

Personalized and Context-Aware TV Program Recommendations Based on Implicit Feedback

Paolo Cremonesi, Primo Modica, Roberto Pagano, Emanuele Rabosio^(✉),
and Letizia Tanca

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano,
Piazza Leonardo Da Vinci 32, 20133 Milano, Italy
{paolo.cremonesi, roberto.pagano, emanuele.rabosio,
letizia.tanca}@polimi.it, modica1@hotmail.it

Abstract. The current explosion of the number of available channels is making the choice of the program to watch an experience more and more difficult for TV viewers. Such a huge amount obliges the users to spend a lot of time in consulting TV guides and reading synopsis, with a heavy risk of even missing what really would have interested them. In this paper we confront this problem by developing a recommender system for TV programs. Recommender systems have been widely studied in the video-on-demand field, but the TV domain poses its own challenges which make the traditional video-on-demand techniques not suitable. In more detail, we propose recommendation algorithms relying exclusively on implicit feedback and leveraging context information. An extensive evaluation on a real TV dataset proves the effectiveness of our approach, and in particular the importance of the context in providing TV program recommendations.

1 Introduction

Television is one of the most popular media in our era, and with the advent of digital TV and the growing offer of satellite services there are at any time of the day hundreds of available TV programs to be watched by the users on hundreds of different channels. On the one hand the user is satisfied by this abundance since the vast choice of programs supports his/her tastes, but on the other hand he/she suffers an information overload problem. This information overload makes the user prone to a tedious channel surfing in order to find what he/she really likes, inevitably leading to annoyance.

In the past the solution was represented by paper channel guides that used to be consulted on a daily basis. Nowadays, these paper supports are fallen into disuse due to the proliferation of channels and shows and the advent of smart

This research is partially supported by the IT2Rail project funded by European Union's Horizon 2020 research and innovation program under grant agreement No: 636078, and by the Italian project SHELL CTN01_00128_111357 of the program "Cluster Tecnologici Nazionali".

TVs and smart devices, and the show schedule information has been embedded into the television software itself through the so-called *electronic program guide (EPG)*. However, the low quality of the EPG in terms of content and its often crude user interface brings to a poor user experience and, as a natural consequence, to ineffectiveness. The answer to this problem consists in providing the user with a short list of recommended programs, representing the subset of the on-air ones that most correspond to his/her preferences.

Recommendation of TV programs is rather a special instance of recommendation for three reasons:

- *Available items change over time*: many TV programs, e.g. the movies, are often broadcast once and then not anymore for a long time. The system must be able to provide recommendations also for items of this kind, if they meet the users' interests.
- *Time-constrained catalog of items*: differently from the more usual video-on-demand setting, programs are transmitted in a predefined schedule. Therefore, the recommendations must consider only the items on air at the moment in which they are requested.
- *The user feedback is usually implicit*, provided in the form of watched/not watched shows.

Note that the first issue makes it impossible to adopt traditional collaborative filtering (CF) recommendation techniques. Indeed, they are not able to recommend new items since such items cannot be compared with the other ones in terms of the feedback provided by the users in the past [1].

Moreover, a fundamental aspect to be considered in TV program recommendation is the *context* [2], i.e. the situation that the user is experiencing when watching television. The context may be characterized by a number of dimensions, the most common being the time. Other contextual information often available is represented by the social setting in which the user is accessing the content, and his/her current interest topic. So, for instance, when alone during daytime the user might prefer different shows with respect to those liked when with friends in the evening.

In this paper we propose a context-aware TV recommender system relying exclusively on implicit feedback. To the best of our knowledge, this is the first attempt to tackle both context-awareness and implicit feedback in the TV domain. The proposed techniques have been extensively evaluated on a real dataset related to Italian television.

The paper is structured as follows. Section 2 surveys the existing literature, while Sect. 3 introduces the framework in which our algorithms are supposed to operate. Section 4 describes the proposed recommendation techniques, and Sect. 5 their experimental evaluation. Finally, Sect. 6 concludes the paper.

