items (Figure 3).

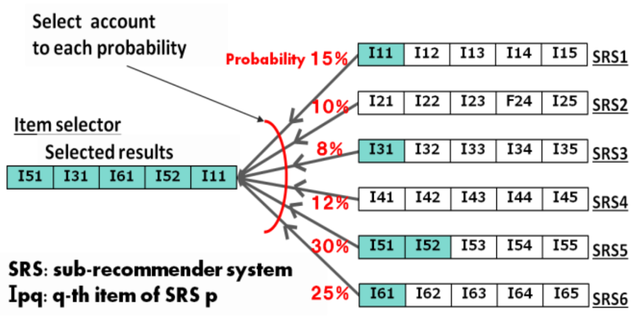

Figure 3. Item selection method in item selector

3. EXPERIMENT

In this section, we explain how we applied the MMRS to a restaurant recommendation service. Its conditions are as follows.

- Users: 2762 mobile phone users who registered online
- Items: about 28000 restaurants in Tokyo and six prefectures around Tokyo
- Recommendation requests: 65,445 times
- Experimental period: 64 days

Therefore, we use two types of positive feedback data.

- Browse: users click to see detailed restaurant information
- Bookmark: users bookmark to save restaurant information

3.1 Profile and context data

In this experiment, we chose profile/context sets with different properties (Table 1). "Age," "Gender," and "Drinker" are profile data (static-feature user data) obtained from service registration information. The others are context data (dynamic-feature user data).

"Day-type," "User-attribute," and "Next user area" in Table 1, are predicted contexts by user behavioral pattern analyses as follows. The user behavioral patterns are composed of "stop places" and "trip routes." Stop places are defined as the places where users had stayed within a 500-meters radius for more than 30 minutes. Trip routes are defined as the paths between two stop places. These are generated from location logs of a user's mobile phone. Before predicting the contexts, we estimate each stop place between a user's home, office (office / second office), and private (except the home) by comparing hours stayed at all stop places, after which we predict the contexts.

"Day-type" tells us whether the day is a workday or day off for the user. For example, Tuesday is a workday for a user if the user has often been in the office on that day. "User-attribute" shows user's current behavior from eight patterns. The attributes of the stop place or trip route determine where the user is. "Next user area" means the user's next destination. We obtained this from trip times, hour, and origin-destination history in the user behavioral patterns.

"Age×Gender" and "Day-type×Time-of-day×User-attribute" denote multiple profile/context sets. We combined "age" and "gender" because we considered that the difference in preferences between women and men in their twenties is greater than that between women and men of other generations. The predicted contexts "Day-type" and "User-attribute" depend on "Time-of-day" because behavioral pattern analyses of them relate to time.

Table 1. User profile and context information (H=6)

Category	Description
Age×Gender ($K_1 = 8$)	"Age"(-29 / 30s / 40s / 50-): four patterns "Gender" (male/female): two patterns
Drinker ($K_2 = 2$)	Whether user checks "like alcohol" in initial input data or not. (drinker / non-drinker)
Weather ($K_3 = 3$)	(sunny/cloudy/rainy or snowy) is obtained by a weather forecast service
Day-type× Time-of-day× User-attribute ($K_4 = 72$)	"Day-type" estimated from user behavioral logs (workday/day off): two patterns "Time-of-day:" (0-4 / 4-8 / 8-12 / 12-16 / 16-20 / 20-24): six patterns "User-attribute" means user's current behavior predicted from user's historical behavioral pattern. (home / office / second office / staying in private place/ commute to work / moving for work / return home / moving to private place): eight patterns
Current user area ($K_5 = 20$)	The user area in Tokyo and six prefectures around Tokyo (e.g. Ginza, Roppongi, and Chiba). This is directly obtained using a GPS system or the base stations of the mobile phone network.
Next user area ($K_6 = 20$)	Predicted area the user will go next, which is derived from the analysis of each user's behavioral patterns.

3.2 User-mode

The MMRS recommended restaurants after predicting "User-modes" (Table 2) by using the profile/context information as a result. In this experiment, we set three user-modes: "Restaurant area," "Preference," and "Ranking method." "Restaurant area" is important, especially for mobile phone users. We also consider "Preference" (types of restaurants) as the main factor for selecting items. We are also interested in "Ranking method" because the MMRS blends results of SRSes.

