A User Interface Concept for Context-Aware Recommender Systems

Patrick Hiesel[1], Wolfgang Wörndl[1], Matthias Braunhofer[2], Daniel Herzog[1]

Department of Informatics, Technical University of Munich[1]
Faculty of Computer Science, Free University of Bozen - Bolzano[2]

Abstract

A context-aware recommender system incorporates the knowledge of different contextual factors - such as time or weather information - to improve the item suggestions made to a user. While this provides great benefit to users, it might be hard for them to grasp why certain items are relevant, given the complexity of a context-aware recommender. In this paper, we propose, implement and evaluate a user interface concept that seeks to tackle this challenge. We show how popularity graphs can be used to inform the user about the relevance of items in different contexts and how users perceive different contextual factors given our concept. A user study with 14 participants demonstrates that our concept is valid and appreciated by users.

1   Introduction

Recommender systems are a composition of software tools and techniques that suggest items to a user (Ricci et al. 2015). Context-aware recommenders are an advancement to traditional recommender systems by incorporating the knowledge of different contextual factors - such as time or weather information - to improve the item suggestions made to a user (Adomavicius & Tuzhilin 2015). Their primary goal is to better match users and their current context with items that are popular in the same or similar contexts. Prime examples for such systems are a recommender for restaurants that ranks beer gardens higher on sunny days in summer and a car that knows a driver's route, fuel level and gas prices to suggest gas stations.

The improved accuracy can also be used to deliver proactive recommendations when the confidence and utility is high enough (Woerndl et. al. 2011). A proactive recommender system pushes recommendations to the user when the current situation seems appropriate, without an explicit user request. Mobile environments are an appealing application area for proactive recommendations, for example, a tourist visiting a city can be provided with notifications for nearby points of interest (POIs).

Traditional recommender systems consider items liked by users in the past and some additional information such as item characteristics to estimate ratings for items (Adomavicius & Tuzhilin 2015). The impact of contextual information on rankings, however, introduces new challenges for user interface design as the relevance of items becomes less apparent to a user. This is especially true for proactive recommendations because users might not be fully aware why an item was recommended and need more information and extra explanations. Contextual recommendations could possibly increase the trust of users in the whole system as well.

In this work, we have designed, implemented and evaluated a novel user interface concept to visualize the relevance of items in different contextual situations. The goal is to support the user to make better decisions in mobile and dynamic environments. We have developed a fully functional recommender for POIs and incorporated the temporal and geographic context. A user study with 14 participants shows clear indications that our concept is valid and appreciated by users. In the following, we explore related work from industry and academia, elaborate the design, data sources and implementation of our prototype and present the results of a user study evaluating our system.

2 Related Work

Context-aware recommender systems (Adomavicius & Tuzhilin 2015), and also how to assist user interaction with them, are receiving more and more attention by both industry and academia in recent years. Hence we explore related work in this section and base our research on the previous findings.

Foursquare has established itself as a major player for context-aware recommendations over the past decade. Despite not being the major social network, Foursquare has aggregated over 50 million venues in 196 countries, 75 million comments and has 60 million registered users of which 50 million are active at least once a month (Weber & Novet 2015). It delivers contextual notifications depending on the user's current location and information about surrounding places. While Foursquare maintains a large user database that it uses for context-aware recommendations, it did not yet provide a user interface to explore the popularity of a place based on context. Google has recently launched a new feature for Google Places – Popularity Histograms – that show the crowdedness of a place as a function of the day of the week and the time of the day. This feature is similar to our research but limited to temporal context. Google Now also tries to assist the user with providing the right information at the right time. Other known systems include Yelp for restaurants, Google now tourism or Spotify for music discovery.

(Lathia 2015) gives an overview on data, algorithm and evaluation of mobile, location-aware recommender systems. Our earlier work focused on utilizing contextual information from various social media sources to find out how relevant items are for the current user context (Braunhofer at al. 2015). The work presented in this paper continues this idea but with the focus on the user interface issues of such a system. (De Pessemier et al. 2014) present a

framework to detect the current context and activity of the user and provide users with personalized content, for example POIs, train schedules and other touristic information.

South Tyrol Suggests is an application available in the Android app store that recommends POIs based on exploiting various contextual factors and a matrix factorization rating prediction model. (Braunhofer et al. 2014) reports the results from a controlled live user study as well as from the log data produced by the application to assess the usability of the app. Results indicate that the user interface is considered simple and intuitive but users did not always understand the true motivation and behavior of certain functions.