2 Related Work

Recommendation of TV programs has raised some interest in the recent literature. The existing proposals can be divided on the basis of their aim:

recommending to build a personalized video recorder (PVR), or recommending to build a personalized EPG in linear television.

A personalized video recorder is a system generating recommendations about TV content that will be stored into an internal hard disk, for a possible future viewing by the user. The work of Engelbert et al. [3] characterizes TV programs with attributes extracted from an EPG, containing information about channel, title, subtitle, genre, actors, year and description. Recommendations of programs to be recorded are generated on the basis of an initial user profile and an adaptive user profile, both sets of TV programs classified as liked or disliked. The initial user profile is manually filled by the users, while the adaptive one is built using implicit and explicit feedback collected after the user has watched the programs. Once defined the users' profiles, the attributes of new programs (taken from the EPG) are compared against those of the programs in the user's profile with the help of a bayesian classifier. Another personalized video recorder is defined by Kurapati et al. [4]; they too propose algorithms for PVRs coupling explicit and implicit feedback, in this case relying on neural networks to combine them. The problem analyzed in these works is related to ours but is not the same, because in PVRs the recommendations do not have to be provided at specific time instants.

In the scope of linear TV, our scenario of interest, Chang et al. [5] provide guidelines to create a TV program recommender, identifying the main needed modules and performance requirements. However, the proposed framework is interesting, but just sketched; among the full-fledged proposals, just a few rely on contextual information.

Some non-contextual linear TV recommenders have appeared in the literature, and many of them rely on hybrid (collaborative and content-based) systems. Barragans-Martinez et al. [6] exploit a hybrid approach to solve new-item, cold-start, sparsity and overspecialization problems; their method uses both implicit and explicit feedback, and mixes together the outcome of content-based filtering, computed using the cosine similarity between item feature vectors, and collaborative filtering, exploiting singular value decomposition. Ali et al. [7] develop TiVo, a television-viewing service for the US market incorporating a recommender system which exploits an item-item form of collaborative filtering mixed with bayesian content-based filtering; the system envisages client and server components, and relies on both implicit feedback and explicit ratings. Another hybrid approach is that of Cotter et al. [8], who present a personalized EPG; users manually input their preferences about channels and genres, and this information is combined with the user's viewing activity by means of case-based reasoning and collaborative filtering techniques. Uberall et al. [9], on the contrary, propose a fully content-based technique, exploiting both the viewing behavior and explicit user preferences on preferred genres, subgenres and TV programs.

Context is taken into account by Ardissono et al. [10], who develop a content-based system able to generate a personalized program guide. In order to model the user, the system employs several information sources: users' explicit preferences, estimates on viewing preferences using program categories and channels, viewing preferences of stereotypical viewers classes, socio-demographic information, and users' viewing behavior. Different modules of the system manage the

different kinds of information, and the results are then combined; the context is considered by the module that estimates user preferences on the basis of the user viewing behavior, since those preferences depend on day and time. Another contextual system is that of Hsu et al. [11]. They propose a hybrid system that combines the collaborative and content-based components by means of a neural network; the contextual information employed by the system is the user’s mood, considered as a strong influencing factor in program selection.

All the described approaches to linear TV recommendations, both the contextual and the non-contextual ones, exploit some form of explicit feedback which must be provided by the users, like, for instance, user ratings. On the contrary, the system we propose relies only on the availability of implicit feedback in terms of history of the past program views, which is the most realistic situation. Moreover, our algorithms exploit context information in a different and more general way with respect to what is done in [10] and [11]. In fact, [10] and [11] deal only with specific kinds of context information, while we devise a framework that can accommodate every type of context dimensions, like the kind of people present during the program view or the fact that it is a weekday or the weekend.

