

Methods of Machine Learning for Linear TV Recommendations

Mikhail A. Baklanov¹(✉) and Olga E. Baklanova²(✉)

¹ Tomsk State University, Tomsk, Russia
baklanov.ma@gmail.com

² D.Serikbayev East-Kazakhstan State Technical University,
Ust-Kamenogorsk, Kazakhstan
OEBaklanova@mail.ru

Abstract. This paper describes methods of improving TV-watching experience using Machine Learning for Linear TV recommendations. There is an overview of existing methods for video content recommendations and an attempt of developing new method that focused only on linear TV recommendations and takes into account all specifics around it. Recommendation system based on this approach was implemented in Russian pay TV provider ZOOM TV, and demonstrated two times churn rate reduction in comparison with same service without recommendation system. Existing methods and new method effectiveness compared with offered approach by analyzing real people content consumption during 1 year.

Keywords: Recommenders · Linear TV · SMART TV · Collaborative filtering · Content filtering · Machine learning

1 Introduction

There are 900 million pay TV customers all over the world. These people pay for hundreds of TV-channels but watch only 10–15 of them and spend about 25 % of TV watching time on channel surfing. The content discovery on TV did not changed since it was invented and anyone who watch linear TV need to zap channel one by another to find something interesting. That worked good while there were only 10 channels but it is not very pleasant user experience when TV viewer needs to check 100–200 channels manually. Popular content discovery techniques like text search and recommendations from friends are not efficient enough on TV because of limited input means.

It must be admitted that for video-on-demand (VOD) and for linear TV this problem has different important traits. VOD is rather young way of content consumption, and in terms of content discovery, there are several different use cases. The most popular use case that many researchers are trying to solve is “to find something new and interesting”. In theory, if an algorithm knows how strong person enjoyed other movies, it is possible to evaluate how strong he will appreciate the new one.

For linear TV most popular case is different. Usually TV-watcher turns on the TV set and want to switch to the most interesting or less annoying TV-show that is on air now. The usage of ordinary machine learning algorithms is complicated by various

factors such as poor input devices, the novelty of TV-shows, patterns of content consumption and the fact that TV-set is not a personal device, it used by a whole family or household. But from the other hand the content on TV-channels already preselected by professional producers and editors, for most cases it is relevant to its target audience.

This paper aims to analyze linear TV content specifics and develop the most effective method for linear TV recommendations.

2 Problem Formulation and Motivation

Recommendation system aims could be different and depends on business needs of the organization that is trying to integrate it. Anyway, usually it aims to improve some of the KPIs, which could make product more profitable or sustainable. That is why it is important to measure the effectiveness of recommendation system not only by some statistical measures like standard deviation but also use some business KPI to make sure that developed system is close enough to real life and demanded.

One of the first papers on linear TV recommendation systems is [1], which describes a personalized Electronic Program Guide. Many works present collaborative and content filtering based approaches [2].

In [4] authors proposed a mechanism that models, records and analyzes users' traces. It allows evaluating competencies for recommending people with more expertise on a certain subject. R. Turrin et al. in [3] are trying to extend common recommendation methods by integrating the current watching context into the user viewing model. In [5] P. Cremonesi et al. are focused on Implicit Feedback and emphasize the specifics of linear TV watching that makes it different from Video-on-Demand such as:

- Available items change over time: many TV programs are often broadcast once and then not anymore for a long time.
- Time-constrained catalog of items: all programs are transmitted in a predefined schedule. Therefore, the recommendations must consider only the items on air at moment in which they are requested.
- The user feedback is usually implicit, provided in the form of watched/not watched shows.

D. Zibriczky et al. [6] also specify different features of linear content paying special attention that the most common device for consuming linear TV-content is TV-set. Their paper is focusing on the following recommendation problems:

- Noisy input: as TV watching is a very, very laid-back environment, users do not provide explicit feedback. The taste is expressed implicitly by channel zapping, recording, recording playback events. One key challenge is how to pre-process and identify the relevant user actions that can be efficiently is used for user profiling.
- Live TV programs are always new: by nature, live TV programs are always new. Recommender algorithms that require user feedback for preference modeling cannot be used, since such data is not available before the program start, and when it becomes available, it is too late, the program is over. The metadata associated with live programs is also much less detailed and relevant to on-demand programs.

