Identifying and Utilizing Contextual Data in Hybrid Recommender Systems

Alan Said
DAI Lab
Technische Universität Berlin
TEL 14, Ernst-Reuter-Platz 7
D-10587 Berlin, Germany
alan.said@dai-lab.de

ABSTRACT

Context-aware recommender systems are becoming a popular topic, still, there are many untouched aspects. In this paper, research involving context identification and the concepts related to hybrid and context-aware systems is presented. Furthermore, a conceptual architecture for a context-aware recommender system for movies and TV shows is introduced. The system consists of a number of processes for context identification and recommendation. Key contextual features are identified and used for the creation of several sets of recommendations, based on the predicted context. The main focus of the research presented here is the identification of context, which in turn is used for recommendation. The approach will be incorporated and evaluated in the recommendation engine of movie recommendation website Moviepilot.

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

Keywords

Recommender Systems, context-awareness, personalization

1. INTRODUCTION

With the increasing quantity of data available online, recommender systems have become an everyday tool for Internet users, providing them with much needed help in discovering information. They have become ubiquitous in fields such as shopping (Amazon), movies (Netflix) and music (Last.fm), simplifying information management and personalization.

Currently, widely used systems take into account either historical *usage data* (collaborative filtering) or *item content*. Systems based on usage data, find items that should be relevant for a certain user based on what other similar users have seen, bought or interacted with previously, etc. Content-based systems on the other hand, compare items in terms of content similarity, and recommend similar items. These recommender systems have been among the de facto standards for nearly twenty years [13].

The integration of additional features, for so-called *hybrid recommenders*, e.g. usage data and content combined, has been shown to improve the quality of recommendations [8, 17]. Closely related to hybrid approaches, *context-aware* ones, utilize not only additional meta data, but are also aware of the user's context, e.g. situation such as time of day, company, mood, etc. From a contextual perspective, hybrid recommenders can be likened to recommenders with statically defined context, or a static feature set. Whereas current context-aware recommenders have one (or several) statically defined contexts, through contextual features or feature dependencies. However, due to the static nature of current context-aware recommenders, they do not have the ability to determine the user's current context. Instead they use one or a set of predetermined contexts.

In situations where each item or piece of information has a cost (i.e. price, time it takes to read or view the item, etc), the relevance of the recommended items is especially important, otherwise the system faces a risk of users losing trust in it or completely stop using it. In this work, the aim is to automatically conjecture the context of the users in order to be able to provide higher quality recommendations. Intuition and experience tell us that currently favored items need not be the same as the favored items of yesterday or tomorrow. Static context-awareness could be applied to each specific case, dynamic context-awareness would allow for accurate recommendation results through an automated context identification process, no matter the time, place or any other circumstance.

1.1 Why Context?

Most of todays recommender systems produce static recommendations, not taking into account parameters such as time of day or year, whether or not the user is in the company of others, etc, instead they are based on item and user similarities. By analyzing real world datasets from a contextual perspective, the aim is to be able to automate the process of context identification in order to incorporate it

RecSys2010, September 26–30, 2010, Barcelona, Spain.

into the process of recommendation, thus improving recommendation results. Automated context identification is, to our knowledge, an untouched problem.
In order to understand the complexity of context, a significant portion of this research lies on the analysis of features and dependencies from a contextual point of view. This provides a basis for continued research on identification and modeling of context, and its utilization in recommender systems. The focus of the research presented in this paper is context identification and its utilization. These two issues have been discussed in the research community during the last couple of years, as the community has grown aware of the benefits in context-awareness [2].

1.2 Related Work

Online movie recommendation is actually older than the Web itself [7,12]. Regrettably, recommendation engines have not evolved as much as the Web has. Many of today's state of the art systems incorporate other aspects of data in their algorithms (annotations, genres), resulting in so-called, hybrid systems [9]. None to very few of them, however, make use of this hybrid data in order to identify context or generate context-aware recommendations.
Twenty years ago, a website would not be dynamic, nor would it have the same quantity of information about its visitors and their interaction with it. The purpose was to present data. Although the purpose has not changed, the quantity of data available has. The goal of this research is to find better techniques to utilize this data to create *context models* and recommender systems that use these instead of the static standards employed today.

1.2.1 Hybrid models

Hybrid recommender systems usually mix content-based recommendation with its collaborative filtering-based counterpart. Gunawardana and Meek [9] create a hybrid algorithm for recommending movies based on unified Boltzmann machines using movie meta data for hybridization which outperforms other non-hybrid approaches. Another hybrid approach is the one of Wetzker et al. [18] where the authors create an algorithm for tag recommendation based on personomy (the individual tag vocabulary of each user) translation, and outperform other state of the art approaches for tag recommendation thanks to the utilization of content in the form of tags.