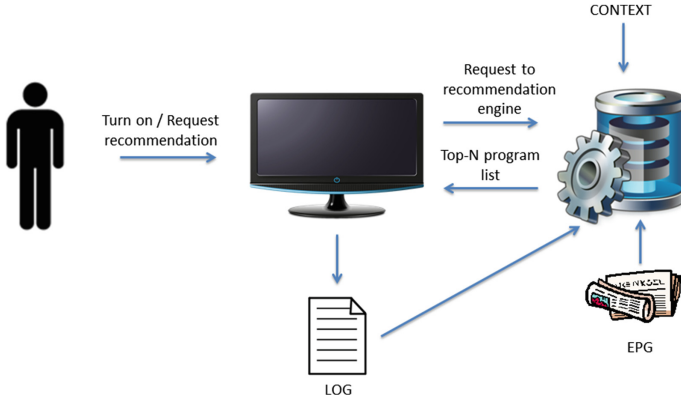
3 System Architecture

The architecture we propose for our recommender system is shown in Fig. 1. The user interacts with a smart TV, and is allowed to request the generation of recommendations; recommendations can be generated also when the TV is turned on. The request is forwarded to the recommendation engine, that exploits the log of the past user’s syntonizations, along with the context of the user and the EPG, in order to determine the list of the top-N programs to be recommended among those currently on air. Note that some kinds of contextual information may be automatically determined by the system, like the time, but others might need to be manually declared by the user, like the people with whom he/she is watching TV or his/her current mood.

4 Recommendation Methodology

Let us consider a set \mathcal{U} of users and a set \mathcal{I} of items, i.e. TV programs. Each item is described by n attribute dimensions $\mathcal{A}_1, \dots, \mathcal{A}_n$, like its genre or the channel on which it is broadcast; we denote by $\mathcal{A}_j(i)$ the value of the attribute \mathcal{A}_j for the item i . The context is described by m context dimensions $\mathcal{C}_1, \dots, \mathcal{C}_m$, like time or mood.

Given the log of the past program views, we build offline, as a model to generate the recommendations, an $(n + m + 1)$ -dimensional tensor T , storing the number of seconds spent by the users watching the TV programs with all the possible attribute values in all the possible contexts. In more detail, consider a user u , attribute values $\mathcal{A}_1 = a_1, \dots, \mathcal{A}_n = a_n$, and a context represented by the dimension values $\mathcal{C}_1 = c_1, \dots, \mathcal{C}_m = c_m$. The value $t_{ua_1 \dots a_n c_1 \dots c_m}$ stored in the tensor represents the number of seconds the user u has spent watching programs described by attribute values $\mathcal{A}_1 = a_1, \dots, \mathcal{A}_n = a_n$ when in context $\mathcal{C}_1 = c_1, \dots, \mathcal{C}_m = c_m$.


Fig. 1. System architecture

Example 1. Consider a set of context dimensions including only the time, and a set of TV program attributes constituted only by the channel. The possible values for the time context dimension are daytime and night, while those for the channel are Ch-1 and Ch-2. Figure 2 shows a possible log of syntonizations for user u .

program_id	time	channel	seconds
p1	daytime	Ch-1	2000
p2	night	Ch-2	1000
p3	daytime	Ch-1	3000
p4	night	Ch-1	2000

Fig. 2. Log of example 1

$$\begin{matrix}
 & \begin{matrix} \text{daytime} & \text{night} \end{matrix} \\
 \begin{matrix} \text{Ch-1} \\ \text{Ch-2} \end{matrix} & \begin{pmatrix} 5000 & 2000 \\ 0 & 1000 \end{pmatrix}
 \end{matrix}$$

Fig. 3. Projection of the tensor of Example 1 for user u

In this example the tensor T has three dimensions: user, time and channel. Figure 3 shows the projection of the tensor on time and channel for user u .

Once the tensor model above has been built, it can be used at runtime to generate the recommendations. The user u requests recommendations in a given time instant t when in context $\mathcal{C}_1 = c_1, \dots, \mathcal{C}_m = c_m$. Let \mathcal{I}_t be the set of programs on air at time instant t . The system extracts from the tensor the appropriate score $r_{uic_1 \dots c_m}$ for each item $i \in \mathcal{I}_t$, as follows:

$$r_{uic_1 \dots c_m} = t_{uA_1(i) \dots A_n(i)c_1 \dots c_m} \quad (1)$$

If N recommendations are required, the system retrieves the N programs with the highest values of $r_{uic_1 \dots c_m}$.