A few approaches have also dealt with proactivity in context-aware recommender systems. (Gallego et al. 2013) have evaluated the usability of widget- and notification-based user interfaces for a proactive restaurant recommender on smartphones. Test users in this study have rated proactive recommendations well if they are relevant in the current situation and delivered timely and by appropriate means. (Dali Betzalel et al. 2015) have investigated the timing of proactive recommendations using data collected from a three-week study. They try to learn a personalized model of good and bad contexts for recommendations and integrate this information for timely recommendations. But this work did not further investigate the user interface of such a proactive recommender system.

3   Prototype

Our prototype is based on a context-aware recommender system developed in a different project within our group. The system is capable of aggregating a large number of points of interest (POIs), collecting check-ins for these POIs and learning their popularities in different contextual situations using inference techniques. In a prototype setting, this system generated a knowledge base of about 200.000 POIs in the greater Munich area and is able to recommend POIs based on the user's current temporal and geographic context. On top, each POI record has information about the POIs popularity depending on timing and weather.

The focus of this paper is on the presentation of this information on smartphones.

3.1   Data Sources

The POI data was aggregated by our backend service blending together information from seven sources: Yelp, Google Places, Facebook, Quermania, Foursquare, Wikipedia and OpenStreetMaps. Following a standard data science process (Runkler 2012), the obtained information was filtered, cleaned and co-referent POIs were merged.

In a consecutive step, we aggregated check-ins from Flickr, Foursquare and Twitter and associated these check-ins with POIs using different techniques. Geo-based check-ins were associated with POIs using geofencing and distance measures. Tweets and Foursquare POIs were associated using an approach proposed by (Melià-Seguí 2012).

Popularity information for POIs was inferred using the check-ins and an activity duration depending on the POI category.

3.2 Design

A primary design goal was to visualize the contextual information in a meaningful way. We therefore rely on proven patterns like a Master-Detail navigation throughout most parts of the app and incorporate our concepts in patterns users already know.

In the design phase, we evaluated a potential implementation of our system in both a stationary (desktop) and mobile context. The knowledge base we have at hand is primarily intended to suggest restaurants, bars and sights to users. Our use cases therefore focus on on-the-go decisions like "Which restaurant should I go to?". While this use case calls for a mobile solution, others like "What sights are a good fit for a trip next weekend?" could be catered using a desktop solution.

Users are introduced to the application with a slide-based tutorial that guides them through the essential parts of the application. Upon registration or login, they are presented with a tab-based navigation - as depicted in Figure 1 - that let's them switch between Tips (recommendations), Browse (see all POIs), Notifications (check proactive recommendations) and Settings (essential app configuration). The application is fully aware of all necessary context factors, including the user's location, current weather and weather forecast as well as date and time. This information is regularly sent to the backend server to update the recommendations. Figure 1 presents the recommendations fitting the current context best.

In our earlier research, we published models for making a push/no-push decision based on the user perceived utility of a push recommendation for an item (Braunhofer et al. 2015). The utility of a push recommendation for an item is determined by two factors – preference fit (i.e., predicted rating) and context fit (i.e., suitability of a proactive recommendation given the current context) – and items with a high enough utility score will be pushed to the user as proactive recommendations. We integrated this concept into our UI and let users subscribe to notifications. Figure 2 shows the notifications page with short information about why an actively pushed item might be relevant to a user.
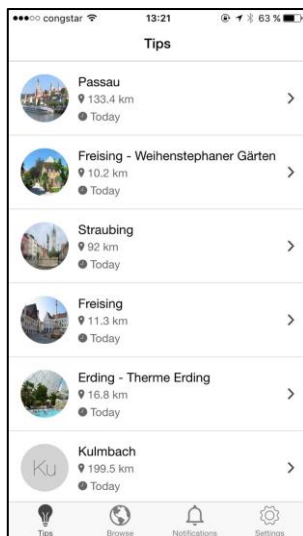

Figure 1: Recommendations

Figure 2: Notification

Figure 3: POI Description

Both Tips and Browse let users explore POIs using the recommended sorting and an alphabetic sorting respectively. We integrate our context visualizations on the details page of each POI (Figure 3-6) to give a user insight into why this POI can be relevant given their current situation.

3.3   Context Visualization

Our vision for context visualization is to denote the user's current context (time, location, weather) along with the context under which places are popular. The intention is that by showing this side-by-side comparison, it should become more apparent to users how the recommender works and why certain items are of relevance to them. In the long run, this should make context-aware recommenders more transparent and raise the user's awareness for the concept.

