A Model for Proactivity in Mobile, Context-aware Recommender Systems

Wolfgang Woerndl[1] Johannes Huebner[1] Roland Bader[1,2] Daniel Gallego Vico[1,3]

[1]Technische Universitaet Muenchen
Boltzmannstr. 3
85748 Garching, Germany
+49 8928918686
{woerndl,huebner}@in.tum.de

[2]BMW Group Research & Technology
Hanauerstraße 46
80992 Munich, Germany
+49 38241909
roland.bader@bmw.de

[3]Universidad Politécnica de Madrid
Avenida Complutense 30
28040, Madrid, Spain
+34 629866417
dgallego@dit.upm.es

ABSTRACT

A proactive recommender system pushes recommendations to the user when the current situation seems appropriate, without explicit user request. This is conceivable in mobile scenarios such as restaurant or gas station recommendations. In this paper, we present a model for proactivity in mobile recommender systems. The model relies on domain-dependent context modeling in several categories. The recommendation process is divided into two phases to first analyze the current situation and then examine the suitability of particular items. We have implemented a prototype gas station recommender and conducted a survey for evaluation. Results showed good correlation of the output of our system with the assessment of users regarding the question when to generate recommendations.

Categories and Subject Descriptors

H.3.4 Systems and Software, H.5.m [Information Interfaces and Presentation]: Miscellaneous, H.3.3 Information Search and Retrieval

General Terms

Algorithms, Experimentation, Human Factors

Keywords

Recommender System, Proactive System, Proactivity, Context, Mobile, Information Filtering

1. INTRODUCTION

Traditional recommender systems usually follow a request-response pattern, i.e. these systems only return item suggestions when a user makes an explicit request. In mobile recommender systems, users cannot browse through many search results and suffer from other restrictions in the user experience, because of limitations in the user interface such as small display sizes or missing keyboards. In mobile environments, having the user not to submit any request or query to get a recommendation could possibly improve the user experience.

Consider the following two scenarios. A mobile restaurant guide running on a smartphone suggests a restaurant to the user when s/he is walking near the restaurant that fits his/her preferences very well, while also factoring in that it is lunch time right now. The second scenario is a gas station recommender built in the navigational system of the car. In this case, the system could proactively suggest a gas station, when the remaining fuel level is nearly empty and a gas station is nearby, without much or any detour.

Proactivity means that the system pushes recommendations to the user when the current situation seems appropriate. Proactive recommender systems have to not only determine *which* item(s) to recommend, but also *when* to make a recommendation. In this paper, we present a model for proactivity in mobile, context-aware recommender systems. To do so, we first review related work. Section 3 then explains our proposed model. In Section 4, we describe a prototype implementation of the gas station scenario that has been evaluated in a user study. We finally conclude the paper with a short summary and outlook on future work.

2. RELATED WORK

A large amount of research and practical applications exist on recommender systems, mobile computing, context-awareness (see e.g. [2]) or location-based services, as well as any combination of the above areas. Kenteris, Gavalas and Economou recently surveyed the field of mobile guides [6], for example. However, proactivity has not gained much attention in personalization and recommender system research or has been put into practice. Most systems require the user to perform some kind of action to trigger the generation or retrieval of recommended items.

The Hippie/HIPS system was an early mobile guide system exhibiting proactive behavior [7]. The approach offered guidance by using maps and indoor positioning technology at room and object level. The system proactively notified users of upcoming exhibits and sights.

More recently, Hong et al. [4] proposed an agent-based framework for proactive personalization services. The approach creates a model according to which a user profile is deduced from a user's context history. The model enables proactive recommendations in the future. However, training time is very important in the proposed model.

Ricci discusses proactivity in mobile recommender systems in his recent survey [8]. Some systems make use of the current user behavior, position and other context information to improve personalization on mobile devices and in ubiquitous computing in general. Ricci concludes that "none of the existing reviewed systems is capable to proactively interrupt the user activity with unsolicited but relevant recommendations" and "[proactive recommendations] can revolutionize the role of RSs from topic oriented information seeking and decision making tools to information discovery and entertaining companions" [8].