Example 2. Consider the situation described in Example 1, suppose that the system generates recommendation lists of length 1 and the user u has requested recommendations at instant t in the context $time=night$. Suppose that at instant t Ch-1 is showing program $p5$ while Ch-2 is showing program $p6$, therefore $\mathcal{I}_t = \{p5, p6\}$. According to the tensor in Fig. 1, the score for $p5$ computed using Eq. (1) is 2000 while the score for $p6$ is 1000. Therefore, the system recommends program $p5$ to u .

5 Evaluation

We start the description of the evaluation by introducing the dataset we employ (Sect. 5.1), then, the evaluation metrics (Sect. 5.2) and the compared methods (Sect. 5.3). Finally, we provide the results in a tabular form (Sect. 5.4), along with a detailed analysis (Sects. 5.5 and 5.6).

5.1 Dataset

We employed a dataset containing TV viewing information related to 7921 users and 119 channels, broadcast both over the air and by satellite. The dataset is composed by an EPG containing the description of 21194 distinct programs, and a log of the program views performed by the users. The attributes available for each program in the EPG are its genre and the channel on which it is transmitted.

The log of program views spans from December 3rd, 2013 to March 1st, 2014, and contains 10313499 entries. We deemed the syntonizations shorter than three minutes as not relevant, retaining 6525541 log entries. Each log row specifies the identifier of the user and that of the program he/she watched, along with the start time, the end time and the people with whom the user watched the program. The latter three pieces of information were used to determine the values of the three context dimensions that we chose: day of the week, time slot and *familiar context*, where with familiar context we mean just the people with whom the user was watching TV. More precisely, start and end time were employed to derive the day of the week and the time slot, where the available values for the time slot are shown in Table 1. We identified five possible relevant values for the familiar context, summarized in Table 2, depending on the age of the people; persons older than 15 years were considered adults.

The log was split in a training set, including the syntonizations between March 3rd, 2013 and February 15th, 2014 (5438977 entries), and a test set, containing the remaining ones (1086564 entries). The former was used to build the model, while the latter to assess the quality of the recommendations.

5.2 Evaluation Metrics

The performance of our recommendation algorithm was evaluated using $Recall@N$, describing the number of test items which have been included in a

Table 1. Time slots

Start time	End time	Description
02:00:00	07:00:00	Graveyard slot
07:00:00	09:00:00	Early morning
09:00:00	12:00:00	Morning
12:00:00	15:00:00	Daytime
15:00:00	18:00:00	Early fringe
18:00:00	20:30:00	Prime access
20:30:00	22:30:00	Prime time
22:30:00	02:00:00	Late fringe

Table 2. Familiar contexts

Familiar context
Mixed: adults + children
Group of adults
Group of children
Adult alone
Child alone

recommendation list of length N computed in the instant in which the viewing of the test items started and in the context in which they have been watched.

More formally, let v be a program view in the test set, v_t the start time of the view, v_u the user that watched the program, v_i the program watched and v_c the context in which the view took place. $TopN(u, c, t)$ is the set of top- N items for the user u in context c among those on air at time instant t , determined with the recommendation methodology to be evaluated. Recall@ N is computed as follows:

$$Recall@N = \frac{|v \in \text{Test Set} : v_i \in TopN(v_u, v_c, v_t)|}{|v \in \text{Test Set}|} \quad (2)$$

We executed experiments for $N=1$, $N=3$ and $N=5$.

5.3 Compared Methods

We executed our algorithm, from now on dubbed *CtxOrd*, using different combinations of context dimensions and program attributes, with the aim of evaluating their usefulness in the generation of the recommendations. In particular, we tested the non-contextual alternatives which build the tensor T exploiting the sole channel and the sole genre. Then, we tried to enrich the tensor with the various context dimensions.

Our algorithm was compared also with a naive non-contextual and non-personalized methodology, dubbed *TopPop*, recommending to each user, in each context, the list of programs broadcast on the N channels that were globally the most seen.

Finally, we considered another less trivial, non-contextual and non-personalized competitor, named *ShortestTimeSinceStart*, always compiling the recommendation list with the programs started since the shortest time.

