Prediction of Video materials offered to a user in a Video-on-demand system

Zoran Gacovski, Gjorgji Ilievski, Sime Arsenovski

FON University, Bul. Vojvodina, bb, Skopje, Macedonia
zoran.gacovski@fon.edu.mk, gjorgji.ilievski@yahoo.com,
sime.arsenovski@fon.edu.mk

Abstract. Prediction of the customer behavior is a subject that is considered to be "the holy grail" in the business. Data mining techniques are not a new subject, but the amount of data that can be processed by the modern computers and the global market that the world has become has opened a lot of opportunities. This paper considers a method for proposal of video materials to the customers in a video on demand (VOD) system, but its broader usage covers any closed system in which the user is identified before the purchase and history of previous user actions is available. By usingthe data from previous purchases in the systemand applying the well-known Apriori algorithm, a set of association rules is generated. An algorithm that uses the history of the client for which the recommendation should be made, compares it with the association rules found previously and produces the prediction for the best fit videos that will be recommended to the customer. The method is simulated using WEKA for the association rules and using T-SQL procedures and functions for the prediction algorithm. Real data from an existing and publicly available VOD (T-home's MAX TV) system is used for the simulation. The data is put in a relational MS SQL database.

Keywords: Data mining, prediction, Apriori algorithm, association rules, video-on-demand, WEKA.

1 Introduction

Making high-quality predictions about the customer's behavior is very important for any business, for planning, marketing, pricing, product management, human resources, training and almost any subject related to a management of business processes. Data mining techniques have been around for a long time, but the development of the IT infrastructure today allows complex data mining techniques to be implemented for a variety of problems. Even problems that require real time response can be modeled and implemented using data mining techniques. At the same time, the implementation costs are reasonable.

VOD systems are closed environments in which offered videos and the users that access them are known. Making a good prediction about the videos that users will rent is crucial for a good and profitable system. It will help both the sales and the procurement of new videos that will be put in the system. There are a few techniques that can be implemented for some prediction to be made, but most of them require

classification of video materials and/or users in groups [1], [3]. Then, the quality of the prediction is dependent on the chosen classifiers as well as the right classification of the videos and the users in them. Another approach is to use the "market basket analysis" that will produce a set of association rules for the videos. The "baskets" will be created from the history of previous rentals of the users [5], [7]. These baskets will be associated with the basket of the target user. His basket will be created from all the videos he has rented in an interval of time ***t***. This interval will be chosen, having in mind the average time the offered videos are available in the video store.

This approach offers one big advantage, dividing the preparation of the data in a useful format and the process of creation of the association rules in an "offline" phase, and making the association "online" when the user is logged on in the system. This makes the process applicable in practice without burdening the existing process of renting videos. The Apriori algorithm is chosen as the algorithm for the preparation of the association rules in the proposed method in this paper. The confidence and the support of the association rules are chosen as the factors that will make the distinction of the valid and not valid association rules. The data used is extracted from a real existing and publicly available VOD system (T-home's MAX TV), for a period of 3 months. The data is part of a relational database created in MS SQL 2007. The Apriori algorithm for the creation of the association rules is implemented in WEKA.

The main objective of our paper is to offer a simple, straight forward model that makes valid and logical predictions in a VOD environment. The model uses a combination of reliable and proven techniques and it is applicable in practice due to its speed, implementation price, simplicity and clearness. The organization of this paper is as follows. First, we introduce available data and how it is organized as well as the basics for the Apriori algorithm. Next, we define our baskets as sets of items and the parameters that will be used to run the Apriori algorithm. Furthermore we discuss the algorithm used for association of the users' videos with the association rules and providing predicted videos. At the end, results from the simulation are provided. Finally, we pose some of our conclusions.

2 Association Rule Mining

For applying association rule mining, three sets of data from the VOD system are needed: all accounts that have been active in the VOD system, the data for all rentals in the period considered and all videos available in the system. The data is part of a relational database.

We define:

$$I=\{i_1, i_2, \ldots, i_n\}$$

as a set of items, in our case a set of active videos.

$$D=\{T_1, T_2, \ldots, T_m\}$$

is a set of transactions. Every transaction is a set of items (videos) that one user has rented in the considered period of time.

Every transaction $T_j, j=1,..,m$ is a subset of all items I, $T_j \subseteq I$.

A set of *k* items is called *k-itemset*.

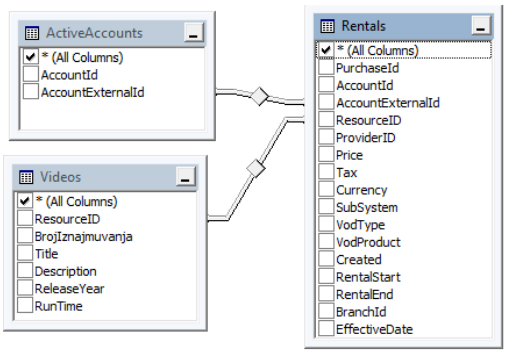

Fig.1. Data relation

Let X and Y are sets of specific items. An implication in form $X=>Y$, where $X \subset I$, $Y \subset I$ *and* $X \cap Y = \emptyset$, is called association rule. X is called antecedent, and Y is called consequent.

