Inferring Contextual User Profiles - Improving Recommender Performance

Alan Said
TU Berlin
DAI Lab
alan.said@dai-lab.de

Ernesto W. De Luca
TU Berlin
DAI Lab
ernesto.deluca@dai-lab.de

Sahin Albayrak
TU Berlin
DAI Lab
sahin.albayrak@dai-lab.de

ABSTRACT
In this paper we present the concept of inferred contextual user profiles (CUPs) which extends the traditional user profile definition by describing the user in a given situation, or context. The approach is evaluated in the scope of movie recommendation. In our evaluation, we infer two CUPs for each user, and use only one of the profiles, instead of the full user profile for recommending movies. We evaluate the model on a data snapshot from the Moviepilot movie recommendation website, with results showing a substantial improvement in terms of precision, recall and mean average precision.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: Retrieval models; H.3.5 [**Online Information Services**]: Web-based services

General Terms
Algorithms, Design, Experimentation, Human Factors

Keywords
recommender systems, collaborative filtering, experimentation, context-awareness, user modeling, information retrieval, human factors, movie recommendation

1. INTRODUCTION
Recommender systems have become a popular component in online services to help and guide users in information retrieval oriented tasks [16]. Frequently, recommender systems infer the preferences of users based on a priori data, i.e. the already consumed data. Collaborative Filtering (CF) models are the de facto standard in when it comes to recommendation of frequently consumed items, e.g. movies, books, etc [14, 16]. CF calculates the relevance of an item for a user based on other users' rating information on items co-rated by the user and his or her peers. CF approaches are commonly categorized as either model-based or memory-based [8]. In this work we focus on the latter, which creates item prediction for a user by finding users similar to that user (in terms of co-rated items), a so-called neighborhood. The information from the neighborhood is then used to predict items not rated by the user which should be of interest. Memory-based, or neighborhood-based approaches commonly use measures such as the Pearson correlation Coefficient or cosine similarity to create the neighborhoods [14].

However, in some situations, approaches using only the historical usage information of users are not capable of identifying relevant items [2], or approaches utilizing other information can provide better recommendations. Instead, if at first identifying the situation, the *context*, a system can provide tailored recommendations for the specific context, provided information about it is available.

In order to create a context-aware recommendation model, one needs to define the concept of context. In this work we use Dey's widely-accepted definition: *"Context is any information that can be used to characterize the situation of an entity"* [11]. Here, the entity is understood as an item which can be influenced by contextual parameters that describe the state of the user and item during consumption.

Context-aware systems commonly use a predefined static set of contexts in order to generate recommendations for the specific situation, e.g. weekday, season, time of day [4, 13].

We propose an approach for automatic context-inference in the scope of movie recommendation, based on the time of a rating event and the information on whether or not the rated movie is still shown in the cinema.

Our approach to context-inference for recommendation is evaluated using a dataset from the Moviepilot[1] movie recommendation website. We present an inferred Contextual User Model (CUP), a user profile, similar to the "micro-profile" concept by Baltrunas and Amatriain [4]. Our model infers the context of where a movie was seen (at the cinema, or at home) through a combination of movie meta data, the dates of when a movie was shown in the cinema, and the creation time of the rating, i.e. the time when the movie was rated by a user. The model creates two "virtual" (context) profiles for each user (two CUPs), the *cinema* CUP and the *home* CUP.

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

[1] http://www.moviepilot.de

The biggest difference between our work and the related work described in section 2 is that we **infer Contextual User Profiles automatically** (i.e. split users into context-aware sub-profiles, as shown in Figure 1), and show that even this simple model of context-inference adds to the quality of a recommender. The process is presented in detail in Section 3.

Our experiments show that when using our context model, we can improve recommendation results significantly compared to the uncontextualized preferences of users. The full details of our evaluation and results are presented in Section 4. The paper is concluded by a summary of the contributions and a discussion about future work in Section 5.