1.2.2 Context-aware models

Context-awareness in recommender systems has, in one way or another, been a part of the recommender systems community for a number of years. The Workshop on Context-aware Recommender Systems (CARS) [1], which followed a tutorial on the same topic [3] one year prior, created a forum where current state of the art research could be shared with other members of the community. In the workshop, Baltrunas and Amatriain [4] presented a time-aware approach for music recommendation, creating time-based user models which were then used to recommend music artists. Lombardi et al. [11] created contextual models and aggregated user profiles based on subsets of historical events, thus being able to recommend items for groups of users for certain contexts. Yet another context-aware approach was presented by Baltrunas and Ricci [5] where items were split into virtual items based on a contextual feature derived from item ratings. Their evaluation shows that even quite simple contextual item splitting improves results.
The concept of dynamically identifying context is still untouched in the topic of recommender systems. Dynamic feature selection however has been discussed by others, for instance Yang et al. [19] for facial emotion recognition, and Islam et al. [10] for spam filtering.

1.2.3 Unified models

As mentioned in Section 1, hybrid approaches are relevant to the context-aware recommender field, for instance, as in the case described in Section 1.2.1 where language could be seen as a static contextual feature [18]. Furthermore, the hybrid model described by Wetzker et al. [16] could quite easily be turned into a context-aware one if the feature selection was made from a contextual perspective.
Adomavicius et al. [2] combined the concepts of contextual and hybrid recommender systems and created models for dimensionality reduction of both multidimensional content data, as well as its contextual counterparts, and then combined these into a *unified recommendation model*. Their approach was shown to perform better than simpler non unified approaches.

2. CONTEXT-AWARE RECOMMENDER

The aim of this work is to create a model for dynamic and context-aware recommender systems. For the analysis and evaluation, datasets with hybrid and contextual information from the movie recommendation websites Moviepilot[1] and Filmtipset[2] are used. A simplified entity relation diagram of one of the datasets is presented in Figure 1. By analyzing the datasets, valid insight into which features are important for context identification is gained. Moviepilot will furthermore provide the possibility to perform online evaluation on their website, thus creating a unique research environment where evaluation will be based on explicit user feedback, instead of theoretical or hypothetical models.
This dissertation is an integral part of a project intended to develop novel architectures for dynamic hybrid and context-aware recommenders, given the predicted user context and live implicit feedback from users of the website.

2.1 Context Identification

In order to identify context, one needs to understand the concept of context itself. In this work we acknowledge three types of context, *user context*, *item context* and *system context*. User context is the situation of the user when she is given recommendations, item context is similar to user context, with the obvious exception that is focuses on the item instead. System context is the situation of the system itself. Thus when identifying context, we conjecture the situation based on any available information about user, item and system. The datasets mentioned above contain a collection of features generally not present in publicly available datasets. Some of the more interesting features are different types of hierarchical keywords given to movies. The keywords range from *movie emotion* and *movie mood* to plot-related keywords, e.g. "weird", "witty", "serious", etc. In addition to this, the datasets also contain common fea-