We had performed some trials also using traditional collaborative filtering. However, as explained in the introduction of the paper, the dynamism of the item catalog makes such techniques ill-suited for TV program recommendation, and indeed the obtained results were extremely poor. Therefore, we do not show collaborative filtering results in the following sections.

Table 3. Recall@1

Algorithm	All	≥ 8 chan	≥ 28 chan
TopPop	19.26 %	17.65	10.28 %
ShortestTimeSinceStart	3.46 %	15.89 %	7.34 %
CtxOrd – Channel	33.04 %	22.93 %	10.92 %
CtxOrd – Genre	13.91 %	12.87 %	5.85 %
CtxOrd – Channel, Genre	31.57 %	13.96 %	6.56 %
CtxOrd – Fam. Cont., Channel	33.95 %	31.92 %	19.33 %
CtxOrd – Fam. Cont., Genre	8.56 %	7.94 %	3.22 %
CtxOrd – Fam. Cont., Channel, Genre	33.20 %	22.94 %	13.96 %
CtxOrd – Day, Time, Channel	39.23 %	37.02 %	22.36 %
CtxOrd – Day, Time, Genre	18.11 %	16.75 %	1.44 %
CtxOrd – Day, Time, Channel, Genre	32.83 %	30.84 %	19.16 %
CtxOrd – Fam. Cont., Day, Time, Channel	39.11 %	36.34 %	21.77 %
CtxOrd – Fam. Cont., Day, Time, Genre	17.38 %	15.96 %	7.58 %
CtxOrd – Fam. Cont., Day, Time, Channel, Genre	33.29 %	31.31 %	18.88 %

All the experiments were repeated three times, considering three different compositions of the test set:

- The whole test set (7921 users, 1086564 program views).
- Subset obtained excluding the users who have shown to be not very active, having watched only 7 channels or less (5824 users, 959141 program views).
- Subset obtained including only the very active users, having watched 28 channels or more (201 users, 51525 program views).

5.4 Results

In this subsection we present in a tabular form the results obtained with the experimented methodologies. Tables 3, 4 and 5 show, respectively, Recall@1, Recall@3 and Recall@5. The tables are divided in two parts: the upper one shows the non-contextual techniques while the lower one shows the contextual ones.

5.5 Result Analysis

In the following the results reported in the tables are analyzed in detail, starting with the whole test set and then considering the reduced ones.

Full Test Set. A first aspect which can be noticed from the results is that the differences between context-aware and baseline methodologies are larger when recommending few items. This happens because many users watch just a limited

Table 4. Recall@3

Algorithm	All	≥ 8 chan	≥ 28 chan
TopPop	45.19 %	43.17 %	24.98 %
ShortestTimeSinceStart	11.33 %	33.03 %	17.77 %
CtxOrd – Channel	63.12 %	41.22 %	23.38 %
CtxOrd – Genre	30.83 %	28.78 %	13.93 %
CtxOrd – Channel, Genre	60.29 %	35.51 %	19.06 %
CtxOrd – Fam. Cont., Channel	63.94 %	61.38 %	38.93 %
CtxOrd – Fam. Cont., Genre	20.72 %	19.41 %	8.53 %
CtxOrd – Fam. Cont., Channel, Genre	49.47 %	46.44 %	29.86 %
CtxOrd – Day, Time, Channel	67.05 %	64.59 %	42.11 %
CtxOrd – Day, Time, Genre	37.94 %	35.72 %	7.33 %
CtxOrd – Day, Time, Channel, Genre	60.98 %	58.20 %	37.88 %
CtxOrd – Fam. Cont., Day, Time, Channel	66.94 %	62.12 %	40.70 %
CtxOrd – Fam. Cont., Day, Time, Genre	36.66 %	34.42 %	17.47 %
CtxOrd – Fam. Cont., Day, Time, Channel, Genre	62.00 %	59.26 %	38.14 %