Our main contribution is showing that a relatively simple inference model based on surrounding information can be used to boost recommendation results considerably.

2. RELATED WORK

At the moment, recommender systems tend to use very simplistic user models, adding new user preferences to the existing profiles as the users interact with more items (e.g. rate new movies, buy new books, etc.). But these approaches often ignore the *"situated action"* of the user. Situated action states that users who interact with a system in a particular context have items that are relevant within that context may find the same items irrelevant in a different context [15].

As stated by Mobasher [15], context plays an important role in psychology for human memory as well as in linguistics for disambiguation purposes. Research in intelligent information systems has also shown that incorporating context, or situational awareness, in the recommendation process increases the performance and perceived usefulness of recommender systems [4].

Adomavicius and Tuzhilin [2] divide context-aware recommender systems (CARS) into three types:

1. Contextual **Pre-Filtering**, where context directs data selection
2. Contextual **Post-Filtering**, where context is used for filtering recommendations computed by traditional approaches.
3. Contextual **Modeling**, where context is directly integrated into the model

Contextual pre-filtering can be achieved by using "micro-profiles" where a single user profile is split into several, possibly overlapping, contextual sub-profiles, each representing the user in one or several particular contexts [4]. Here, the recommendation process uses these micro-profiles, not only a single user model. The performance is shown to be better than that of traditional Collaborative Filtering methods.

Contextual post-filtering is applied within traditional approaches, while contextual modeling directly involves the model, e.g. adapting a generic tensor factorization approach. An example of this is the tensor factorization-based Collaborative Filtering method, by Karatzoglou et al. [13], which allows a flexible and generic integration of contextual information using a User-Item-Context N-dimensional tensor for modeling data, instead of the traditional User-Item matrix. In their "Multiverse Recommendation" model, every different type of context is considered as an additional dimension in the data representation, extending the user-item matrix to a tensor. The factorization of this tensor leads to a compact data model that can be used to provide context-aware recommendations.

Bogers [6], presents a movie recommendation algorithm, ContextWalk, based on taking random walks on the contextual graph. In addition to the common CF user-item relations, this algorithm allows the inclusion of different types of contextual features, such as actors, genres, directors, etc. It supports other recommendation tasks with the same random walk model without the need for alteration or retraining, e.g. recommending interesting movies or actors for a specific group of users.

Contextual user modeling, and context-awareness in general have been hot topics during recent years with numerous papers [4, 13, 17], workshops [3, 10], etc. covering the field. However, the topic is not new, and has been touched upon for the better part of the last 20 years. One of the earliest systems using the concept of location-based context, the Active Badge Location System by Want et al. [18], introduced this type of context-awareness as a means of providing services to people in an office environment. Similar systems have been subsequently put to use both in research and the industry, Bokun and Zielinski [7] for instance, created the Next Generation Active Badge System which broadcast the location of the badge wearers. Abowd et al. [1] wrote about context for mobile environments in the form of location for automated tour guides already in 1997.

3. CONTEXTUAL USER MODELING

Given an analysis of user modeling in the scope of recommender systems, in this paper, we choose to extend the term to *contextual* user modeling as our focus is on defining *context-aware user profiles* (CUPs). Each CUP is specific for the situations a user encounters.

The context profile model we describe is based on the location and time, the context (or "situated action" [15]), in which a user watches a movie. Given a set of users, movies and ratings with timestamps of when the rating event occurred, we infer the context of the rating event. We define two CUPs, *home* and *cinema* and assign each user's movie ratings to one of these as shown in Figure 1. Assignment of ratings is based on the assumption that movies rated within two months of their cinema premiere date have been seen in the cinema[2], we consequently assume movies rated at a later point in time are assumed to have been seen at home.