Table 2. User-mode information (I=3)

Category	Description	
Restaurant area ($J_1 = 20$)	The area the user prefers in Tokyo and six prefectures around Tokyo $(M_{11} \sim M_{1,20})$.	
Preference ($J_2 = 5$)	The type of meal the user prefers.	
	Casual meal (M_{21})	With price under ¥4,999
	Gourmet meal (M_{22})	With price over ¥5,000
	Alcohol (M_{23})	Bars and pubs
	Café (M_{24})	For soft drinks and light meals
	Entertainment (M_{25})	e.g. Karaoke, darts
Ranking method ($J_3 = 2$)	The method for ranking that provides recommended items.	
	Global ranking (M_{31})	Determined by popularity
	Personalized ranking (M_{32})	This scores items in accordance with how closely they match the user's stated preferences.

The MMRS had three steps in this experiment. First, it selected one "Restaurant area" (M_{1j_1}) by the highest value of Equation (1). Next, it estimated probability of "Preference" (M_{2j_2}) by

$$P\left(M_{2j_2} \middle| C_{1k_1}, C_{2k_2}, \dots, C_{Hk_H}, M_{1j_1}\right). \tag{4}$$

Finally, it estimated probability of "Ranking method" (M_{3j_3}) by

$$P\left(M_{3j_3} \middle| C_{1k_1}, C_{2k_2}, \dots, C_{Hk_H}, M_{1j_1}, M_{2j_2}\right). \tag{5}$$

Equations (4) and (5) are easy extensions of Equation (1).

Item selector stochastically selects items from SRSes. We can flexibly change user-modes and profile/context sets explicitly that are important selection criteria based on services.

4. RESULTS AND DISCUSSIONS

We elucidated the effect of each context on user-modes. As listed in Table 3, we obtained the intensities of the relationships between user-modes and profile/context sets from the Cramer's coefficient association values [4]. Value 0 means no dependency between user-modes and profile/context data, and value 1 means that the user-mode is fully predictable from the profile/context.

First, we compared the values of each profile/context set in Table 3. We found that context data, except "Weather," affected user-modes more than "Age×Gender" and "Drinker" did. Therefore, location context strongly correlates with user-modes. The results indicate that location information is more useful in predicting user-modes. The values of "Next user area" were higher than those of "Current user area" in all user-modes. The results indicate that context data computed by analyzing user's behavioral logs can improve context-aware recommendations.

- Context data (except "Weather") > Profile data
- Location data > Time data
- Next user area (predicted context) > Current user area

We compared the value of each user-mode in Table 3. For all context sets, the intensity of three user-modes is as follows.

- Restaurant area > Preference > Ranking method.

We considered "Preference" to be the most important user-mode because it is closest to user interests with regards to restaurants. Figure 4 shows the drilled-down data of the relationship between "Preference" and "Next user area." We chose this because "Next user area" has the highest coefficient (0.354) in "Preference" in Table 3. As shown in Figure 4, the MMRS predicted that many users going to Shibuya/Roppongi (popular nightlife districts in Tokyo) would be interested in "Gourmet" restaurants. Figure 4 also indicates that users going to an urban area like to browse for high-end restaurants ("Gourmet") and cafés while those going to a suburban/rural area browse for cheaper restaurants and alcohol.

In the case of only profile data, we chose the relationship between "Preference" and "Age×Gender" because "Age×Gender" has a higher coefficient (0.083) than "Drinker" in Table 3. Figure 5 shows the drilled-down data of it. As shown in Figure 5, women like to select "Cafés" and men like to select "Casual meals" (inexpensive restaurants). In addition, men become more interested in "Casual meals" as they age, and women over 50 very are interested in "Gourmet meals" (expensive restaurants). This may be due to the fact that Japanese women in their 50's tend to have a lot of discretionary money.

In this way the model learner can clarify the relationship between any user-modes and any profile/context data. The user-mode estimator predicts current user-mode intensities, and the item selector can provide recommendation items that are automatically adapted to user interests using the intensities.

Table 3. Cramer's coefficient of association between user-modes and profile/context sets

