User Behaviour Analysis and Personalized TV Content Recommendation

Ana Carolina Ribeiro[1][0000-0003-1557-654X], Rui Frazão[2] and Jorge Oliveira e Sá[3][0000-0003-4095-3431]

[1] University of Minho, Guimarães, Portugal
[2] University of Aveiro, Aveiro, Portugal
[3] University of Minho, Guimarães, Portugal
[1]anacfr1@hotmail.com, [2]ruifilipefrazao@ua.pt, [3]jos@dsi.uminho.pt

Abstract. Nowadays, there are many channels and television (TV) programs available, and when the viewer is confronted with this amount of information has difficulty in deciding which wants to see. However, there are moments of the day that viewers see always the same channels or programs, that is, viewers have TV content consumption habits. The aim of this paper was to develop a recommendation system that to be able to recommend TV content considering the viewer profile, time and weekday.

For the development of this paper, were used Design Science Research (DSR) and Cross Industry Standard Process for Data Mining (CRISP-DM) methodologies. For the development of the recommendation model, two approaches were considered: a deterministic approach and a Machine Learning (ML) approach. In the ML approach, K-means algorithm was used to be possible to combine STBs with similar profiles. In the deterministic approach the behaviors of the viewers are adjusted to a profile that will allow you to identify the content you prefer. Here, recommendation system analyses viewer preferences by hour and weekday, allowing customization of the system, considering your historic, recommending what he wants to see at certain time and weekday.

ML approach was not used due to amount of data extracted and computational resources available. However, through deterministic methods it was possible to develop a TV content recommendation model considering the viewer profile, the weekday and the hour. Thus, with the results it was possible to understand which viewer profiles where the ML can be used.

Keywords: Recommender Systems, Machine Learning, User Behaviour Analytics.

1 Introduction

Currently, consumers have access to a massive quantity of information about lots of products everyday, which makes the decision-making of choosing to process harder. This problem is known in technical literature as "Information Overload", which refers to the fact that there are finite limits to the ability of humans to assimilate and process

information [1]. This is considered a major difficulty in decision-making process in many fields.

Specifically, in television (TV) consumption, growing number of channels available leads to a more complex and time-consuming choice of content to the viewer. In this paper, the user behind the screen is called 'viewer'. With the increase of channels, zapping and TV programming magazines are not effective in the selection of content [2].

Thus, the objective of this paper is to develop a model capable of describing and inferring the preferences of TV content of viewers for a selection more personalized, based on the records activity of Set Top Boxes (STBs). STBs do not present user profiles, which means that if there is more than one viewer using STB, it is not possible to differentiate. To overcome this situation, it was decided to analyse each activity record of STB per hour.

This paper aims to develop a model that can describe a viewer in each time-slot by using information from the preferences profile. The viewer behaviour will be analysed to be adjusted to a behaviour profile that will allow to quickly identify the type of content he is looking for.

In this way, the goal of this paper is the construction of a prototype that, for each STB and in each time-slot, choose one of the three types of solutions:

1. When time-slots do not have enough visualizations to infer who is viewing, does not perform a recommendation.
2. When the time-slot history shows a regular pattern of visualizations, this allows to make a prediction of TV content with a very high probability of being accepted by the viewer.
3. When the time-slots history shows a complex pattern of visualizations, in these situations, there is a high probability that the Machine Learning (ML) techniques will work.

For the recommendation system development, two approaches are considered: a deterministic approach and ML based approach with K-Means algorithm.

The Design Science Research (DSR) methodology enables the creation and evaluation of information technology artefacts to solve organizational problems and involves a rigorous process of developing artefacts to solve the identified problems, contributing to the research and evaluating the projects [3]. This paper aims to create an artefact based on deterministic or ML approaches for an effective recommendation of TV contents to viewers. Thus, this paper takes in to account the guidelines of the DSR in parallel with the data mining methodology CRISP-DM [3].

