Proceedings of the International MultiConference of Engineers and Computer Scientists 2015 Vol I,
IMECS 2015, March 18 - 20, 2015, Hong Kong

Extraction of User Context for Recommendation from Access History

Takaaki Idera, Kenji Nakamura, and Shigeru Oyanagi

Abstract—**Recommendation system is widely used to provide favorite information for a user. Current works show that the quality of recommendation can be enhanced by using user context such as time, location and situation. However, preparing enough amount of contextual information is not easy. The method to generate user context systematically is neccesary.**

This paper proposes a method to extract user context from calendar information and WWW access history for recommending restaurants. Five elements of user context such as time, location, situation, category, and budget are extracted systematically. The proposed method uses SVM(support vevtor machine) to associate access history into situation. The accuracy of the proposed method is evaluated from various aspects by experiment. The result shows the effectiveness of this approach. The proposed system is implemented on the Android, and shows the useful recommendations for users.

Index Terms—**Recommendation, Access History, User Context, SVM, TF/IDF.**

I. Introduction

The explosion of information on the web has made it difficult for users to find relevant information quickly. Recommendation system is proposed as a solution to help users to access relevant information by providing a list of recommended items that match user's preference. Nowadays, recommendation systems are widely used in practical applications.

Conventional recommendation systems deal with applications having only two types of entities, users and items, and do not take into consideration any contextual information such as time and place. Since the user's preference differs by the context, recommendation systems must take into account contextual information to satisfy users.

Several researches have been performed for recommendation systems that handle contextual information[1],[2],[6]-[12], however they must prepare substantial amount of contextual information beforehand. Hence, it is laborious for users to prepare contextual information. This paper aims to extract contextual information without user's labor in the situation of recommending restaurant. It extracts contextual information from the calendar information and web access history.

The rest of this paper is organized as follows. Section 2 explains the background of this work. Section 3 gives an outline of the proposed method. Section 4 shows the evaluation of the proposed method from several aspects. Section 5 describes the implementation of the proposed system. Section 6 concludes this work.

T.Idera is with the Guraduated School of Information Science and Engineering, Ritsumeikan University 1-1-1 Nojihigashi, Kusatsu, Shiga, 525-8577 Japan (e-mail:is0004pe@ed.ritsumei.ac.jp)

K.Nakamura is with the Faculty of Information Management, Osaka University of Economics 2-2-8 Osumi, Higasiyodogawa, Osaka, 533-8533 Japan (e-mail: k-nakamu@osaka-ue.ac.jp)

S.Oyanagi is with the College of Information Science and Engineering, Ritsumeikan University 1-1-1 Nojihigashi, Kusatsu, Shiga, 525-8577 Japan (e-mail: oyanagi@cs.ritsumei.ac.jp)

II. Background

Advances of mobile device has accelerated the research of context-aware system. Context-aware system is defined that it can provide relevant service that matches to the context without user's input. Context is initially defined as a location of the user that can be obtained from GPS. Other factors have been added subsequently. For instance, Brown[14] added date, season, and temperature. Recently, user profile is added to the context, and context-aware recommendation system has been studied. PILGRIM[13] is a mobile recommendation system to recommend a Web page that matches to the user's location and profile. COMPASS[7] is a mobile recommendation system implemented on the open platform WASP.

Oku[9] incorporates additional contextual dimensions such as time, companion, and weather into the recommendation process, and uses machine learning techniques to provide recommendations in a restaurant recommender system. They empirically show that the context-aware approach significantly outperforms the corresponding non-contextual approach in terms of recommendation accuracy and user satisfaction with recommendations. However, the common drawback is that it is not easy to prepare contextual information carefully beforehand. In this article, extracting user context from calendar and WWW access history is proposed.

III. Proposed Method

A. User Context

This paper aims to extract user context for restaurant recommendation. The target user of the proposed system is a user who needs a restaurant recommendation during his trip. It assumes that the user uses calendar to note the trip plan, and he accesses internet to search favorite meal. The aim of the proposed system is to extract user context for restaurant recommendation from the information on the calendar and WWW access history.