<table>
<tr><th rowspan="4"></th><th colspan="3">Profile</th><th colspan="6">Context</th></tr>
<tr><th colspan="2" rowspan="1"></th><th></th><th></th><th colspan="3">Time</th><th colspan="2">Location</th></tr>
<tr><th colspan="2">Age×Gender</th><th rowspan="2">Drinker</th><th rowspan="2">Weather</th><th colspan="3">Day-type × Time-of-day ×User-attribute</th><th rowspan="2">Current user area</th><th rowspan="2">Next user area</th></tr>
<tr><th>Age</th><th>Gender</th><th>Day-type</th><th>Time-of-day</th><th>User-attribute</th></tr>
<tr><td rowspan="2">Restaurant area</td><td colspan="2">0.142</td><td rowspan="2">0.185</td><td rowspan="2">0.064</td><td colspan="3">0.144</td><td rowspan="2">0.643</td><td rowspan="2">0.716</td></tr>
<tr><td>0.158</td><td>0.188</td><td>0.077</td><td>0.053</td><td>0.144</td></tr>
<tr><td rowspan="2">Preference</td><td colspan="2">0.083</td><td rowspan="2">0.069</td><td rowspan="2">0.048</td><td colspan="3">0.129</td><td rowspan="2">0.209</td><td rowspan="2">0.354</td></tr>
<tr><td>0.062</td><td>0.103</td><td>0.031</td><td>0.026</td><td>0.086</td></tr>
<tr><td rowspan="2">Ranking method</td><td colspan="2">0.043</td><td rowspan="2">0.020</td><td rowspan="2">0.009</td><td colspan="3">0.067</td><td rowspan="2">0.082</td><td rowspan="2">0.197</td></tr>
<tr><td>0.019</td><td>0.014</td><td>0.002</td><td>0.009</td><td>0.028</td></tr>
</table>

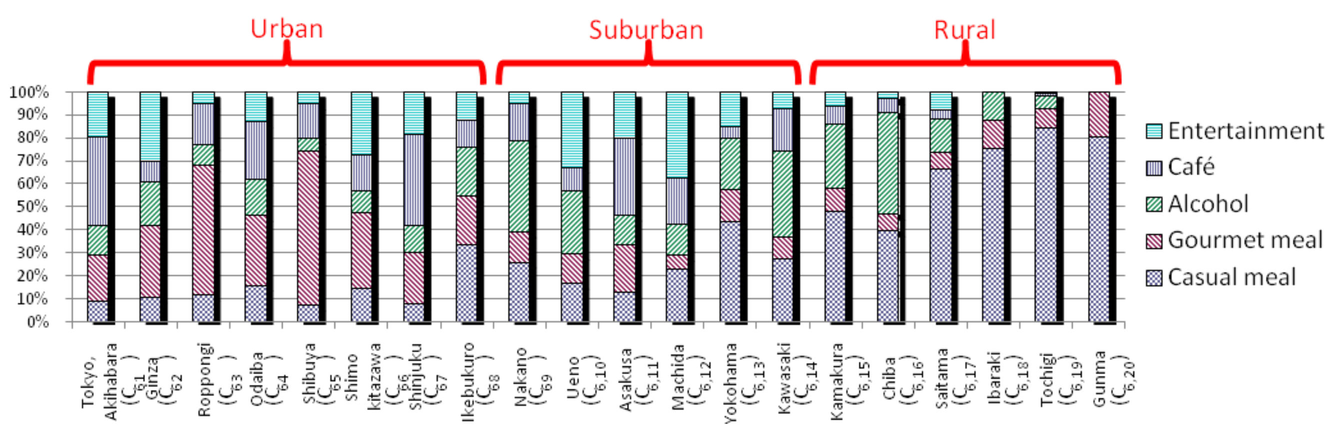

Figure 4. Relationship between user-mode "Preference" and context "Next user area" (0.354)

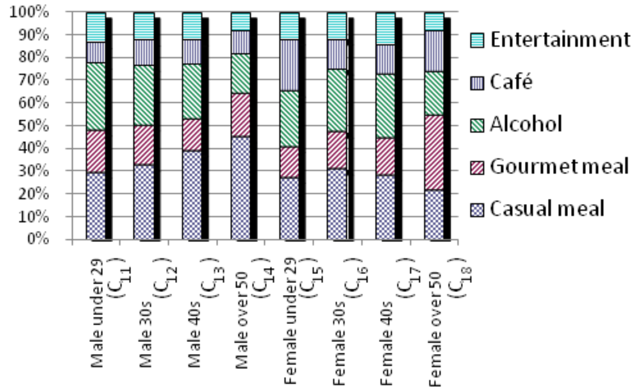

Figure 5. Relationship between user-mode "Preference" and profile "Age×Gender" (0.083)

Finally, we discuss the relationship between the machine learning mechanism of the MMRS system and user satisfaction by observing the click rate, which is defined as the number of clicked items divided by the number of retrieved items. Each plot in Figure 6 is the click rate on one day, and the line is the linear regression line. It also shows that the click rate gradually increased during the experiment. There are several possible factors for this increase.