Table 5. Recall@5

Algorithm	All	≥ 8 chan	≥ 28 chan
TopPop	59.93 %	57.81 %	31.14 %
ShortestTimeSinceStart	18.07 %	45.27 %	25.51 %
CtxOrd – Channel	77.38 %	51.19 %	30.64 %
CtxOrd – Genre	42.07 %	39.17 %	19.83 %
CtxOrd – Channel, Genre	65.18 %	48.53 %	27.72 %
CtxOrd – Fam. Cont., Channel	77.93 %	74.09 %	51.23 %
CtxOrd – Fam. Cont., Genre	73.89 %	30.14 %	14.09 %
CtxOrd – Fam. Cont., Channel, Genre	63.17 %	60.26 %	39.81 %
CtxOrd – Day, Time, Channel	78.58 %	76.56 %	52.65 %
CtxOrd – Day, Time, Genre	49.82 %	47.14 %	15.45 %
CtxOrd – Day, Time, Channel, Genre	75.31 %	72.96 %	48.90 %
CtxOrd – Fam. Cont., Day, Time, Channel	78.67 %	72.30 %	51.00 %
CtxOrd – Fam. Cont., Day, Time, Genre	48.26 %	45.59 %	24.32 %
CtxOrd – Fam. Cont., Day, Time, Channel, Genre	76.29 %	74.03 %	49.67 %

number of channels, and therefore even simple strategies are able to identify the proper program in lists containing several items.

Let us consider the non-contextual alternatives, above the horizontal line in the tables. We immediately note that the non-personalized methods TopPop and ShortestTimeSinceStart show very poor performance, while the personalized

model based on the channel obtains very high recall. This suggests that the users' preferences are more important than the time elapsed since the program started to determine the right suggestion. Note also that the results for the personalized model relying on the genre are not good. The usage of the genre seems to confuse the system instead of helping; in fact, adding the genre to the model based on the channel brings disturbance instead of improvement.

Consider now the contextual models, below the horizontal line in the tables. First, we observe that also in this case the models with the channel behave better than those envisaging the genre, that again seems to confuse the system. The addition of the familiar context brings some improvements, but these are really small: the recall increase is less than 1 % with respect to the model based only on the channel. A significant gain, on the contrary, is provided by the usage of date and time. The best-performing model – the one including day, time and channel – improves the non-contextual alternative based on the channel of 6.19 % for Recall@1, 3.93 % for Recall@3 and 1.20 % for Recall@5; as explained above, the shorter the recommendation list, the larger the recall increment. The addition of the familiar context to the model envisaging day, time and channel does not provide significant improvements, with the exception of Recall@5.

The fact that the best model is the one envisaging day, time and channel, together with the good performance shown by the non-contextual model with only the channel, suggests that the habit factor is very important in the choices of TV viewers: many users watch very often the same channels in the same time slots.

Test Sets Obtained Excluding the Less Active Users. In this case we note that the differences between the methodologies are wide also for Recall@3 and Recall@5: this happens because the users in these test sets are used to see many channels, and so it may be difficult to discover the right program to be suggested even through long lists of recommendations.

Moreover, the negative results of the models including the genre of the program are confirmed also in this case.

In general, for each experimented model, the recall value decreases with respect to that measured with the same models on the whole test set, again because these users have seen several channels and so the recommendation is more difficult. An exception is represented by the non-personalized methodology *ShortestTimeSinceStart*, for which the recall obtained for the active users is greater than that achieved on the full test set. This is an interesting result, and seems to suggest that the active users are more resolved in the choice of TV programs: they know what they want to watch and change the channel when they know it is starting. The other users, on the contrary, seem to proceed in a more random way among the few channels they are used to take into account.

The two subsets of active users confirm that the contextual strategies show better performance than the non-contextual ones. The best model is again that envisaging day, time and channel, and the increment with respect to the recommendations generated considering only the channel is even larger than that

registered with the full test set. For instance, in the test set containing only the users having seen at least 28 channels, the increments are 11.44 % for Recall@1, 18.73 % for Recall@3 and 22.01 % for Recall@5.

Differently from what we observed in the experiments with the full test set, in this case the familiar context introduces a significant increment in the quality of recommendations. For instance, when the test set containing only the users having seen at least 28 channels is taken into account, the increments with respect to the model envisaging only the channel are 8.41 % for Recall@1, 15.55 % for Recall@3 and 20.54 % for Recall@5. However, the contribution of the familiar context is canceled when the familiar context is considered in addition to day and time. This means that the effect of the users' habits remains stronger than the impact of the familiar context, also for the active users.