In the POI detail page (Figure 3), we integrated a weather forecast for possible visit times of users to show their current or future weather context (Figure 4). A map shows both the user's and the POI's location and the distance. The popularity of a POI in different contexts is essential to our concept and displayed in two different ways: First, we show a textual summary of the popularity peak for both weather and day/time (Figure 4). Second, we render line and bar charts to show the popularity across different context dimensions, such as day of the week (Figure 5), time of day or season (Figure 6).
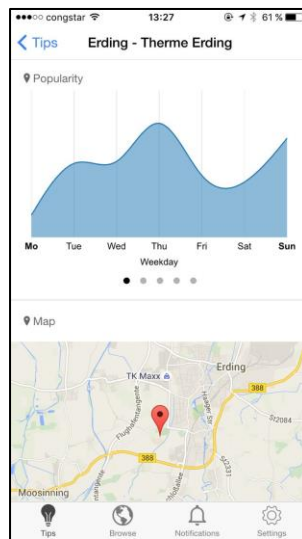

Figure 4: POI Popularity Forecast   *Figure 5: POI Popularity Graph*   *Figure 6: POI Popularity Graph*

3.4 Implementation

We implemented our concept using the Ionic Framework – a cross platform UI framework for all mayor operating systems including Android and iOS. Ionic is an open source UI framework on top of Apache Cordova and Angular.js that makes it easy to prototype apps fast and distribute them to users with different mobile operating systems.

4   Evaluation

To get feedback for our concept, we evaluated our system in a user study with 14 participants. Our test population comprised mainly of young adults aged between 23 and 30 years. Participants received the link to download the app on iOS or Android and were asked to use the app for at least two weeks. Part of the study was to test if our concept was intuitive enough to be used without further explanation.

4.1 User Survey

At the end of a two-week period, users were provided with a link to a short survey. The main part of the survey consisted of ten questions that evaluate different aspects of the system on a 5-point Likert scale:

1. The app did a good job on-boarding me as a user through a short tutorial.
2. The popularity graphs for the time of day (0-24h) were useful and would have influence on my decision to visit a place.
3. The popularity graphs for the weekday (Mon-Sun) were useful and would have influence on my decision to visit a place.
4. The popularity graphs for the weather condition (Sunny-Snowy) were useful and would have influence on my decision to visit a place.
5. The popularity graphs for the temperature (Hot-Freezing) were useful and would have influence on my decision to visit a place.
6. The popularity graphs for the season (Spring-Winter) were useful and would have influence on my decision to visit a place.
7. Did you find the app's recommendations (Tips Tab) relevant?
8. Would you use such an app to get recommendations for weekend trips?
9. Would you use such an app to get recommendations for restaurants or bars?
10. Would you subscribe to a (free) service that actively informs you about possible weekend trips or restaurants you could go to?

Table 1 shows the detailed results for all ten questions in the interval between -2 (Strongly Disagree) to +2 (Strongly Agree) along with the mean and standard deviation. The user study yielded great feedback for our system and provided evidence that the popularity graphs provide benefit for users. We have received the strongest positive evidence for the day of week popularity. In general, the temporal popularity was perceived as more interesting when

compared to the geographic popularity. The temperature popularity chart received the lowest score relative to the others.

The large majority of users would use such a system to get recommendations for bars, restaurants and weekend trips.

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Ø	σ
1	0	1	4	4	5	0,92	0,99
2	0	1	2	4	7	1,21	0,97
3	0	1	0	4	9	1,5	0,85
4	0	0	2	9	3	1,07	0,61
5	0	1	5	7	1	0,57	0,75
6	0	1	4	2	7	1,07	1,07
7	1	1	0	8	4	0,92	1,14
8	0	1	2	7	4	1,0	0,87
9	0	1	0	5	8	1,42	0,85
10	0	0	2	2	10	1,57	0,75

Table 1: User Study Results (n=14)

4.2 Open Feedback

In addition to the Likert-scale questions, we asked users to provide open feedback for the system and suggest improvements and potential areas for future work.

Positive feedback included that users were fond of the clean UI and the popularity graphs as it gave them a sense of when places are crowded. Some users noted that they liked the popularity graphs for the fact that they would get an option to choose a less crowded time to visit a place. This marks an additional benefit for our system. This information could also be used for proactive recommendations to suggest a place when it is open but not too crowded.

Negative/constructive feedback included that there should be a way to display contextual popularity information on the list views in addition to the information on the details page. Users felt it may save a lot of time while browsing places with the app and make the recommender even more transparent. On top, textual comments on why places are relevant like: "A visit to this place is recommended in the afternoon of a sunny spring day." should be added to the list view and app details page.

5   Conclusion

We designed, implemented and evaluated a user interface concept to display contextual information about places in a context-aware recommender system. Goals included making complex systems more transparent to users and giving them a tool to explore when places are popular. A user study with 14 participants indicates that the concept is valid and appreciated by users. The user study yielded great feedback for our system and provided indications that the key research area - popularity inference and graphs - provides benefit for users. Users stated that popularity graphs assisted them while deciding which place to visit. Besides find popular places, some users stated that they would use popularity graphs also for finding less crowded places.