The information extracted from calendar consists of title, start time, and location. The WWW access history information consists of title, date, and URL. The keyword which is used at the WWW search is stored in the title of access history.

We select five elements (time, location, situation, category, and budget) for user context, which are commonly used in famous restaurant search sites in Japan such as gurunabi[3], or tabelog[4]. These elements of user context are extracted as follows.

- Time: Time is important for recommending restaurant because the menu changes by the time. Time can be extracted from the calendar information.

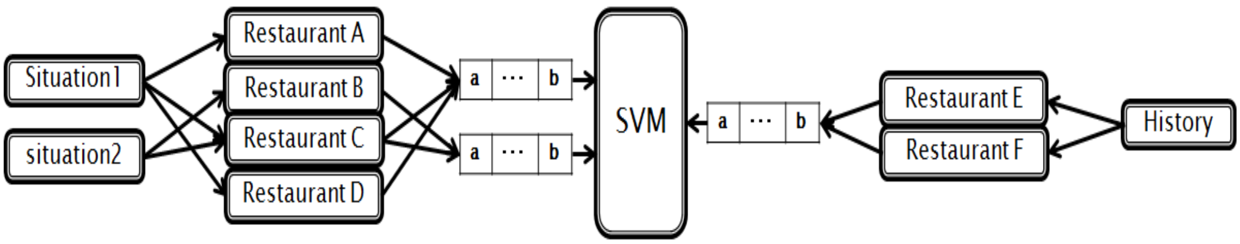

Fig. 1. feature vector generation

TABLE I
SITUATION LIST

(1)friend	(4)party
(2)couple	(5)family
(3)business	(6)single

TABLE II
CATEGORY LIST

(1)Japanese	(4)Ethnic	(7)Casserole	(10)Cafe
(2)Western	(5)Curry	(8)Pub	
(3)Chinese	(6)Korean	(9)Noodle	

- Location: Location is important especially in the case of trip. User wants to select local restaurants on the trip. Location is extracted from the calendar information. There are several notations to represent the location. The proposed system uses latitude and longitude for location notation. If a name of a place is written on the calendar, it is converted to latitude and longitude by geo-coding.
- Situation: Even if the time and the location is the same, the user may want to eat different foods according to the situation. For example, recommended restaurant should be different between formal meeting and family party. Hence, recommendation must consider the situation of restaurant utilization. The situations are listed in Table I. The appropriate situation is selected by analyzing the words included in the WWW access history. The detail of this process is explained in the next subsection.
- Category: Restaurants can be classified into several categories as shown in Table II. The appropriate category is selected by analyzing the words included in the WWW access history.
- Budget: Budget for meal can be estimated by the average of prices of restaurants which are included in the WWW access history.

B. Determination of Situation

In general, a restaurant can be utilized for several situations as shown in Table I. The relation between each restaurant and situation is combined at first. Determination of utilization situation is performed by the following method as shown in Fig.1.

1) Feature words are extracted from restaurant pages which are associated to each situation, and feature vector of each situation is generated.
2) Feature words are extracted from restaurant pages which are included in the access history, and feature vector of the access history is generated.
3) Compare the feature vector of the access history with that of each situation.

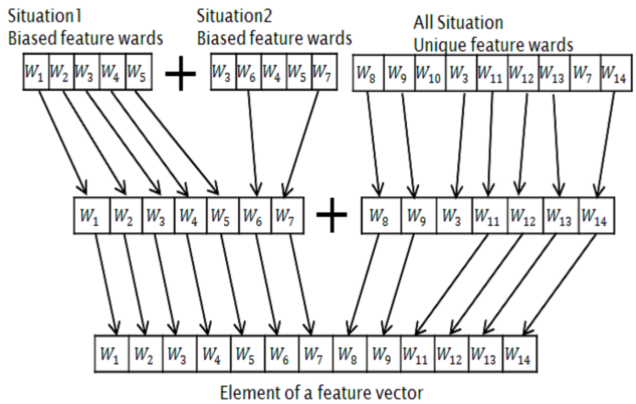

Fig. 2. Element of a feature vector

Each step is explained in more detail.