This paper is organized as: section 2 describes the related work; in section 3 the data available, the most important features and the data statistical analysis performed is presented; section 4 describes the recommendation model development and ML technologies used; section 5 presents the results obtained from the recommendation model developed and the evaluation of the results. Finally, the conclusions and future work of this paper are summarized in section 6.

2 Related Work

2.1 Recommender Systems

Since the world is becoming more and more digital, it is considered the existence of a parallel between humans and technology: on the one hand, individuals use more and more technology, and on the other, digital systems have become more and more centred on the user. This way, the systems should allow users to be able to synthesize information and explore the data [4].

Therefore, there is a need for computing techniques that facilitate this research and the extraction of information in the interest of the user. One of the solutions to this problem is the use of ML techniques to find explicit and implicit patterns of user preferences, for the purpose of customizing the search for content of the user's interest [1].

An approach used to the suggestion of the content of the user's interest is the recommendation systems [5]. A recommendation system can be defined as any system that provides the user with recommendations of services, products or certain potentially interesting content. To provide suggestions and help users in decision-making, the recommendation systems should be include some characteristics such as users' needs, their difficulties, goals, preferences and some know-how about domain of business [4], [5]. They consist on the capability of providing suggestions for items[1] [5] .

There are several recommender systems, but the most used are content-based recommender systems, collaborative and hybrid systems [1]:

Content-based Systems – systems that try to recommend new items that are like items that a user has shown interest in the past.

Collaborative Filtering systems – the recommendations are based on the analysis of the similarity between users. The suggested items are those that users with similar preferences have had an interest in the past.

Hybrid systems – systems that implement a combination of two or more recommendation techniques. These systems try to take advantage of all techniques used to improve the performance of the system and reduce the disadvantages of each technique used individually.

The interest in the recommender systems is increasingly high, due to the growing demand in applications capable of providing personalized recommendations and dealing with information overload [5]. Some challenges and limitations can be found in the recommendation systems, namely:

Cold-start - There are some situations in which the lack of data causes the recommender system not to make recommendations or the recommendations generated do not present a high level of confidence [6]. For example, in content-based filtering, it is necessary for the system to have access to the user's interests in the past, to decide which items are like those. This problem may occur because of the addition of new users or a new item [6].

[1] "Item" is the general term used to denote what the system recommends to users. Products, movies, music and news are some examples of what can be recommended.

Data dispersion - Data dispersal is a common problem in most recommender systems since users typically classify only a small proportion of the items available [5].

Limited context - The location, time, date, etc., are some of the context factors that recommender systems should take into consideration. In addition, factors such as user emotion, mood and other parameters should also be considered as they influence users' decisions [5].

2.2 TV Centred Recommender Systems

With the rise of TV content and new functionalities available it was necessary to find adequate tools to help users to choose the content of their interest. Although recommender systems allow users to take an active role and request content on the fly, it also gives the possibility to recommend personalized content based on the users preferences without a prior request [7]. Interactive platforms like Electronic Programming Guide emerged as a tool to help TV consumption. On Video on Demand (VOD) recommender systems emerge as a proposal to improve the process of discovery of new movies, with a relative success and that makes recommender systems have a high importance in the field of TV. These systems tend to have a more effective impact on platforms of Subscription VOD (SVOD), an example of that is Netflix [8].

The development of effective recommendation systems is complex due to some particularities of the TV content. One of the difficulties of systems that have access to a catch-up TV system is that they are constantly entering new content for the catalogue and the older contents are removed due to the time window of the automatic recordings to be limited [8].

An important factor in TV recommender systems is time. For example, a viewer's favourite movie can be displayed in a channel while the viewer is watching another program with less interest, so this is the right time to suggest the movie to the viewer if the recommender systems not suggest the movie to the viewer at right time, this recommender system becomes an imprecise recommender system with high cost to maintain and users tend to disable this kind of functionality [9].

3 Data Analysis

The life cycle of CRISP-DM methodology consists of 6 phases: business understanding, data understanding, data preparation, modelling, evaluation and implementation and the sequence of the phases is not rigid [10].