Having created two CUPs for every user, we can now use a collaborative filtering approach to recommend movies for

[2] the specific time a movie is shown in the cinema usually varies depending on the number of visitors, however the time between the cinema and home release of a movie usually varies between 4 weeks - 4 months [9], 2 months being typical for German cinema

each of the CUPs based on the ratings in each specific context.

	u_i	u_j	u_k	u_m	u_l
m_a	1	3		5	
m_b		4			4
m_c			5	2	
m_d	5	3		3	
m_e	3	4	1		1

(a) Uncontextualized rating matrix.

	u_i		u_j		u_k		u_m	u_l
	home	cinema	home	cinema	home	cinema	cinema	home
m_a	1			3			5	
m_b				4				4
m_c						5	2	
m_d	5		3				3	
m_e		3		4		1		1

(b) The same rating matrix, where users from (a) have been divided into CUPs.

Figure 1: Shown is an example of a user-movie matrix (a) and a user-movie-context (b) matrix. Columns with identifier $u_{i\ldots l}$ refer to users and rows with identifiers $m_{a\ldots e}$ to movies. The elements of the matrix are the ratings of users given to movies. All users might only have one CUP, as is the case with u_k.

This type of modeling is in agreement with the pre-filtered context-awareness concept discussed in Section 2. It is also related to the time-based "micro-profiles" approach presented by Baltrunas and Amatriain [4] where users are also divided into sub-profiles, however these sub-profiles are based on the time of the event only, without taking its location and item specific meta data into consideration.

The rationale for this division is the assumption that people have different rating profiles, or different tastes, based on where and when they see a movie, consequently the movies which should be recommended to users should be different depending on how the movie will be consumed.

Our model is built upon the assumption that users rate movies they have seen within a short amount of time from the time of viewing, i.e. generally not saving up ratings for, rather rating them continuously . This is supported by the general rating trend shown in Figure 2. The graph shows the average number of ratings per user from the initial month of registration for both the subset used in our experiments (introduced in Section 4.1) and the full dataset. As some users stop using the service, the number decreases over time. The high amount of ratings in the beginning indicates that users rate a "larger than normal" amount of movies just after registration, in order to create their profiles, but after one or two initial rating sessions, the average number of ratings per user per month stabilizes at between 10 and 12. There are no extreme anomalies (peaks) in the curve, would there be any, these would indicate accumulated rating sessions.

4. EXPERIMENTS AND RESULTS

We evaluated our contextual user profile model on a dataset from the German movie recommendation community Moviepilot. It should be noted that the algorithm itself is not the focus of our evaluation, rather the concept of inferred contextual user profiles.

4.1 Dataset

The Moviepilot website contains information and news about movies, actors, directors, etc., as well as the ratings of movies seen by its users. One of the services offered by Moviepilot are movie recommendations. Each user is presented with a set of movies which should be of interest. These recommendations are based on the users', and their peers', previously rated movies.

This dataset is a subset of the full, unfiltered, data that creates the basis for the Moviepilot website. The dataset was obtained directly from Moviepilot, thus eliminating any inconsistencies which might be the result of crawling a website like this. The dataset contains ratings by $10,000$ randomly selected users who have rated at least one movie. In addition to the ratings, the dataset also contains information on when movies had their cinema premieres. The total number of ratings in our subset is $1,539,393$ spread over four years. The total number of ratings in Moviepilot over the same amount of time is more than 7 million. Figure 2 shows the number of ratings per month in both datasets. The ratings are stored on a 0 to 100 scale with 0 being the lowest and 100 being the highest. The scale the users are presented with is 0.0 to 10.0.

4.2 Experimental Setup

The algorithm used to produce the recommendations is based on collaborative filtering [16]. We evaluate our results on a subset of $10,000$ randomly selected users due to the long running times of the experiments when the full dataset was used. Even for this subset, each experiment took circa 3 hours to complete on a 2.4GHz dual core PC.