[1] `http://www.moviepilot.com`
[2] `http://www.filmtipset.se`

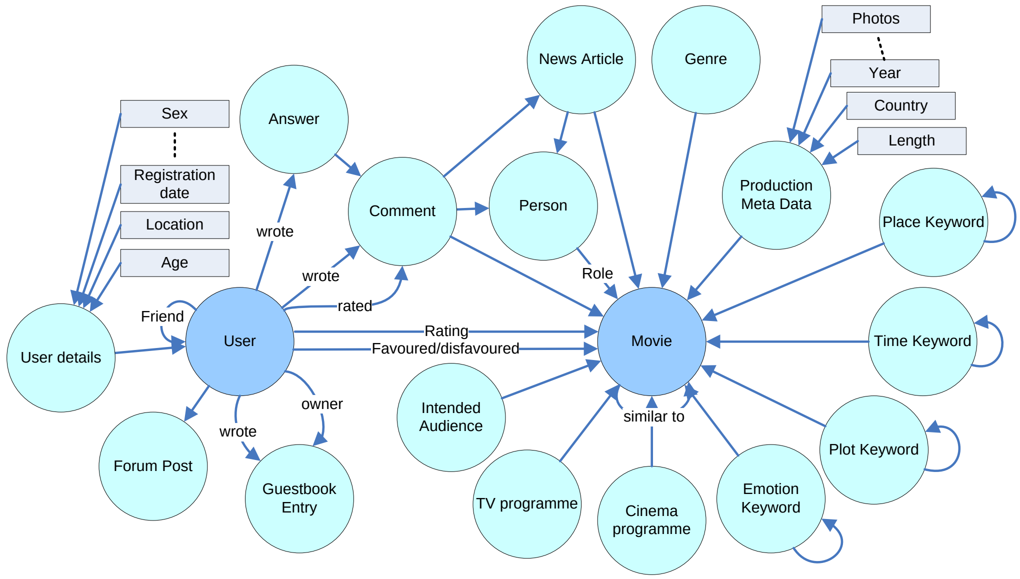

Figure 1: An abstract entity-relationship diagram of one of the datasets used. Every user generated relation, e.g. rating, comment, etc. holds the timestamp of when it was generated or altered.

tures such as genre, production details, actors, directors, etc. Furthermore, all user-created relations, such as ratings, comments and friendships are labeled with timestamps which state the time of creation or last alteration. Given the heterogeneity and feature richness of the datasets, finding context-dependent features and dependencies is very plausible. One such possible dependency might for instance be a mood-time-company dependency per user. Intuitively, the mood of a user on a Sunday afternoon is different from the same user's mood during working hours mid-week, likewise the context is affected by the user's current company. Knowing these dependencies, whether they are daily, weekly or seasonal brings the possibility to create personal contextual models. Further combining these into hybrid contexts, e.g. a certain weekday during summer together with family, and additionally including a feedback loop from the website (link clicks, backwards, etc) will provide a means of deciding whether or not the predicted context is the user's actual one.

2.2 Previous, Current and Future Work

For the purpose of identifying context and evaluating features, contextual as well as hybrid, we have worked on different kinds of recommendations datasets. For hybrid approaches the focus was on folksonomies [17]. Folksonomies, consisting of triples of *user, item, tag* are well suited for hybrid algorithm development as the tri-partite data contains all information necessary. Among other things, we explored how adding an additional feature space to the usual collaborative filtering concept affects latent topic modeling for the purpose of item recommendation [15]. The addition of tags was shown to improve recommendation results. The algorithmic solution can easily be extended to include additional features, thus also suitable for multi-dimensional and context-aware topic modeling. In previous context related work, we used datasets with features similar to the one describe in Figure 1. One of these was introduced and evaluated in [14] where different kinds of social relations (social context) were analyzed and results presented on how relationships could be used to generate higher recommendation accuracy. The focus of this work this far has been related to evaluating how different user-related features affect outcomes of recommendation [14]. Currently, we are exploring contextual dependencies between different keywords and user types. Other areas investigated are social contexts, evaluating how friendships affect users preferences. Future work includes further analysis of contextual features and exploring dependencies between combinations of features and contexts.

2.3 Architecture

The proposed recommender system architecture is divided into two main parts, a *contexter* and a *set of recommenders*. The contexter attempts to identify the current user context given historical data and feedback from the website, whereas the set of recommenders create the ensemble used for item recommendation. An abstract architectural overview is shown in Figure 2. Conceptually, *Usage data* in Figure 2 corresponds to all non explicitly contextual relations in Figure 1, i.e. user-item, user-comment, movie-actor, etc., whereas *Meta data* corresponds to all non trivial relations, such as user-mood-time, user's preferences - cinema/TV listings - company, etc.

2.3.1 Contexter

The contexter is a module where context is identified based on available data and live feedback from the website. The identification of context is based on users and items histories combined with time and any features that may be of value. Extracting the contexter from the recommender itself will reduce the complexity of it due to the dynamically reduced feature set delivered from the contexter, this corresponds to so-called *pre-filtering*. This does, however, not imply that only one recommendation algorithm will be

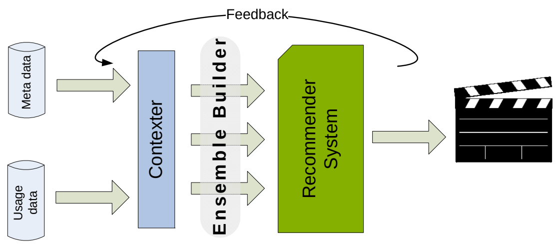

Figure 2: The proposed architecture of a dynamic context-aware recommender system.

employed, neither does this imply that the same reduced feature set will be forwarded to all employed algorithms.
The contexter will, essentially, work as a dynamic filter for what should (or should not) be used by the recommender system.