In the data understanding phase of CRISP-DM, it was found that data by the STBs correspond to 5 months of registers (from January to May) of 2017 of a total of 1.5 million STBs. For this paper, data were provided by a telecommunications organization in Portugal. The data provided presents different types of information about TV contents. To complete the data understanding phase, some data statistical exploration was performed to find out mistakes, missing values and to know the attributes meanings. Initially, the data distributions were analysed to know the normal patterns from population analysed so that, when extracting samples for experimentation, it was possible to

evaluate if these would be representative of the remaining population or not. Some examples are given below.

It was calculated the distribution of viewing time in hours and per day, for all STBs between January and March. With this distribution, it was possible to observe that there are regularly higher values corresponding to weekend values, that is, viewers see more TV at the weekend. This result corresponds with reality because, in general, people have more free time on the weekend. An analysis was also performed about the content viewing time, because the number of view records may be high, but the duration time of each record can be very small. In this way, the viewing time of television content is a relevant factor in understanding viewer preferences. From this analysis, it was found that 38% of the records have a viewing percentage of 75%-100%, which means they see a large part of the content or in your totality. These results contrast with the 35% of records that have a viewing percentage between 0-15%, which means that they only see part of the content and where zapping moments can be represented. There is a class that represents views above 100% (views with a time greater than the total time of the program). This phenomenon can occur if the viewer pauses the program for a long period or uses de timeshift functionality and reviews parts of the program. These are just a few examples.

It was also carried out, in the data quality, some inconsistencies were found, such as, missing values and errors (for example, the same program is classified as a series and a program, simultaneously). The identified errors and missing values were reported, and others are corrected.

In consideration of dimensionality of the data, in the data selection phase it was decided to use a sample with only 3 months of 500 STBs that correspond to about 1 million of views. It was decided to select only 500 STBs because the available computational resources were not enough to support the total amount of data and due to the limited time for prototype development. In addition, of the total of 5 months, only 3 (March, April and May) were selected due to the constraints of the available computational resources. Thus, it was decided to exclude January for having only 15 days of records and February for being the shortest in relation to the remainder. Thus, it will be possible to use two months as training and a month of testing. Still in the selection phase of the data were selected some attributes that were considered relevant to the development of the recommendation system, for example: programs, channels, channel thematic, time and weekday of visualization.

Thus, after the phases of understanding and preparing data it is possible to apply modelling techniques to the dataset in the modelling phase, described in the next section.

4 Recommendation Model

4.1 Technologies

Among the numerous ML technologies available, chosen for the development of this project was H2O.ai along with Python programming language. H2O.ai is a Java-based open source, in-memory, distributed, fast, and scalable ML and predictive analytics

platform that allows to build ML models on big data [11]. H2O.ai was recently classified as a leader technology in Gartner's Magic Quadrant for Data Science & ML Platforms [12]. H2O.ai also lacks methods for data manipulation and data visualization compared to the most used python packages for data handling, Pandas, and data visualization, Matplotlib.

In addition to the H2O.ai, two notebooks were used for project development: Apache Zeppelin and Jupyter. Zeppelin is an open-source notebook that allows the ingestion, exploration and data visualization. Zeppelin allows data visualization in various formats allowing the user to get a quick and easy data perception [13]. Jupyter notebook, such as Zeppelin, it is an open-source notebook that allows you to create and share documents with code, visualizations, and narrative text. This notebook provides a suitable web-based application to capture the entire computing process: development, documentation and code execution [14].

Zeppelin was used for the data understanding phase due to the quality it presents in the data visualization. In data processing phase and recommendation system development, Jupyter was favourite by the ability to be used in tasks requiring greater code development as transformation of Data, statistical modelling or machine learning.

4.2 Recommendation model development

In the model development (modelling phase in CRISP-DM), as previously mentioned, two approaches were tested: a deterministic approach and a ML approach.