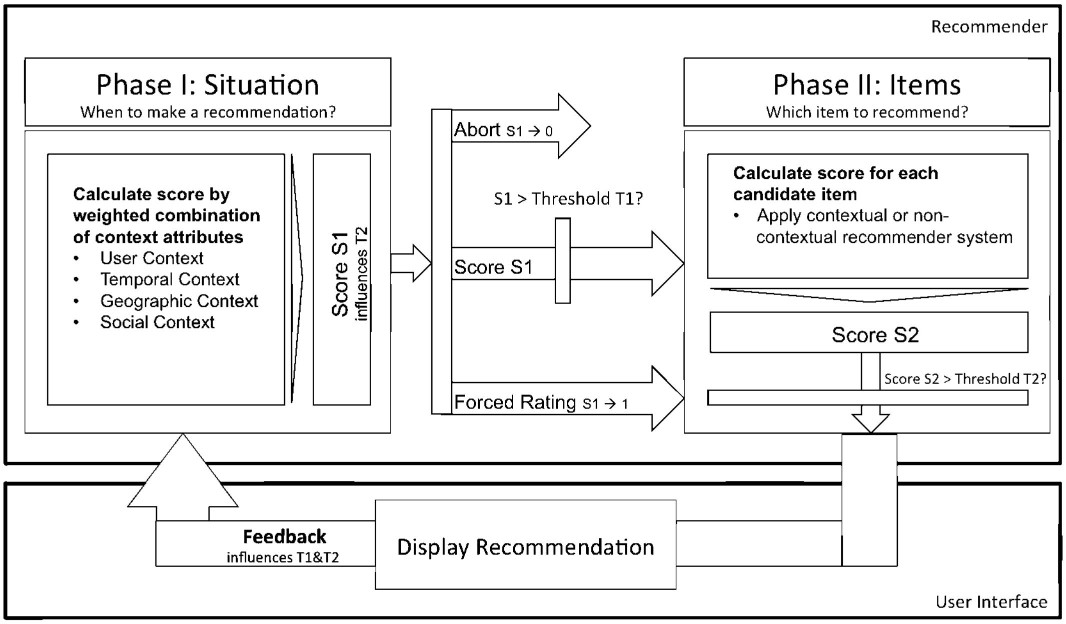

Figure 1. Proposed two-phase proactivity model

3. INTRODUCING PROACTIVITY INTO MOBILE RECOMMENDER SYSTEMS

Our model for proactivity incorporates contextual information to evaluate whether a recommendation is suitable in a given situation. We first define what we mean by context and then develop our model.

3.1 Context

We follow the definition of Dey et.al. [3]: "Context is any information that can be used to characterize the situation of entities (i.e. whether a person, place or subject) that are considered relevant to the interaction between a user and an application, including the user and the application themselves". [3] identifies location, identity, activity, and time as most important context dimensions. Based on this classification and an analysis of the two previously described scenarios, we identified the following four context categories as main influence factors for our model: 1. User context, e.g. the remaining fuel level in the gas station scenario, 2. Temporal context, e.g. current time for restaurant scenario, 3. Geographic context, e.g. distance of available points-of-interests, and 4. Social context, e.g. whether the user is alone, or walking or driving with other people.

In each of the categories, several context attributes can be modeled and utilized. The attributes are evaluated using a score on a range from 0 to 1 in our model. The higher the score for a context attribute, the higher the indication that a proactive recommendation could be useful. For example, if the remaining fuel level is nearly empty, the corresponding attribute will be close to 1. Each attribute is weighed depending on the relative importance of the parameter to the recommendation process in a given scenario. The context attributes and weights are domain-dependent and have to be predetermined or learned for every application scenario. We explain more specific examples as part of our prototype implementation in the gas station scenario, see chapter 4.1. below.