The users study also pointed towards areas of potential future work. First, the contextual information of a place should be summarized and displayed in the tips and browse list views. On top, users would like to receive a textual explanation why a place is recommended to them. Furthermore, the contextual information of places would allow for more advanced filtering options and calls for user interface concepts to provide an intuitive way of exploring and finding using contextual filters. Future work could lie in adapting popularity graphs as filter criteria and letting users discover places that are less crowded given their current context. Finally, we plan to evaluate our ideas in a larger study with real data in a realistic setting and also compare the solution to an app without the introduced concepts. This would also allow for differentiating the different graphs and visualizations with respect to their perceived usefulness.

References

Adomavicius, G. & Tuzhilin, A. (2015). Context-Aware Recommender Systems. In Ricci, F., Rokach, L. & Shapira, B. (Eds.): *Recommender Systems Handbook. Second edition.* Boston: Springer US, pp. 191-226.

Braunhofer, M., Elahi, M. & Ricci, F. (2014). Usability assessment of a context-aware and personality-based mobile recommender system. In Hepp, M. & Hoffner, Y. (Eds.): *E-commerce and web technologies.* Cham: Springer International Publishing, pp. 77-88.

Braunhofer, M., Lamche, B., Ricci, F. & Wörndl, W. (2015). A Context-Aware Model for Proactive Recommender Systems in the Tourism Domain. In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct.* New York: ACM, pp. 1070-1075.

Dali Betzalel, N., Shapira, B. & Rokach, L. (2015). "Please, Not Now!": A Model for Timing Recommendations. In: *Proceedings of the 9th ACM Conference on Recommender Systems.* New York: ACM, pp. 297-300.

De Pessemier, T., Dooms, S. & Martens, L. (2014). Context-Aware Recommendations Through Context and Activity Recognition in a Mobile Wnvironment. *Multimedia Tools and Applications, 72(3),* pp. 2925-2948.

Gallego, D., Woerndl, W. & Huecas, G. (2013). Evaluating the Impact of Proactivity in the User Experience of a Context-Aware Restaurant Recommender for Android Smartphones. *Journal of Systems Architecture, 59(9),* pp. 748-758.

Lathia, N. (2015). The Anatomy of Mobile Location-Based Recommender Systems. In Ricci, F., Rokach, L. & Shapira, B. (Eds.): *Recommender Systems Handbook. Second edition.* Boston: Springer US, pp. 493-510.

Melià-Seguí, J., Zhang, R., Bart, E., Price, B. & Brdiczka, O. (2012). Activity Duration Analysis for Context-Aware Services using Foursquare Check-ins. In: *Proceedings of the 2012 International Workshop on Self-Aware Internet of Things.* New York: ACM, pp. 13-18.

Ricci, F., Rokach, L. & Shapira, B. (2015). Recommender Systems: Introduction and Challenges. In Ricci, F., Rokach, L. & Shapira, B. (Eds.): *Recommender Systems Handbook. Second edition.* Boston: Springer US, pp. 1-34.

Runkler, T. A. (2012). *Data Analytics: Models and Algorithms for Intelligent Data Analysis.* Wiesbaden: Vieweg+Teubner Verlag.

Weber, H. & Novet, J. (2015). Foursquare by the Numbers: 60M Registered Users, 50M MAUs, and 75M Tips to Date. *Retrieved April 8, 2016 from http://venturebeat.com/2015/08/18/foursquare-by-the-numbers-60m-registered-users-50m-maus-and-75m-tips-to-date/.*

Woerndl, W., Huebner, J., Bader, R. & Gallego-Vico, D. (2011). A Model for Proactivity in Mobile, Context-Aware Recommender Systems. In: *Proceedings of the fifth ACM Conference on Recommender Systems.* New York: ACM, pp. 273-276.

Contact Information

Technical University of Munich (TUM), Chair of Connected Mobility (I11),
Boltzmannstr. 3, D-85748 Garching.
Email: woerndl@in.tum.de

Authors

Patrick Hiesel has recently completed his M.Sc. in Computer Science at Technical University of Munich. His research interests include recommender systems, machine learning and artificial intelligence.

Wolfgang Wörndl is a senior researcher and lecturer at Technical University of Munich. His current research interests include interactive recommender systems in mobile scenarios.

Matthias Braunhofer has recently completed his PhD in Computer Science at the Free University of Bozen - Bolzano. His research interests include context-aware recommender systems, mobile recommender systems and active learning in recommender systems.

Daniel Herzog is a research assistant and PhD student at Technical University of Munich. His research interests include mobile recommender systems, context-aware applications and human-computer interaction.