In the ML approach, a clustering experience was performed with the aim of finding similar visualization profiles through STBs visualization and consequently recommendations are based on the similarities found. For this experience, where the goal is to find data similarities and to group them with these similarities, would be necessary an unsupervised learning algorithm, since data are not previously classified. Given these requirements, the algorithm chosen to apply in modelling phase was the K-means.

In the deterministic approach, viewers behaviors will be analysed to be adjusted to a profile that will allow you to identify the type of content looking for, considering the 3 types of actuation identified in the introduction. Initially, the following profiles were identified:

- **No previews** - STBs do not present visualizations records and, therefore, recommendation is not carried out;
- **Program preference** - STBs present an explicit program preference if the percentage of the content display is equal to or greater than a parameter X, in this case, 70%. In this case, the most viewed program is recommended;
- **Channel preference** - STBs present an explicit channel preference if the percentage of the channel display is equal to or greater than a parameter, in this case 70%. So, the most viewed channel is recommended;
- **No pattern** - STBs present a complex visualizations pattern, without preferences defined.

Still in this approach, after a new problem analyse, it was decided to reformulate the model to increase the capacity of solution (Figure 1). In this new analysis, in addition to the profiles of program preferences and channel preferences, new profiles arose

where the recommendation goes through a set of 3 suggestions of the most viewed programs or channels, that is, in which the sum of the percentages of visualizations is equal or more than a new parameter Y and, in this case, Y=90%. In addition to the previously found profiles, the following have emerged:

- **Top 3 programs –** recommendation of the 3 most viewed programs (sum of the duration of visualizations of the 3 programs must be Y=90%);
- **Top 3 channels -** recommendation of the 3 most viewed channels (sum of the duration of the visualizations of the 3 programs must be Y=90%);
- **Thematic preference** - recommendation of the channel thematic most viewed;
- **No pattern** – no default preference set.

The 'No pattern' profile represents the profiles with an undefined visualizations pattern and recommendation by deterministic methods would not be appropriate. Here, the way of recommending would pass through ML techniques if it was justified to employ machines in this processing, that is, if the percentage of STBs in this profile is significant.

After analysing of these two approaches, section 6 will be presented the results of the two approaches and the justification for which a deterministic approach has been used.

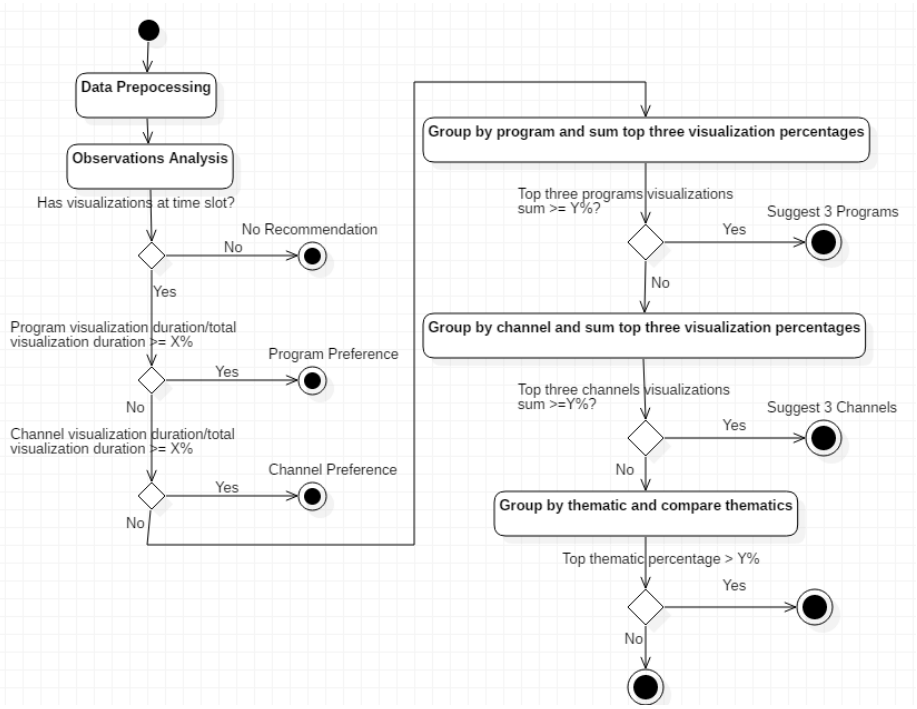

Fig. 1. Recommendation Model: deterministic approach.