2.3.2 *Recommender*

The recommender, or set of recommenders, is several recommendation algorithms fed with data features selected by the contexter. The outputs of these algorithms are combined to create the final set of recommendations. While some of these algorithms have already been developed and evaluated, additional ones still remain as a part of the future work of my dissertation.

3. OPEN CHALLENGES

Context-awareness is not only applicable to movie recommendation, as shown in the related work in Section 1.2. However, there is relatively little research being conducted in this field. For several years, the ACM Recommender Systems conference has highlighted the topic in tutorials and workshops. One of the challenges in context-awareness is the lack of contextual data for researchers. In order to stimulate more research in the field, we provide the community unique datasets with contextual features [6], one of them being an anonymized subset of the dataset described here. The expectation is that the topic will be more accessible to a larger public through these datasets.

4. CONCLUSIONS AND PLANNED WORK

Studies have shown that context is of great importance when it comes to recommendations. Recommending irrelevant items comes with the risk of losing users trust for the system. Additionally, recommending items that come with a cost, whether it is money (shopping) or time (movies, books) creates a need for truly accurate recommender systems. In today's overwhelming quantity of available information, this becomes a task that needs more effort than the simple standards employed today.
In ongoing work we intend to explore different fields of context-awareness, among other the effect of the cost of items on context, implicit context identification, and context-aware evaluation.

5. REFERENCES

[1] G. Adomavicius and F. Ricci, 'Recsys'09 workshop 3: Workshop on context-aware recommender systems', in *Proc. of RecSys'09*, (2009).

[2] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin, 'Incorporating contextual information in recommender systems using a multidimensional approach', *ACM Trans. Inf. Syst.*, **23**(1), (2005).

[3] G. Adomavicius and A. Tuzhilin, 'Context-aware recommender systems', in *Proc. of RecSys'08*, (2008).

[4] L. Baltrunas and X. Amatriain, 'Towards time-dependant recommendation based on implicit feedback', in *RecSys'09 workshop 3: CARS*, (2009).

[5] L. Baltrunas and F. Ricci, 'Context-dependent items generation in collaborative filtering', in *RecSys'09 workshop 3: CARS*, (2009).

[6] S. Berkovsky, E. W. De Luca, and A. Said. RecSys'10, Challenge on Context-aware Movie Recommendation. `http://www.dai-labor.de/camra2010` - May, 2010, 2010.

[7] T. Berners-Lee. The WorldWideWeb browser. `http://www.w3.org/People/Berners-Lee/WorldWideWeb` - March, 2010.

[8] T. De Pessemier, T. Deryckere, and L. Martens, 'Context aware recommendations for user-generated content on a social network site', in *Proc. of EuroITV '09*, (2009).

[9] A. Gunawardana and C. Meek, 'A unified approach to building hybrid recommender systems', in *Proc. of RecSys'09*, (2009).

[10] M.R. Islam, W. Zhou, and M.U. Choudhury, 'Dynamic feature selection for spam filtering using support vector machine', in *Proc. of ICIS 2007*, (2007).

[11] S. Lombardi, S.S. Anand, and M. Gorgoglione, 'Context and customer behavior in recommendation', in *RecSys'09 workshop 3: CARS*, (2009).

[12] C. Needham. IMDb history. `http://www.imdb.com/help/show_leaf?history` - December, 2009.

[13] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, 'Grouplens: an open architecture for collaborative filtering of netnews', in *Proc. of CSCW '94*, (1994).

[14] A. Said, E. W. De Luca, and S. Albayrak, 'How social relationships affect user similarities', in *IUI'10 Workshop on Social Recommender Systems*, (2010).

[15] A. Said, R. Wetzker, W. Umbrath, and L. Hennig, 'A hybrid PLSA approach for warmer cold start in folksonomy recommendation', in *RecSys'09 Workshop on Recommender Systems & The Social Web*, (2009).

[16] R. Wetzker, A. Said, and C. Zimmermann, 'Understanding the user: Personomy translation for tag-recommendation', in *ECML PKDD Discovery Challenge 2009*, (2009).

[17] R. Wetzker, W. Umbrath, and A. Said, 'A hybrid approach to item recommendation in folksonomies', in *ESAIR '09: Proc. of the WSDM '09 Workshop*, (2009).

[18] R. Wetzker, C. Zimmermann, C. Bauckhage, and S. Albayrak, 'I tag, you tag: translating tags for advanced user models', in *Proc. of WSDM'10*, (2010).

[19] Y. Yang, G. Wang, and H. Kong, 'Emotion recognition based on dynamic ensemble feature selection', in *Man-Machine Interactions*, (2009).