TF/IDF method is used to extract feature words. TF/IDF is a well-known method to extract feature words among a set of documents. Expression 1 shows the TF/IDF calculation.

$$TF \times \log \frac{|D|}{IDF} \quad (1)$$

TF(Term Frequency) shows the frequency of words in a document, and IDF(Inverted Document Frequency) shows inverted value of the frequency of words among a set of documents, and D is total number of documents.

Feature vector is composed of two sets, one for the biased feature word of each situation, and the other for the unique feature word of situation as shown in Fig.2. They are extracted as follows.

- Biased feature word: The biased feature word includes frequently used words in each situation. It is aimed to reduce bias of vector element by using general words. TF/IDF is calculated for each situation. A document corresponds to a page of restaurant in all situations. TF is set as a frequency of a word W in a restaurant page. D is a total number of restaurants associated to the situation, IDF is a ratio of restaurants that include word W within the total restaurants associated to the situation. The topmost 514 words for each situation are selected as an element of feature vector.
- Unique feature word: This set includes unique feature words for each situation. It is aimed to compare the

ISBN: 978-988-19253-2-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

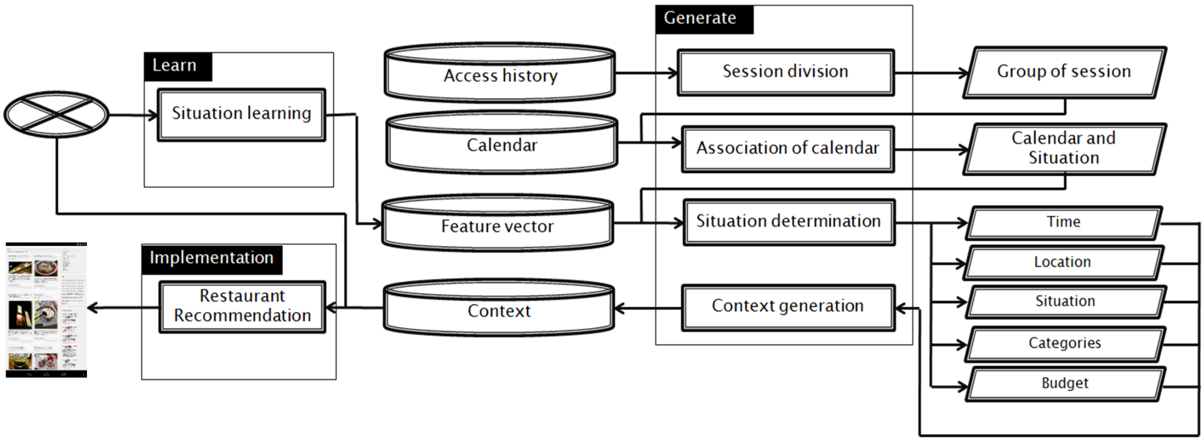

Fig. 3. Flow of the system

feature between situations. TF/IDF is calculated for total situation. All restaurant pages associated to a situation correspond to a document. Hence, there are 6 documents. TF is a frequency of a word in a document, D is six, IDF is the number of situations that the word W is included. The top-most 3084 (514 $\times$ 6) words are selected as an element of feature vector.

Biased feature words and unique feature words are combined as shown in Fig.2. The duplicated words are unified, and the result is used as the feature vector for classifying situation by SVM[5].

Feature vector for each situation is calculated by the average of the frequency of the word in corresponding restaurant pages. Feature vector of access history is calculated by the average of the frequency of the word in corresponding restaurant pages. Number of labels to classify is set to 6 as same as a number of situations. In addition, radial basis function kernel (RBF kernel) is used in this study. The kernel parameter gamma is set to 1, and the weight of C is set to 10. Feature vectors of each situation is used for learning data, and feature vector of access history is used for evaluation data. By this comparison, each feature vector of access history can be classified into one of the situations.