We define:

$|X|$ - number of items in X
$|Y|$ - number of items in Y
$|D|$ - number of items in D

To find the significant association rules we use the confidence and the support for every association rule. We say that the item-set X in the transactional set D has support *s,* if *s%* of the transactions in D include X.

$$s(X) = \frac{|X|}{|D|}$$

We say that the rule X=>Y in the transactional set *D,* has support *s,* if *s%* of the transactions in D include all the items in X and in Y $(X \cup Y)$.

$$s(X \Rightarrow Y) = \frac{|X| + |Y|}{|D|}$$

We say that the rule X=>Y in the transactional set *D,* has confidence *c,* if *c%* of the transactions in D that include the set of articles X also include the set of articles Y.

$$c(X \Rightarrow Y) = \frac{s(X \Rightarrow Y)}{s(X)}$$

We find our association rules from the set of transactions D by finding the rules $X=>Y$ whose support is larger than the previously selected minimum, called *minsupp* and whose confidence is larger than a previously selected minimum called *minconf*. A set of items is called large itemset if its support is larger than *minsupp*.

As an example, we need to find association rules in form:

$video_{1x}, ..., video_{nx} => video_{1y}, ..., video_{ny}$

We will read this association rule: The users that have rented videos: $video_{1x}, ..., video_{nx}$ have also rented videos: $video_{1y}, ..., video_{ny}$. If the support of this rule is 0.3, it means that 30% of all transactions include the videos: $video_{1x}, ..., video_{nx}$ and videos $video_{1y}, ..., video_{ny}$. The confidence of this rule is 0.8, if 80% the transaction that include $video_{1x}, ..., video_{nx}$ also include $video_{1y}, ..., video_{ny}$.

To find all available association rules, we've used the Apriori algorithm [1].

```
L1 = Large 1-itemsets
for (k = 2 ; Lk ≠Ø; k + +)
{
Ck = Apriori_Gen(Lk-1)
forall transactions t ∈ D
        {
Ct= subset(Ck , t)
forall candidates c ∈ Ct
                {
c.count + +
                }
              Lk = {c ∈ Ck|c. count ≥ minsupp ∗ |D|}
        }
}
return ∪k Lk
```

Fig. 2. Pseudo code of Apriori algorithm.

```
insert intoCk
selectp.item1, p.item2, …, p.itemk- 1, q.itemk- 1
from Lk- 1 p, Lk- 1q
wherep.item1 = q.item1,…, p.itemk-2 = q.itemk- 2, p.itemk-1 <
q.itemk- 1

forallitemsets c ∈Ckdo
    forall(k – 1)-subsets s of c do
          if (s ∉Lk- 1) then
                delete c from Ck;
```

Fig. 3. Pseudo code of Apriori_Gen algorithm.

By selecting different values for *minsupp* and *minconf*, different number of association rules is generated. If these values are smaller the number of association rules rises but the validity of these rules loses strength and vice versa. Choosing the appropriate values for *minsupp* and *minconf* is crucial for making a valid prediction.

The values can be adjusted by making statistical investigation when the model is applied for a longer period of time.

As discussed previously, we define all videos that one user has rented in a specific period of time as one transaction e.g. as one "basket". When a particular user logs into the VOD system, his/hers basket is compared to the baskets of the rest of the users by finding the association rules that are the best fit for this user. These association rules will give the videos that will be presented to the user. If the user data cannot be associated to any of the association rules, the overall most popular videos are presented to the user.

The data for these "transactions" is then formatted to be applied to the Apriori algorithm.

3 Predictions made by associations rules

To find the videos that will be best suited for a particular user, his/her history of rentals has to be compared to the association rules that were "mined" with the Apriori algorithm. An algorithm is proposed that should do the job. Because there might be users that have no history of rentals or their rented videos cannot be fitted in any of the association rules, a list of three most popular videos is prepared. If there is such a case, this user will be given a list of these 3 most popular videos.

The algorithm will present the user 3 videos in any case. All of these can be calculated from the association rules, or if the association rules cannot give 3 videos, the number will be populated from the most popular videos.

The popularity $p_i$of a video is calculated as a quotient from the total number of times the video was rented b_i and the total number of days the video is available in the system d_i.

$$p_i = \frac{b_i}{d_i}$$

We note N as the set of all videos that a user has rented in a period of time t. If N=∅, then the user will be presented with the 3 most popular videos: p_a, p_b and p_c. The set of the 3 videos is noted as P.