For the experiments, 50 training and evaluation sets each for the original and for the contextual user profiles were created. The evaluation sets consisted of circa 5000 ratings for 500 randomly selected CUPs for the contextualized evaluation. Analysing the $10,000$ users in our dataset, we were able to identify $7,487$ cinema CUPs and $4,670$ home CUPs - meaning that not all users seem to rate movies in both contexts. For the uncontextualized case, the CUPs were merged into the original user, meaning a fewer number of columns in the input matrix (see Figure 1(a)). The merged columns have roughly twice as many ratings each though[3].

In order to avoid problems related to cold start, for both users and items, we decided that users in the evaluation sets had to have rated at least 30 movies. For each of these users, 10 movies having been rated with a value above the user's average rating were extracted into the evaluation set (i.e. the set of True Positive recommendations). The rest of the ratings were used for training. The recommendation algorithm was run one time each for the 50 pairs of original and CUP datasets. The results presented in this paper are averaged over all 50 runs.

The recommendation algorithm used in our experiments was *K-Nearest Neighbor* using the Pearson Correlation Coefficient as the neighbor similarity measure. Experiments were performed for $K = 150$. We evaluate our recommendations with the Mean Average Precision (MAP), Precision at N, and Recall at N measures. These measures where chosen since they are well-known and widely-used in the field of

[3]which should bias the results positively for the original setup as the number of true positives becomes twice as high (at most) for the merged users compared to the CUP's.

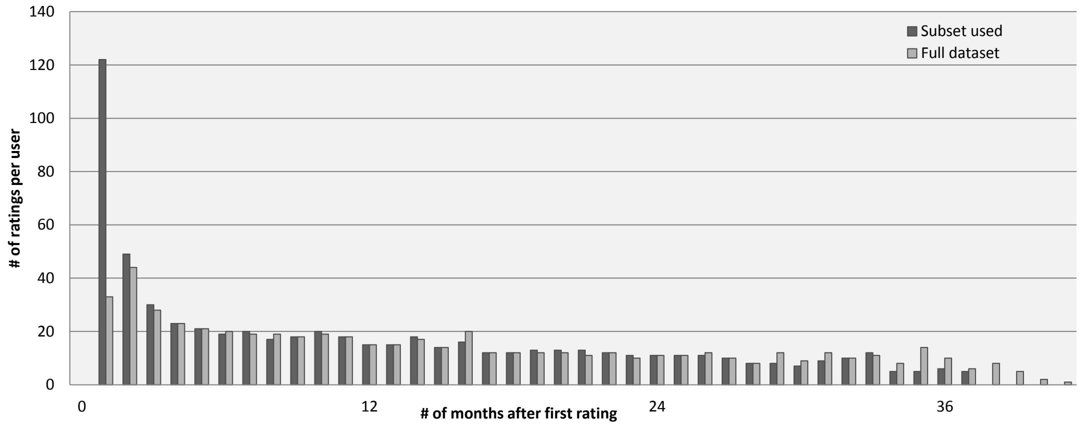

Figure 2: The sum of the total number of ratings per month per user since their first rating. The number of ratings, in both the full dataset as well as in our subset stabilizes at around 10 ratings per month per user. The high number for the first month in our dataset is explained by the users in our dataset being active users, i.e. who create a profile for the purpose of returning. The significantly lower value in the full dataset is due to users who create a profile, rate very few items and never return.

Recommender Systems and Information Retrieval, providing a statistically sound estimate of the recommendation quality [12].

4.3 Results

Figure 3 shows the precision levels obtained in our experiments. The recommendations using the contextualized user profiles outperform the original dataset by 200% when recommending one item only in terms of average precision. The approach consistently outperforms the baseline until the recommended set reaches circa 50 items. In terms of recall, shown in Figure 4, the CUP approach consistently outperforms the baseline. When looking at each CUP separately we see that the home CUP outperforms all other approaches (contextual and not contextual) by even more. The performance in terms of recall is similar, however the original users profiles never seem to be able to outperform the CUPs. When looking at MAP, shown in Table 1, the improvement is somewhat smaller, which is expected given the fact that precision is higher for the original user profiles at high N's.