About 60 % of live TV programs belong to news, sports, reality, talk shows, and many of them are broadcasted in live. Either the program descriptions are the same or very similar that makes the recommendation problem very challenging.

- Time-based recommendation: consumption patterns changes on daily and weekly bases. Users are not interesting in the same type of content in the morning and in the evenings, and it is also true for weekends.
- Shared, multi-user device: TV is a shared, multi-user device. The algorithms should be aware to this problem and if there is no explicit indication, who is in front of the device, it should also predict.

This paper analyzes recommenders from the viewpoint of customer loyalty, which can improve such important business metrics like ARPU and churn rate. So the problem is how to make a recommender that will be efficient enough to make it users more satisfied with a product.

Let U - number of users and S - the set of all objects that can be recommended. Let h - a function that measures user satisfaction, i.e. the extent to which a subject s like user u . Thus, for each user $u \in U$ we want to select an object $s' \in S$ that would maximize user satisfaction. To put it more formally:

$$\forall c \in C, s'_c = \arg \max_{u \in U} h(u, s). \quad (1)$$

3 Common Machine Learning Methods

The basic idea of recommender is to predict what user will prefer and how strong will he or she satisfied by an object. All existing methods are start from the attempt to learn from user's actions and understand their habits and tastes. The strategy should be based on the information which recommender can access and objective function.

3.1 Item-Based Methods

Content Filtering (item-based) involves the creation of user profiles and objects. In order to take into account relevant recommendations of the object parameters corresponding user preferences. The recommendation system objects are described using keywords (tags), and create a profile that characterizes its attitude to certain objects.

In the item-based systems, the function of satisfaction $h(u, s)$ user u of a particular content item s is determined on the basis of information on user satisfaction with the content item $s_i \in S$, which are "similar" to s . For example, the recommendation system for films to recommend movies to the user u , the system tries to understand what is common between the films having the user previously praised highly. After that, the user will be recommended movies as much as possible similar to the ones that the user has given high marks.

In fact, content that the user consumes forms a profile in the form of a plurality of parameters characterizing the object s . As a parameter often, protrude keywords corresponding weights for each object. Thus, the problem arises of how to weigh these

parameters. One of the most prominent ways to determine the weights of keywords in information retrieval is TF-IDF [7] measure.

3.2 User-Based Methods

Collaborative filtering uses a well-known group of users' preferences to predict the unknown preference of another user. The basic assumption is that those who are equally evaluated objects of any kind in the past tend to give a similar assessment of other subjects in the future.

In contrast to the Item-based methods, User-based methods try to predict user satisfaction with an object on the basis of how other users rate this item. Those users' satisfaction of objects s $h(u, s)$ is calculated on the basis of users u_j which for any signs are "similar" to u . An important feature here is that the characteristics, by which the user can be considered similar to the other users, may go beyond the system. For example, social networking profiles or demographic information obtained explicitly, can be the basis to assess the similarity of the new primary user of existing ones. This approach is often used to solve "cold" start problem.

3.3 Hybrid Recommendation Systems

Hybrid recommendation systems are a combination of content filtering and collaborative filtering. Some of recommendation systems use hybrid methods combining user-based and item-based methods that partially or completely neutralize some of the disadvantages of each approach. You can select the following ways of combining different approaches in hybrid systems of reference:

1. The introduction of item-based and user-based methods separately and combining them received from assessment;
2. The introduction of some item-based settings in the user-based approach;
3. The introduction of some user-based settings in the item-based approach;
4. Creation of a general model, which uses parameters of both approaches.

One way to build hybrid recommender system is the implementation of two independent item-based and user-based systems. Then there are two possible scenarios. In the first, we can combine the ratings received from each system using a linear combination of ratings [8] or the scheme described in [9]. The second scenario is to use only one of the systems depending on what herein will work better. For example, the system «DailyLearner» [10] selects recommender systems, which can give a recommendation to the lowest level of error.