5 Result Analysis and Evaluation

In ML approach, the K-means algorithm was trained with records of March and April, for a weekday and time. In Table 1, it is possible to observe some observations that have been grouped because they are similar. The values from 1 to 13 of the Table 1 columns, correspond to the channel thematics. The purpose of this approach is to group the STBs with similar profiles (in this case, considering the channel thematic) on a given day at a certain time. In Table 1, it is possible to verify that, all STBs have a significant visualization percentage of thematic 5, which corresponds to the thematic 'Information'. This means that this set of STBs, on a certain day at a certain time, see the same thematic and, therefore, have been grouped. However, a cost-benefit assessment of the application of this approach was realized, and it was rejected because, given the amount of data and resources available, it would not be possible in the time available for the realization of the project.

Table 1. Cluster observations.

STB / **Thematic**	1	2	3	4	**5**	6	7	8	9	10
1	0.02	0.35	0.00	0.00	**0.61**	0.00	0.00	0.00	0.00	0.00
12	0.00	0.00	0.00	0.00	**0.69**	0.00	0.00	0.30	0.00	0.00
96	0.00	0.00	0.00	0.00	**0.86**	0.00	0.00	0.00	0.00	0.13
130	0.00	0.29	0.00	0.00	**0.70**	0.00	0.00	0.00	0.00	0.00
131	0.00	0.00	0.00	0.00	**0.48**	0.42	0.00	0.00	0.00	0.07

In this way, the model was developed through the deterministic approach. For the application of this model, March and April correspond to the training set and May corresponds to the test set for a sampling of 100 SBTs. The goal is to get through two months of records to predict the content that viewers will see in the following month.

In Table 2 and Table 3 it is possible to observe results obtained from model development. It is possible to verify percentage of cases in which a recommendation is not carried out correctly is 32.68% (Table 3). This value may change with changes in the values of the X and Y parameters of the model (70% and 90%, respectively) and may achieve lower values, making the recommendation more accurate. In Table 3, the percentage of cases where the recommendation cannot be made through deterministic methods correspond to the 'No pattern' profile, that is, corresponds to 7% of the 32.68%. It is necessary, in the future, to assess whether this value is significant. If so, a machine learning recommendation system may be implemented.

About correct recommendations (Table 2), it is verified that the percentage value of the profile 'No visualization' is high. This is an important value because it allows to know which time-slots where it is not necessary to employ resources financial and computational resources to carry out recommendations. Also, 'program Preference ' profile and ' Top 3 Program ' profile present a percentage of correct recommendations lower

than the percentage of incorrect recommendations. This is because it was not possible to use meta-information on the programs of the period studied in the model development.

Thus, about 67% of the recommendations made by the deterministic model are correct.

Table 2. Correct Recommendations

Class	Occur.	Proportions
No visualization	7383	43.95%
Program preference	701	4.17%
Channel preference	908	5.40%
Top 3 programs	762	4.54%
Top 3 channels	1179	7.02%
Thematic preference	377	2.24%
	Total	67.32%

Table 3. Incorrect Recommendations

Class	Occur.	Proportions
Program preference	1564	9.31 %
Channel preference	569	3.39%
Top 3 programs	1519	9.04%
Top 3 channels	544	3.24%
Thematic preference	81	0.48%
No pattern	1213	7.22%
	Total	32.68%

6 Conclusion and Future Work

With the development of this recommendation model, it is noticeable that with only statistical and deterministic methods is possible to make recommendations based on visualization history, making the model less computationally expensive and faster. Even though the parameters have not been optimized, the results seem to fit the expectations for a recommender system on this kind of system. Like most recommender systems, this model needs data to retrieve information about users' preferences and without it a user is not capable of receiving recommendations.