- Machine learning makes MMRS more effective
- The ratio of active users who give a lot of clicks increases because some inactive users quit using this services

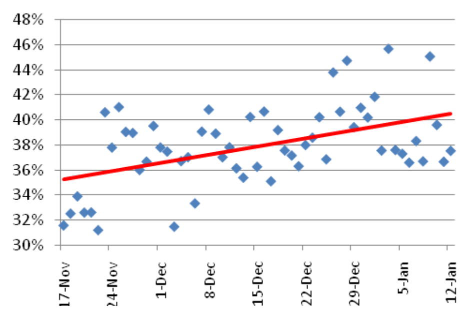

Figure 6. Click Rates

5. CONCLUSIONS

We proposed a context-aware recommender system that retrieves items and user-modes. We elucidated the effect of each context on various user preferences. We applied it to a large-scale restaurant recommendation service with 2,762 mobile phone users over a two-month period. From the results, we found that context information is more effective for making recommendations than profile information. This indicates that recommender systems using context input data can effectively improve user satisfaction. Furthermore, location data is fairly useful for predicting user intentions. User-modes depend on "Next user area" more than "Current user area". The results indicate that predicted context data computed by analyzing user's behavioral patterns can improve recommendations. Finally we discussed the relationship between the machine learning mechanism of the MMRS and user satisfaction. We found that the click rate gradually increases. However, there may be several factors for this increase. We will clarify the relationship between the MMRS and satisfaction with comparative experiments in future work.

6. ACKNOWLEDGMENTS

The research presented in this paper is part of the Information Grand Voyage Project funded by the Japanese Ministry of Economy, Trade and Industry. NTT DOCOMO produced "My Life Assist Service" [8] for the project, and we used their data. We thank Isao Kobayashi of NTT DOCOMO, Kyota Kanno, Kenshi Nishimura, Tsunehisa Kawamata, Nobuyuki Saji, and Yasuhiro Miyao of NEC for their helpful comments. We also thank all members that contributed to system implementation.

7. REFERENCES

[1] Abowd, G., Atkeson, C., Hong. J., Long, S., Kooper, R. and Pinkerton, M. Cyberguide: a mobile context-aware tour guide. Wirel. Netw. 3, 5 (1997), 421-433.

[2] Bellotti, V., Begole, B., Chi, Ed H., Ducheneaut, N., Fang, J., et al., Activity-based serendipitous recommendations with the Magitti mobile leisure guide, In Proc. of the twenty-sixth annual SIGCHI conference on Human factors in computing systems (CHI '08) , pp. 1157-1166, 2008.

[3] Cheverst, K., Davies, N., Mitchell, K., et al., Developing a context-aware electronic tourist guide: some issues and experiences. Proc. CHI'00. ACM Press, NY, 2000, p. 17-24

[4] Cramer, E. M. et al., Some Symmetric, Invariant Measures of Multivariate Association, Psychometrika, 44, 43-54. (1979).

[5] Koren, Y., The BellKor Solution to the Netflix Grand Prize, (2009). http://www.netflixprize.com/assets/GrandPrize2009_BPC_BellKor.pdf

[6] Linden, G., Smith, B. and York, J., Amazon.com Recommendations: Item-to-Item Collaborative Filtering, IEEE Internet Computing. Jan./Feb. 2003.

[7] Manning, C.D., Raghavan P. and Schütze H., Introduction to Information Retrieval, Cambridge University Press, p. 240.

[8] METI of Japan, Information Grand Voyage Project: My Life Assist Service, 2009-2010.

http://www.meti.go.jp/policy/it_policy/daikoukai/igvp/contents_en/activity09/ms09/list/personal/ntt-docomo-inc-1.html

[9] Oku, K., et al., A Recommendation System Considering Users' Past / Current / Future Contexts, CARS2010.

[10] Piotte, M. and Chabbert, M., The Pragmatic Theory Solution to the Netflix Grand Prize, (2009). http://www.netflixprize.com/assets/GrandPrize2009_BPC_PragmaticTheory.pdf

[11] van Setten, M., Pokraev, S., Koolwaaij J., Context-Aware Recommendations in the Mobile Tourist Application COMPASS. In Adaptive Hypermedia 2004, vol. 3137 of LNCS, p.235-244, 2004.

[12] Shiraki, T., Kanno, K., Nishimura, K. and Kawamata, T., Evaluation Methods for a Multi-Mode Recommendation System, 71th IPSJ Conference 2009 (in Japanese).

http://www.ipsj.or.jp/01kyotsu/award/taikai_yushu/71award_paper/2C_4.pdf

[13] Töscher, A., Jahrer M. and Bell, R., The BigChaos Solution to the Netflix Grand Prize, (2009).

http://www.netflixprize.com/assets/GrandPrize2009_BPC_BigChaos.pdf