The observed results confirm the assumption that the location and situation ("situated action" [15]) influences the consumer in such a way that the taste (i.e. rating value) differs from situation to situation. This confirms the notion of users having separate rating profiles depending on the combination of where, how and when the movie is seen. More importantly, the performance of a recommender system can be improved considerably if this information is used.

5. CONCLUSION

In this paper we presented a method for automatic contextualization of rating events in a movie recommendation scenario, in order to create contextual user profiles, CUPs. By using the date of the rating, and the information on how new a movie was at the time of rating, we were able to infer the venue (at home, or at the cinema) in which a movie was seen.

We evaluated the inferred contextual user profiles and were able to considerably improve recommendation results in terms of precision, recall and mean average precision. Results indicate that automatic contextualization of user profiles into CUPs affects the quality of recommendations positively. We showed that, in a movie recommendation scenario, the venue and time of a consumption as well as the "freshness" of the item is reflected in the rating behavior of users and that this information can be used for recommendation purposes.

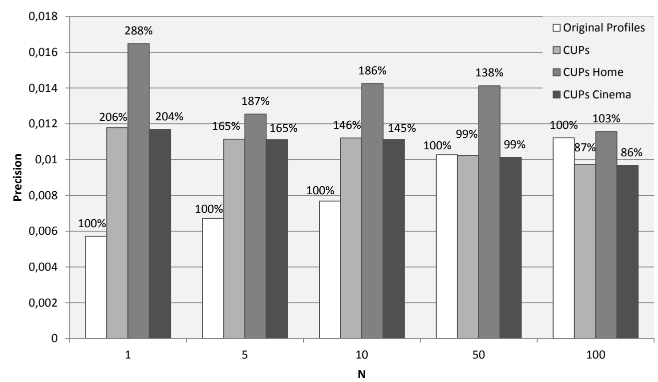

Figure 3: Precision@N with N={1, 5, 10, 50, 100} for the original user profiles, the average value for both home and cinema CUPs and for each of the two inferred CUPs.

The situation in which users consume a particular product, has an effect on their taste or rating behavior. However, the context covered in this work needs to be extended and further researched to gain more insight into the way contextualized user profiles should be inferred, managed and used. For instance, the profiles explored in this work are mutually exclusive, which, in the presented recommendation scenario, seems plausible, as the location of an event can only be singular. If the context profile would be extended to include factors such as company, mood or ambiance of the venue,

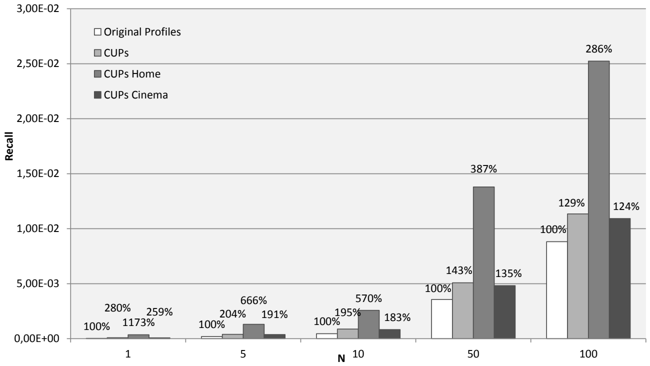

Figure 4: Recall@N with N={1, 5, 10, 50, 100} for the original user profiles, the average value for both home and cinema CUPs and for each of the two inferred CUPs.

Recommender	**MAP**	% improvement
Original users	$5.26E-3$	0%
Contextual user profiles	$6.05E-3$	15%
Home Context	$7.97E-3$	51%
Cinema Context	$6.00E-3$	14%

Table 1: The Mean Average Precision values and the relative improvements for our CUPs model and the original user profiles.

the assumption on mutual exclusiveness of the contexts may need to be relaxed.