The most popular approach to this category of hybrid recommender systems is a decrease of dimension for the group of content-based user profiles. For example, in [11] uses a latent semantic analysis (LSA), to create user profiles Collaborative representation, where the user profile is represented as a set of vectors, which leads to a significant improvement in performance compared to pure item-based approach.

To construct the general advisory model is most commonly used user-based and item-based parameters (for example, the age of the users and the genre of the film) in a single advisory system.

4 Recommender Based on Personal Choice Strategies

In contrast to the classical problem of the recommendations of video on demand, which is available at any time, the aim of this work, in fact, are recommendations for changing the current time on any channel of an arbitrary limited list of channels. This clarification is so significant that even with the use of electronic program recommendation algorithms that work well in video on demand, here is difficult.

The main idea of this approach is to take into account how person choose what to watch. Analyzing the history of channel switching it is possible to make for every device for each timeslot it has the vector with specific coefficients that models how important for a household this or that tag. For example, for somebody it could be important to watch sports live and for another person the most important is to watch English Premier League. For first person the biggest impact will be made by the tag “live”, for the second person the tag “league”.

The final rating of transmission (Rbr_s) for any user (u) is different and, in general, is calculated as follows:

$$Rbr_{su} = Q_{su} + P_{su}, \tag{2}$$

$$Q_{su} = Q_{su-1} + a_{su}l_{su} + \sum_{j=1}^n (b_{su}t_{isu} + c_{su}w_{isu}), \tag{3}$$

$$P_{su} = \sum_{j=1}^m k_{jsu}tag_{jsu}, \tag{4}$$

where:

- $l_s \in \{-1, 0, 1\}$ and indicates the presence of a mark by the user that this transfer him like or dislike;
- $t_i, w_i \in \{0, 1, 2, \dots, n \dots, \infty\}$ and denote, respectively, how many times were used pause and rewind within it (timeshift), and how many issues the user looked through this transfer to the end;
- a, b, c, k_j – coefficients are calculated for each individual user on the basis of how great value to the user preferences;
- i – number of an episode;
- j – number of a parameter;
- tag_j – value of parameters for TV-show s ;
- s – number of TV-show;
- n – the number of views made by user for all episodes of one TV-show;
- m – the number of parameters that are characteristic of this transmission;
- Q_s – the number of parameters specific to a given TV-show;
- P_s – the sum of the weights of all the parameters of the TV-show multiplied by

Unlike most popular approaches related to the clustering of users and the issuance of recommendations based on the user’s membership in a particular group, this approach presume absolute individual tastes user and provides this calculation of the importance of particular criteria on which man rests in the choice of transmission. The importance of a fixed coefficient, the larger the ratio is the higher the importance of this coefficient.

5 Results and Discussion

In this part will be analyzed and compared the effectiveness of the “classical” approach and an approach based on filtering and ordering. As the representatives of the “classical” approach using collaborative filtering algorithms based on users and object-based.

The study will take place on the data collected from more than 300,000 users of the Pay TV for 1 year of work. These data provide a history of decision-making by the user on which TV-show to watch, the order in which the channels have been submitted to it and user actions according these TV-shows.

As a measure of efficiency is considered the number of actions that are necessary for the user to find the most interesting programs for each method on average.

The most interesting programs here and further we consider the transfer, the user ultimately chose. We believe that the user has chosen the transfer, if it is:

- watched it for 5 min and looked through to the end (± 2 min);
- noted the transfer as a favorite;
- use a pause or rewind.

To implement cloud platform will be used Microsoft Azure Machine Learning Studio, which allows users to run in the cloud and scale the infrastructure for big data (Big Data). Users at any given time have access to 80 channels.