3.2 Two-Phase Process Model

We propose a two-phase model for utilizing the available context information in a proactive, context-aware recommender system (Figure 1). In the first phase, the system determines whether or not the current situation warrants a recommendation. The second phase deals with evaluating the candidate items. If one or more items are considered good enough in the current context in the second phase, the recommender system communicates it to the user. The point of time when the system displays the recommendation does not necessarily have to be immediately after the recommendation generation. In the gas station scenario, the in-car navigational system could delay the recommendation until the car is not moving, e.g. waiting at a red light.

The first phase is executed periodically in the background. The second phase is only executed when the first phase indicates a promising situation. Both phases utilize several context parameters in the four context dimensions as defined above.

3.3 Phase I: Situation Assessment

In the first phase, the system needs to determine whether or not the user should receive a recommendation. To do so, the system calculates an overall score S1. S1 is a number between 0 and 1. The first phase can have various outcomes in principle (cf. Figure 1), based on the value of S1.

If S1 exceeds a threshold T1, the second phase will be initiated. If S1 = 1, the highest possible value, then a recommendation will be triggered in any case. An example is when the remaining fuel level of the car is very low and there is only one gas station in the fuel range. In this case, a recommendation for the only option should be triggered no matter how pricey this gas station is. Note that the first phase does not take properties of single items into account, but does consider general properties of the set of candidate items, e.g. number of reachable gas stations as part of the geographic context.

If the current situation does not warrant a recommendation, no matter how high a particular item would score, S1 is set to 0 and the recommendation process is aborted without considering items for recommendation. An example is a situation with a fuel level of nearly full in the gas station scenario.

Furthermore, the score S1 has an impact on the threshold T2 of the second phase T2. If the situation for a recommendation is perfect, T2 will be decreased in order to make it more likely that at least one item score S2 in the second phase reaches the required threshold and will be recommended to the user. Following this, the threshold T2 is a function of S1, in the simplest form:

T2 = | 1 - S1 |

If the situation assessment in phase I indicates a high relevance (e.g. fuel level is low), the score S1 will be high, phase II will likely be initiated and the threshold T2 in phase II will be lower. On the other hand, if the situation assessment leads to a mediocre score S1, phase II might still be initiated but only an extraordinary high rated item might score good enough to be recommended.

Concluding, the first phase can trigger an abort of the process, force a recommendation or yield to a score S1 between 0 and 1 that either initiates phase II or not, depending on the threshold T1. The threshold T1 has to be configured a priori and can be adjusted according to user feedback.

3.4 Phase II: Item Assessment

The second phase evaluates the suitability of particular items. To do so, any recommender algorithm can be used. The result of phase II is a score S2 for each item in the candidate set. S2 corresponds to the predicted rating of standard collaborative filtering. S2 is again normalize to the [0,1] range, with S2 = 1 being the best possible score. Similar to phase I, each item can be immediately eliminated from the recommendation process if triggered so by certain criteria (e.g. if a restaurant is closed). In this case S2 is 0 and this item won't be considered in any case.

The items will be ranked according to S2 and tested against the threshold T2. If S2 > T2, then this item is finally considered for recommendation. Depending on the application scenario, the best item, or k best items, above the threshold will be displayed. If no item score exceeds the threshold, then no item is recommended, the process is aborted and restarted with phase I at the next configured interval. If a forced recommendation was triggered in phase I, the best-ranked item will be proposed in any case, since the threshold T2 is 0.

To determine the score S2, it is possible to combine contextual and non-contextual recommenders in a hybrid approach [10]. In this case, the system calculates a "ctx-score" by evaluating the item against context attributes and categories, as explained above. In addition, a collaborative filtering algorithm predicts the rating of an item factoring in ratings of other users ("cf-score"). The integration can be then done as a linear combination with a weight w for instance: score = w * cf-score + (1-w) * ctx-score.

3.5 User Feedback

After the recommended item(s) is/are communicated to the user, the user can optionally give some feedback on the recommendation. This is done in particular with regard to the point of time of the recommendation, for example: "I don't want a recommendation now". The feedback influences the thresholds T1 and T2. When the user does not like the point of time of the recommendation, the thresholds are increased to decrease the chance of a proactive recommendation in the future. The feedback process is similar to critique-based recommender systems that have been proposed and used in other mobile recommender systems [8]. The actual user interface is out of the scope of the paper.