Our current work includes the in-depth analysis of data in order to be able to accurately identify other contexts, infer them from implicit relations and subsequently use them for recommendation purposes.

In conclusion, it appears that even trivial context inference models can be used to considerably improve recommender systems quality, without adding much complexity to the recommendation algorithms themselves.

In this paper we have covered the topic of inferred Contextual User Profiles (CUPs), and showed that, even with rather simple inference models, there is much to gain in terms of recommendation quality. The contexts covered in this work have been one related to watching movies in the comfort of one's home, and one where the watching takes place at a cinema. Both contexts improve recommendation quality considerably.

6. ACKNOWLEDGMENTS

The authors would like to express their gratitude to the Moviepilot team who contributed to this work with dataset, relevant insights and support.

The work in this paper was conducted in the scope of the KMulE project which was sponsored by the German Federal Ministry of Economics and Technology (BMWi).

7. REFERENCES

[1] G. D. Abowd, C. G. Atkeson, J. Hong, S. Long, R. Kooper, and M. Pinkerton, 'Cyberguide: a mobile context-aware tour guide', *Wirel. Netw.*, **3**, (10/1997).

[2] G. Adomavicius and A. Tuzhilin, *Context-Aware Recommender Systems*, 217–257, Springer, 2011.

[3] G. Adomavicius, A. Tuzhilin, S. Berkovsky, E. W. De Luca, and A. Said, 'Context-awareness in recommender systems: research workshop and movie recommendation challenge', in *RecSys 2010*. ACM.

[4] L. Baltrunas and X. Amatriain, 'Towards Time-Dependant recommendation based on implicit feedback', in *CARS 2009*, (2009).

[5] S. Berkovsky, A. Said, and E. W. De Luca, eds. *CAMRa '10*. ACM, 2010.

[6] T. Bogers, 'Movie recommendation using random walks over the contextual graph', in *CARS 2010*.

[7] I. Bokun and K. Zielinski, 'Active badges–the next generation', *Linux J.*, **10/1998**, (1998).

[8] J S Breese, D Heckerman, and C Kadie, *Empirical analysis of predictive algorithms for collaborative filtering*, volume 461, 43â52, San Francisco, CA, 1998.

[9] Ben Child. Closing the window on the multiplex | ben child | guardian.co.uk. `http://www.guardian.co.uk/film/filmblog/2010/may/28/cinema-window-dvd-release-multiplexes` (retrieved 07/2011), May 2010.

[10] E. W. De Luca, A. Said, M. Böhmer, and F. Michahelles, 'Workshop on context-awareness in retrieval and recommendation', in *IUI*. ACM, (2011).

[11] A. K. Dey, 'Understanding and using context', *Personal Ubiquitous Comput.*, **5**, (01/2001).

[12] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, 'Evaluating collaborative filtering recommender systems', *ACM Trans. Inf. Syst.*, **22**, (01/2004).

[13] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver, 'Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering', in *RecSys 2010*. ACM, (2010).

[14] G. Linden, B. Smith, and J. York, 'Amazon.com recommendations: item-to-item collaborative filtering', *Internet Computing, IEEE*, **7**(1), (jan/feb 2003).

[15] B. Mobasher, 'Contextual user modeling for recommendation', in *Keynote at the 2nd Workshop on Context-Aware Recommender Systems*, (2010).

[16] Moviepilot. Wie funktioniert moviepilot? `http://www.moviepilot.de/pages/faq#wie_funktioniert_moviepilot` (retrieved 03/2011).

[17] A. Said, 'Identifying and utilizing contextual data in hybrid recommender systems', in *RecSys*. ACM, (2010).

[18] R. Want, A Hopper, V. Falcão, and J. Gibbons, 'The active badge location system', *ACM Trans. Inf. Syst.*, **10**, (01/1992).