C. *Flow of the system*

Fig.3 shows the flow of the proposed system. Each component is explained below.

- Situation learning: It generates feature vector for each situation by feature words included in the corresponding restaurant page.
- Session division: It divides WWW access history into sessions. Fig.4 shows an example of session division. WWW access history consists of page title, time, and referrer. The process of session division is performed in three steps.
  - At the first step, access history is divided into groups by the accessed time. Successive accesses within a threshold interval are grouped together.
  - At the second step, each group is further divided based on the search and its result viewing. In Fig.4, the top of session group is WWW search.
  - Finally at the third step, groups are combined to form a grouped session based on the similarity of keywords. In Fig.4, first and second session is combined by using similar woads.
- Association of calendar: it extracts location information from calendar, and associates them to the grouped session. The matching is done by comparing location information on the calendar with keywords involved in a grouped session.
- Situation determination: It extracts user contextual information of location, time, category, and budget. The situation is extracted by the method described in the previous subsection.
- Context generation: It stores the extracted contextual information.
- Restaurant Recommendation: It searches restaurants by the context, and sends the result to the Android.

IV. Experiment

A. *outline of the experiment*

The following experiments are executed in order to determine appropriate parameters for the accuracy of the proposed method.

- Size of learning data
- Number of restaurants involved in the access history.
- Effect of session grouping
- Evaluation of the total system

In the experiment, data size of access history is 13,580 with respect to 100 trip plans.

The procedure of experiment is as follows.

- trip plans are written into the calendar.
- Information of the trip is gathered through internet search. At least one restaurant page is included in the history.
- User context is built by the system.
- The resulted context is evaluated by the user himself.

B. *Result of Experiment*

1) *Experiment on the size of learning data:*

The size of learning data is set between from 300 to 1200. Fig.5 shows the result. X-axis shows the size of learning data, and Y-axis shows the accuracy.

Large size of learning data is necessary to generate appropriate feature vector. It is saturated at 1000, hence 1000 is the best.