4. IMPLEMENTATION AND EVALUATION

4.1 Prototype Implementation of Gas Station Scenario

In a next step, a prototype for the gas station scenario was implemented to simulate phase I of the presented model. The considered criteria, their weights and the overall score calculation is presented in more detail in [5]. The parameters were in part derived from an earlier study investigating the influencing factors of gas station selection such as fuel level, detour and total length of route [1]. Due to limited space, we only give a few examples of the context model.

Fuel level is part of user context with a relatively high weight. Any fuel level above 50% results in a score of 0 for this attribute, a fuel level of 25% results in 0.5, and empty in 1. The user context also considers the average remaining fuel level when the user was refilling in the past. Traffic is part of the temporal context. In this case, the system evaluates the current speed and the route ahead as given from the navigational system. A current speed of 0 km/h results in a score of 1, because the situation of a waiting car should lead a higher probability of recommendation generation.

Minimum detour, i.e. detour to the nearest gas station from the current route, is an example for geographic context. This attribute is also dependent on the total length of route since drivers are assumed to try to avoid long detours on short routes. Finally social context in this gas station scenario corresponds to the number of persons in the car at this time, to avoid having passengers wait while refueling. This social context category is assigned a low weight in the gas station example but could be of more importance in other scenarios.

4.2 Evaluation

We evaluated the implementation of our introduced model in the scenario of the gas station recommender. For this purpose, we set up an online questionnaire. The evaluation was targeted at the first phase of the suggested model, i.e. the point of time when to make recommendations. The second phase was neglected, since it was not the focus of this paper. Therefore, the result of the prototype system is the score S1.

35 participants took part in our study. First, all participants were asked to enter demographic data. In the subsequent section, 13 scenarios were constructed in each of which the test subjects were asked to rate on a 5-point Likert scale (1 = very bad, 5 = very good), how useful and convenient they would find a recommendation made by the navigation system in each case. The ratings were rescaled to the [0,1] range to fit our model.

For example, scenario 8 read as follows: "You drive home on a 10 km route with little time pressure. Your remaining fuel level is ¼ and the detour to a gas station is 1.5 km". In all scenarios, the fuel level was assumed to be rather low, so that a recommendation is a possibility. The parameters for each scenario were used as input into the prototype system to find out in which cases the proposed model would lead to a recommendation. Afterwards, the prototype's decisions were compared to the assessment of the test users.

Although the recommender algorithm is pretty straightforward and rather simple, the online questionnaire showed that it yields good results (Table 1). In most scenarios the results are reasonably close. In some cases, e.g. scenario 3, the system could not quite predict the assessment of users. It seems that real drivers give more weight to the fuel level and only factor in rather few context information. A more detailed discussion of the results can be found in [5].

Table 1. Evaluation results

Scenario No.	Prototype result	Evaluation MEAN (μ)	Evaluation STD (σ)
1	0	0.39	0.23
2	0.53	0.74	0.26
3	0.46	0.80	0.20
4	0.51	0.74	0.28
5	1	0.95	0.10
6	0.46	0.63	0.29
7	0.57	0.68	0.25
8	0.39	0.46	0.25
9	0.64	0.86	0.23
10	0.79	0.57	0.27
11	0.50	0.49	0.25
12	0.72	0.80	0.23
13	0.53	0.76	0.23

This was also reflected in the answers provided in a free input field regarding the question of how often recommendations should be made. Instead of naming frequencies, most participants entered that they wish to receive suggestions on the event that their fuel level reaches a certain threshold, e.g. ¼ of the fuel level, or the remaining fuel range less than e.g. 100km. This corresponds well to our general model with a score of the situation is matched against a threshold.