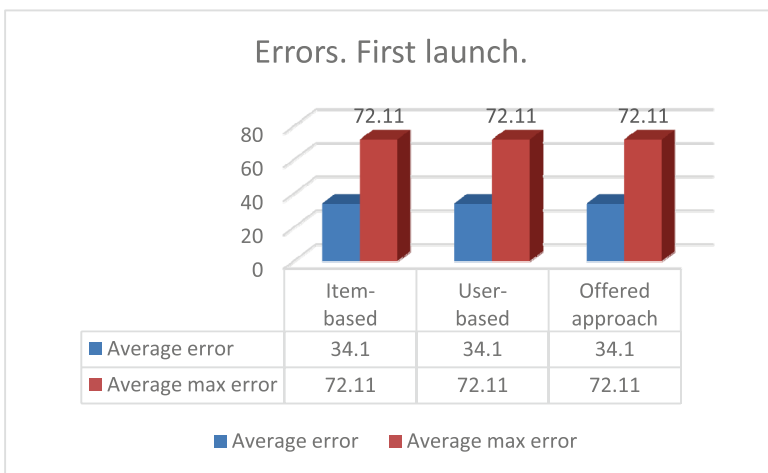


Fig. 1. Errors. First launch (Color figure online)

For the analysis, how to lead the above methods of the recommendations used a sample of users who have not specified before using the categories that they are not interesting.

At first launch shown on Fig. 1, due to the lack of any history all systems equally ineffective.

Data accumulated over the week, it enough that each of methods significantly reduces errors compared with random TV-shows ordering. As can be seen from the chart, the offered method provides delivery much closer to what the user wants to select, but the maximum error for users is still very large. It is shown on Fig. 2.

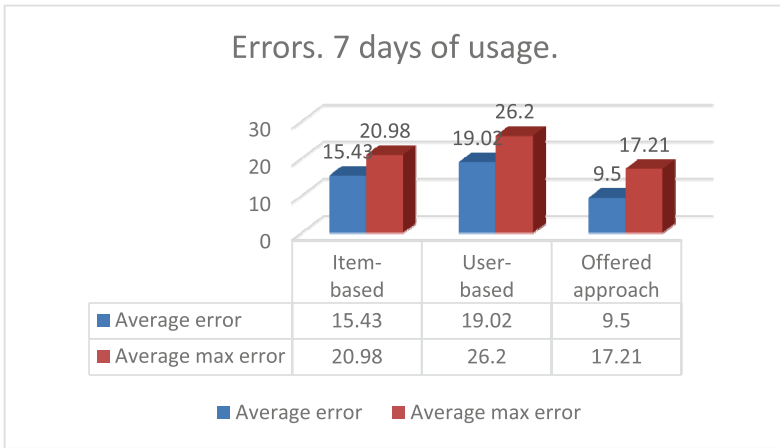


Fig. 2. Errors. Second launch (Color figure online)

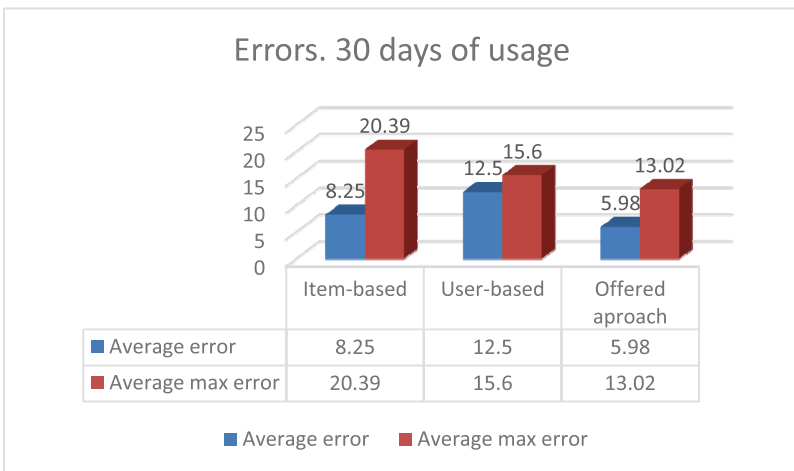


Fig. 3. Errors. Third launch (Color figure online)