There are some improvements that could be made to improve the recommendation accuracy like standardizing the program titles on the source data, analysing the threshold values used in the model (X and Y parameters) and tune them to achieve better results and reduce the percentage of the "No Pattern" class.

In a next step of this project, an evaluation of the significance of the values of the "No Pattern" class could be made based on the cost-benefit ratio of that operation. Making recommendations to that set of users could be computationally expensive and not financially worth.

Acknowledgement

This work has been supported by COMPETE: POCI-01-0145-FEDER-007043 and FCT (Fundação para a Ciência e Tecnologia) within the Project Scope: UID/CEC/00319/2013 and was developed in partnership with AlticeLabs.

Reference

1. P. Cotter and B. Smyth, “Ptv: Intelligent personalised tv guides,” in Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, 2000, pp. 957–964.
2. M. Soares and P. Viana, “TV recommendation and personalization systems: Integrating broadcast and video on-demand services,” *Adv. Electr. Comput. Eng.*, vol. 14, no. 1, pp. 115–120, 2014.
3. K. Peffers, T. Tuunanen, M. Rothenberger, and S. Chatterjee, “A Design Science Research Methodology for Information Systems Research,” *J. Manag. Inf. Syst.*, vol. 24, no. 3, pp. 45–78, 2007.
4. E. Negre, *Information and Recommender Systems*. John Wiley & Sons, Inc., 2015.
5. R. Francesco, R. Lior, and B. Shapira, *Recommender system handbook*, 1st ed. Springer US, 2011.
6. K. Madadipouya and S. Chelliah, “A Literature Review on Recommender Systems Algorithms, Techniques and Evaluations,” *BRAIN Broad Res. Artif. Intell. Neurosci.*, vol. 8, no. 2, pp. 109–124, 2017.
7. Y. Blanco-Fernandez, J. J. Pazos-arias, A. Gil-Solla, M. Lopez-Nores, and B. Barragans-martínez, “AVATAR : A Multi-Agent TV Recommender System using MHP Applications,” in *IEEE International Conference on e-Technology, e-Commerce and e-Service*, 2005, pp. 660–665.
8. J. Abreu, J. Nogueira, V. Becker, and B. Cardoso, “Survey of Catch-up TV and other time-shift services: a comprehensive anlysis and taxonomy of linear and nonlinear television.,” in *Telecommunication Systems*, 1st ed., 2017, pp. 57–74.
9. J. Oh, S. Kim, J. Kim, and H. Yu, “When to recommend : A new issue on TV show recommendation,” *Inf. Sci. (Ny).*, vol. 280, no. 1, pp. 261–274, 2014.
10. P. Chapman *et al.*, “Crisp-Dm 1.0,” *Cris. Consort.*, p. 76, 2000.
11. H2O.ai, “Welcome to H2O 3 — H2O 3.20.0.3 documentation,” *Welcome to H2O 3*, 2018. [Online]. Available: http://docs.h2o.ai/h2o/latest-stable/h2o-docs/welcome.html. [Accessed: 01-Jul-2018].
12. C. J. Idoine, P. Krensky, E. Brethenoux, J. Hare, S. Sicular, and S. Vashisth, “Magic Quadrant for Data Science and Machine-Learning Platforms,” no. February, pp. 1–26, 2018.
13. A. Zeppelin, “Apache Zeppelin 0.8.0 Documentation:,” *What is Apache Zeppelin?*, 2018. [Online]. Available: https://zeppelin.apache.org/docs/0.8.0/. [Accessed: 01-Jul-2018].
14. Jupyter, “The Jupyter Notebook — Jupyter Notebook 5.5.0 documentation,” 2015. [Online]. Available: https://jupyter-notebook.readthedocs.io/en/stable/. [Accessed: 01-Jul-2018].