The goal of the last part of the questionnaire was to determine the overall acceptance and usefulness of such a recommender system. To assess these properties, a semantic differential [9] with nine adjective pairs was used. As can be seen in Figure 2, the semantic differential questions revealed a positive attitude towards such a proactive recommender system, while some users expressed mild concerns about the system being annoying or irritating.

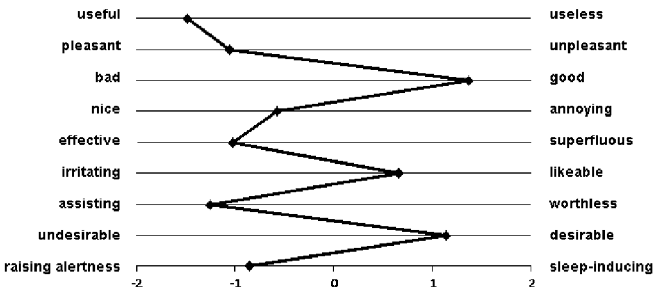

Figure 2. Semantic differential of survey results

5. CONCLUSION

In this paper, we have presented a new model for proactivity in mobile, context-aware recommender systems. The main idea is to first assess the current context to determine whether a recommendation is warranted at this point of time. In the second step, candidate items are examined to finally decide if the system should generate a recommendation and which item to suggest.

Current and future work includes refinement of the model and implementation and evaluation in more scenarios and field studies. By doing so, we cannot only evaluate the implementation of the particular scenario, but also the general model as well. Furthermore, future work is to try to learn the factors and weighs of the different context categories based on past user behavior.

One area of on-going work is the user interface of such a system, for example in the scenario of a mobile recommender system on smartphones for restaurants and other points-of-interests, because user acceptance is a crucial aspect of a proactive recommender. We have designed two options for the user interaction on Android smartphones: a Widget- and a Notification-based solution. We are currently evaluating this approach in a user study.

6. REFERENCES

[1] Bader, R., Neufeld, E., Woerndl, W. and Prinz, V. 2011 Context-aware POI recommendations in an automotive scenario using multi-criteria decision making methods. *Proc. Workshop on Context-aware Retrieval and Recommendation (CaRR 2011)*, Conference on Intelligent User Interfaces (IUI 2011), Palo Alto, CA, USA.

[2] Baldauf, M., Dushdar, S. and Rosenberg, F. 2007. A survey on context-aware systems. *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. 2, No. 4, 263–277.

[3] Dey, K., Abowd, D. and Salber, D. 1999. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications, *Human Computer Interaction*, Vol. 16, 97– 66.

[4] Hong, J., Suh, E.-H., Kim, J. and Kim, S. 2009. Context-Aware System for Proactive Personalized Service Based on Context History. *Expert Syst. Appl.*, Vol. 36, 7448–7457.

[5] Huebner, J. and Woerndl, W. 2011. *Proactivity in a Mobile, Context-Aware Recommender System.* Tech. Report TUM-I1108, Institut fuer Informatik, Technische Universitaet Muenchen, Munich, Germany.

[6] Kenteris, M., Gavalas, D. and Economou, D. 2011. Electronic mobile guides: a survey. *Personal and Ubiquitous Computing*, Vol. 15, No. 1, 97–111.

[7] Opperman R. and Specht, M. 2000 A context-sensitive nomadic exhibition guide. *Proceedings of the 2nd Symposium on Handheld and Ubiquitous Computing*, 127–149.

[8] Ricci, F. 2010. Mobile Recommender Systems. *Int. Journal of Information Technology and Tourism*, Vol. 12, No. 3.

[9] Van Der Laan, J.D., Heino, A. and De Waard, D. 1997. A simple procedure for the assessment of acceptance of advanced transport telematics. *Transportation Research Part C: Emerging Technologies*, Vol. 5, No. 1, 1–10.

[10] Woerndl, W., Muehe, H., Rothlehner, S. and Moegele, K. 2009. Context-aware recommendations in decentralized, item-based collaborative filtering on mobile devices. *Workshop on Innovative Mobile User Interactivity*, MobiCASE Conference, San Diego, USA