ISBN: 978-988-19253-2-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

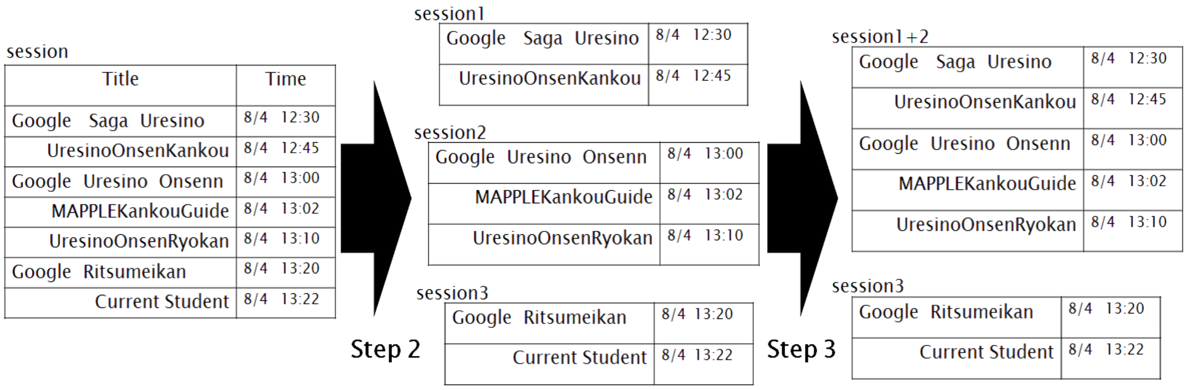

Fig. 4. session division

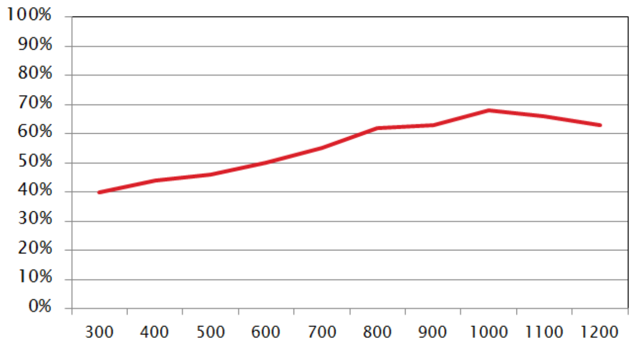

Fig. 5. evaluation of the size of learning data

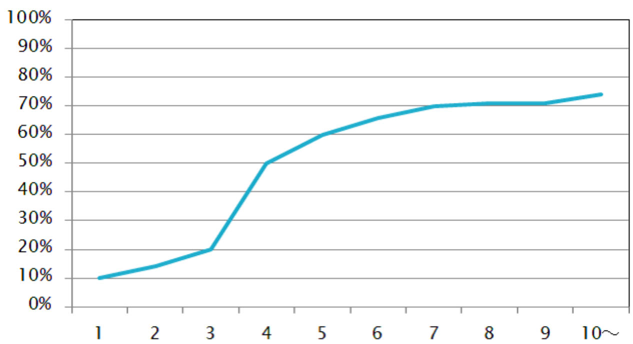

Fig. 6. evaluation of number of restaurants

2) *Experiment on the number of restaurants:* Fig.6 shows the result. X-axis shows the number of restaurants involved in the access history, and Y-axis shows the accuracy. It shows that the case of bigger number of restaurants performs better accuracy. It is because restaurant pages included in the access history are used to determine the feature vector, which is matched with the feature vectors of the situations. The result shows that the larger number of restaurants is necessary to increase the accuracy. Hence, the user should view multiple restaurants for better recommendation.

3) *Experiment on session grouping:* Fig.7 shows the result. X-axis is a plan, and Y-axis is number of restaurants. It is shown that session grouping increases the number of restaurants in a session. By combining the result of Fig.6 and Fig.7, it is shown that session grouping can contribute to improve accuracy of extracting user context.

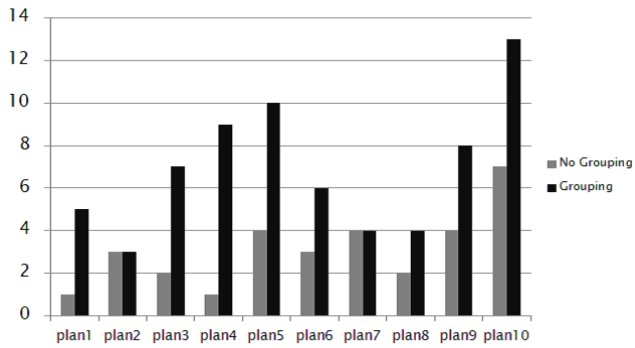

Fig. 7. evaluation of session grouping

TABLE III
EVALUATION OF ACCURACY ON TOTAL SYSTEM

	time	location	situation	category	budget
Grouping	100%	100%	68%	72%	81%
No Grouping	100%	100%	52%	62%	76%

4) *Evaluation of the total system:* We fixed the threshold size of learning data to 1000. Then the total proposed system is evaluated. The result is shown in Table III.

Table III shows the accuracy of extracting user context by the total system for each item of user context, namely time, location, utilization situation, category, and budget. The value shows the accuracy in percentage. The upper case is the experiment with grouped session, and the lower with separate session. It is apparent that the accuracy of grouped session is better than the separate session. It shows the effectivity of session grouping. With respect to time and location, the proposed system always extracts correct information because they are written in the calendar. On the other hand, accuracy of utilization situation, category, and budget is not high enough. In order to improve the accuracy of utilization situation, extraction method of feature words should be improved. In order to improve the accuracy of category and budget, generation of feature vector from access history should be improved, because of noisy page view.