5.6 Summary of the Evaluation

The described experiments showed that our methodology can provide accurate recommendations to TV users relying exclusively on implicit feedback. In addition, the experiments proved that in the considered scenario the context is decisive in the recommendation process. In more detail, the day and time context dimensions showed to be relevant for all the users, while the familiar context proved significant only for the most active users.

6 Conclusion

This paper has proposed a content-based context-aware technique to provide TV program recommendations relying exclusively on implicit feedback. An extensive evaluation on a real TV dataset has been carried out, showing the effectiveness of the proposal.

Several directions for future works exist. First of all, in this work we have exploited counters of the number of seconds which the users have spent in seeing certain programs in certain conditions, weighting each second in the same way. However, after some time it is possible that the TV is let turned on when the user has started other activities or has fallen asleep. Therefore, it could be interesting to modify the construction of our model by introducing a decay factor able to weight more the first seconds of view. Other relevant research possibilities concern the study of strategies to increase novelty and serendipity of TV recommendations.

References

1. Burke, R.: Hybrid web recommender systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) *Adaptive Web 2007*. LNCS, vol. 4321, pp. 377–408. Springer, Heidelberg (2007)
2. Dey, A.K.: Understanding and using context. *Pers. Ubiquit. Comput.* **5**(1), 4–7 (2001)

3. Engelbert, B., Blanken, M.B., Kruthoff-Brüwer, R., Morisse, K.: A user supporting personal video recorder by implementing a generic bayesian classifier based recommendation system. In: Proceedings of the PerCom Workshops, pp. 567–571. IEEE Computer Society, Los Alamitos (2011)
4. Kurapati, K., Gutta, S., Schaffer, D., Martino, J., Zimmerman, J.: A multi-agent tv recommender. In: Proceedings of the 1st International Workshop on Personalization in Future TV (2001)
5. Chang, N., Irvan, M., Terano, T.: A TV program recommender framework. In: Proceedings of KES 2013, 17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems, pp. 561–570. Elsevier, Amsterdam (2013)
6. Barragáns-Martínez, A.B., Pazos-Arias, J.J., Fernández-Vilas, A., García-Duque, J., López-Nores, M.: What's on TV tonight? An efficient and effective personalized recommender system of TV programs. *IEEE Trans. Consum. Electron.* **55**(1), 286–294 (2009)
7. Ali, K., van Stam, W.: TiVo: making show recommendations using a distributed collaborative filtering architecture. In: Proceedings of KDD 2004, 10th International Conference on Knowledge Discovery and Data Mining, pp. 394–401. ACM, New York (2004)
8. Cotter, P., Smyth, B.: PTV: Intelligent personalised TV guides. In: Proceedings of the 17th National Conference on Artificial Intelligence and 12th Conference on Innovative Applications of Artificial Intelligence, pp. 957–964. AAAI Press / The MIT Press, Cambridge (2000)
9. Überall, C., Rajarajan, M., Rakocevic, V., Jäger, R., Köhnen, C.: Recommendation index for DVB content using service information. In: Proceedings of ICME 2009, International Conference on Multimedia and Expo, pp. 1178–1181. IEEE Computer Society, Los Alamitos (2009)
10. Ardissono, L., Gena, C., Torasso, P., Bellifemine, F., Difino, A., Negro, B.: User modeling and recommendation techniques for personalized electronic program guides. In: Ardissono, L., Kobsa, A., Mark, T. (eds.) *Personalized Digital Telev.*, pp. 3–26. Springer, Heidelberg (2004)
11. Hsu, S.H., Wen, M.-H., Lin, H.-C., Lee, C.-C., Lee, C.-H.: AIMED- A personalized TV recommendation system. In: Cesar, P., Chorianopoulos, K., Jensen, J.F. (eds.) *EuroITV 2007*. LNCS, vol. 4471, pp. 166–174. Springer, Heidelberg (2007)