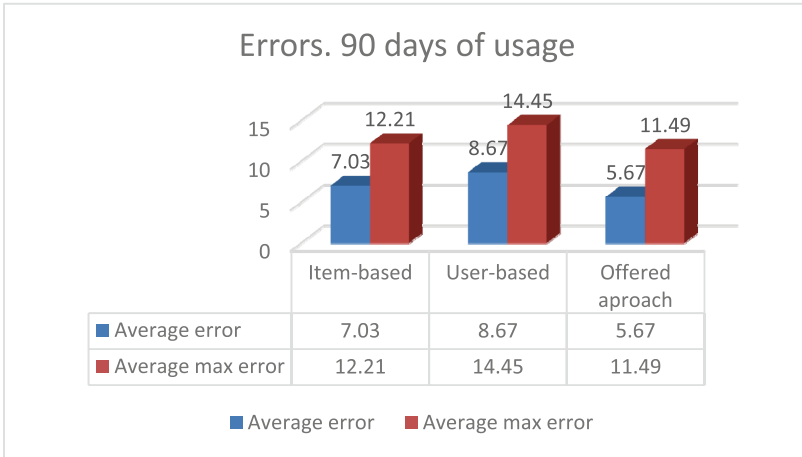


Fig. 4. Errors. Forth launch (Color figure online)

For thirty days, all the algorithms have shown significant progress, compared with 7 days. It is striking that the Item-Based approach, despite the significant decrease in the average error continues to send some people almost in the middle of the list. It is shown on Fig. 3.

The offered algorithm is slightly improved its results and, despite progress on the part of the other two, still shows the best results both in the middle and on the average maximum error. It is shown on Fig. 4.

6 Conclusion

The results showed that the offered approach is much better suits for specific tasks of recommending linear content. It allows recommendations be personal in terms of understanding persons tastes and allows to work effectively with premier TV-show that was never shown before to anybody just by analyzing meta-information and TV-viewer tastes that was demonstrated before. That makes recommendation logics work more like human do. Recommendation system based on this approach was implemented in Russian pay TV provider ZOOM TV, and demonstrated two times churn rate reduction in comparison with same service without recommendation system.

References

1. Das, D., Horst, H.: Recommender systems for TV. In: Workshop on Recommender Systems, Proceedings of 15th AAAI Conference, Madison, Wisconsin, pp. 35–36 (1998)
2. Ali, K., Stam, V.: TiVo: making show recommendations using a distributed collaborative filtering architecture. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2004, pp. 394–401 (2004)

3. Turrin, R., Condorelli, A., Cremonesi, P., Pagano, R.: Time-based TV programs prediction. In: 1st Workshop on Recommender Systems for Television and Online Video at ACM RecSys (2014)
4. Wang, N., Abel, M.-H., Barthes, J.-P., Negre, E.: A recommender system from semantic traces based on bayes classifier. In: Proceedings of the 2nd International Conference on Knowledge Management, Information and Knowledge Systems (KMIKS), Hammamet, Tunisia, pp.49–60, April 2015
5. Cremonesi, P., Modica, P., Pagano, R., Rabosio, E., Tanca, L.: Personalized and context-aware TV program recommendations based on implicit feedback. In: Stuckenschmidt, H., et al. (eds.) EC Web 2015. LNBIP, vol. 239, pp. 57–68. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-27729-5_5](https://doi.org/10.1007/978-3-319-27729-5_5)
6. Zibriczky, D., Hidasi, B., Petres, Z., Tikk, D.: Personalized recommendation of linear content on interactive TV platforms: beating the cold start and noisy implicit user feedback. In: TVMMP: Proceedings of the International Workshop on TV and Multimedia Personalization (TVMMP) held UMAP 2012, Montreal, Canada (2012)
7. Salton, G.: Automatic Text Processing. Addison-Wesley, Reading (1989)
8. Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., Sartin, M.: Combining content-based and collaborative filters in an online newspaper. In: Proceedings ACM SIGIR 1999 Workshop Recommender Systems: Algorithms and Evaluation, August 1999
9. Pazzani, M.: Framework for collaborative, content-based, and demographic filtering. *Artif. Intell. Rev.* **13**, 393–408 (1999)
10. Billsus, D., Pazzani, M.: User modeling for adaptive news access. *User Modeling User-Adap. Interact.* **10**(2-3), 147–180 (2000)
11. Soboroff, I., Nicholas, C.: Combining content and collaboration in text filtering. In: Proceedings International Joint Conference on Artificial Intelligence Workshop: Machine Learning for Information Filtering (1999)