ISBN: 978-988-19253-2-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Fig. 8. Display image of recommendation

Fig. 9. Display image of map

V. Implementation

A recommendation system based on the proposed method is implemented on Android smart phone. The system utilizes WWW access history after the user writes a trip schedule on the calendar to extract user context. It also updates user context whenever the user accesses to the WWW. The system recommends a list of restaurants based on the user context. Restaurants are searched from the restaurant search site tabelog[4]. Recommendation is done when the time specified at the context, or the current location is within 1 kilometer to the recommended restaurant. Display image of recommendation is shown in Fig.8. It displays a picture and explanation of the recommended restaurant, and jumps to the restaurant page when he touches the object. It also shows the current location and restaurant location on the map as shown in Fig.9. The system is practically used with user satisfaction.

VI. Conclusion

This paper proposed a mechanism to generate user context for restaurant recommendation. The contextual information is extracted from calendar and WWW access history. The proposed method uses SVM to classify utilization situation of the user, and recommends appropriate restaurants based on the extracted context. The experiment shows that the accuracy of extracting user context is high, and the result of recommendation gets user satisfaction. The proposed method is implemented on the Android. Further work is necessary to improve the accuracy of extracting user context.

References

[1] Oku Kenta, A recommendation method considering user's contexts, NAIST-IS-DD0661004, 2009.(Japan)
[2] Ishida Shogo, Kawaguchi Nobuo, Context-Aware Information Recommendation Method using Personalized Server System, DICOMO2009, nformation Processing Society of Japan, pp.1162-1170, 2009.(Japan)
[3] gurunavi, http://www.gnavi.co.jp/top/
[4] tabelog, http://tabelog.com/
[5] Corinna Cortes and Vladimir Vapnik Support-Vector Networks. MachineLearning,Vol.20,No.3,pp.273-297,1995.
[6] Shogo Seki, Shinsuke Nakajima Jianwei Zhang Information Recommendation System Considering Users'Contexts of Using Items,DEIM Forum 2011 B1-1.(Japan)
[7] Mark van Setten, Stanislav Pokraev and Johan Koolwaaij：Context-Aware Recommendations in the Mobile Tourist Application COMPASS, Adaptive Hypermedia and Adap-tive Web-Based Systems, Lecture Notes in Computer Sci-ence, Vol.LNCS3137, pp.235-244 (2004).
[8] Gediminas Adomavicius, Alexander Tuzhilin Context-Aware Recommender Systems, Recommendation Systems Handbook, Springer, pp.217-253, 2011
[9] Kenta Oku, Shinsuke Nakajima, Jun Miyazaki, Shunsuke Uemura, Hirokazu Kato, A RecommendationMethod Considering Users' Time Series Contexts, Proc.of 3rd ICUIMC-09, 2009
[10] Zhi-mei Wang, Fan Yang, An Optimized Location-based Mobile Restaurant Recommend and Navigation Systems, WSEAS Trans Inf Sci Appl, Issue 5, Vol.6, pp.809-818, May 2009
[11] Moon-Hee Park,Han-Saem Park, Sung-Bae Cho, Restaurant Recommendation for Group of People in Mobile Environment Using Probabilistic Multi-criteria Decision Making, APCHI 2008, LNCS 5068, pp.114-122,2008
[12] Gediminas Adomavicius,Bamshad Mobashe, Francesco Ricci and Alex Tuzhilin,: Context-Aware Recommender Systems, AI Magazine, pp.67-80, Fall 2011
[13] Mauro Brunato and Roberto Battiti,PILGRIM: A Location Broker and Mobility-aware Recommendation System, PerComm 2003
[14] B.Brown, M.Chalmers, M.Bell, M.Hall, J.MacColl, P.Rudman, Sharing the square: collaborative leisure in the city streets, 9th Conference on European Conference on Computer Supported Cooperative Work, pp.427-447, Springer, 2005

ISBN: 978-988-19253-2-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)