$p_a = max(\{b_i : i=1,..,n\})$
$p_b = max(\{b_i : i=1,..,n\} \setminus \{p_a\})$
$p_c = max(\{b_i : i=1,..,n\} \setminus \{p_a, p_b\})$

If $N \neq \emptyset$, then the user has his "own" basket of videos. This basket is compared with the association rules of type X=>Y. The algorithm starts from the rule with the highest confidence and continues until it reaches the rule with the lowest confidence, or while it finds 3 resulting videos. Starting from the first association rule the algorithm checks if $X \subseteq N$. If so, the videos from Y are the potential resulting videos. If some of the videos in Y are already members of the set N, then these videos are disregarded. We calculate:

$Y'=Y\backslash N$

If the number of videos in Y', noted as $|Y'|$ is larger or equal to 3, ($|Y'|\geq 3$), then the algorithm finishes and the first 3 videos of Y' are presented to the user as the predicted videos. If $|Y'|<3$, then the videos in Y' are presented as resulting videos and the algorithm continues to the next association rule, until it finds a total of 3 videos, or there are no more association rules. As it was mentioned previously, if there are no 3 resulting videos from the association rules, the rest of the videos are found from the list of the 3 most popular videos.

The pseudo code of the algorithm that compares the user's "basket" with the association rules is given below. The proposed algorithm can be divided in 2 parts – one that can be performed "offline" and one that can be performed "online". In real systems, when the user logs on to the VOD system, he/she should get the resulting predicted videos in a reasonable time. For this to happen, the association rules generation (call of the Apriori procedure in the pseudo code) as well as the calculation of the table of the most popular videos (FindMax function) can be done in the background.

These operations can be done when the VOD system has available resources. The "basket" of every user can also be prepared in this "offline" mode. The "online" part is performed when the user logs in the system. Then, only the search through the previously calculated association rules and the prepared most popular videos can be performed.

```
PROCEDURE comparison_algorithm

INPUT VALUES
minsupp, minconf;  // constants, numbers of type REAL
varv(1,n);  //vector of n videos available in the VOD
varb(1,n);  //vector of rentals in the VOD system
vard(1,n); // number of days a video is available

OUTPUT VALUES pa, pb, pc; //predicted videos

begin
 for i=1 to n
  p(i)=b(i)/d(i);
  i++;
next i;

 pa =FindMax(p(1,n));//find the video with maximum
rentals
 pb =FindMax (p(1,n)\{pa});//find the second max rented
video
 pc=FindMax (p(1,n)\{pa, pb});//find the third max
rented video

 Call Apriori(in:TRANSACTIONS, minsupp, minconf; out:
```

```
X_i=>Y_i; supp_i; conf_i; k);
  //generation of assoc.rules with Apriori algorithm with
support larger than
  //minsupp and confidence larger than minconf,
  //ordered descending by the confidence, that returns
  //the rules with their support and confidence and the
// total number of rules k.

  var N(0,r);  // r videos rented by user in interval t.
  var f=0;  //number of found videos. = 0 on the start
  var vid(1,5);  // finds max. 5 videos in a search

  if N is NOT NULL
  for i=1 to k // k - number of associat. rules generated
  if f<3 then
  ifX(i)⊆N then
  Y(i)=Y(i)\N; //eliminate videos that were rented before
        if|Y(i)|≥3then f=f+1;
  vid(f)= |Y(i)|(1);f=f+1;vid(f)= |Y(i)|(2); f=f+1;
  vid(f)= |Y(i)|(3);
  end if;
  if |Y(i)|=2 then f=f+1; vid(f)=|Y(i)|(1);f=f+1;
vid(f)=|Y(i)|(2);end if;
  if |Y(i)|=1then f=f+1; vid(f)=|Y(i)|(1);end if;
  end if;
  end if;
  i++;
  next i;
  if f>2 then vid=f(1); pb= vid(2); pc= vid(3); end if;
  if f=2 then pa= vid(1); pb= vid(2); end if;
  if f=1 then pa= vid(1);
  end if;
   end if;
return pa, pb, pc;       //return 3 videos
end.
```

Fig. 4 Pseudo code of the algorithm that compares the user data with the association rules and returns predicted videos.

The proposed model has a cycle of six steps given in figure 5. The pseudo code above is an implementation of the fifth step. The model can be calibrated by performing a statistical analysis on the videos rented from the proposals, as well as the total videos rented with different values for *minsupp* and *minconf*. When satisfactory results are gained, they can be used for valid prediction making. In practice, the calibration process can be done on a regular basis.

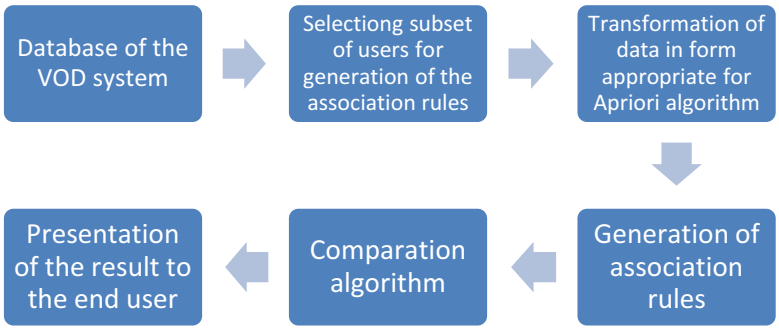

Fig. 5 Steps of the prediction process.

4 Simulation Results

The data used in the simulation is extracted from an existing and publicly available VOD system. The system is based on Microsoft Mediaroom 1.1 platform. The data is for the period of 3 months. It is a part of a relational MS SQL 2007 database.

We have developed an application in Microsoft SQL 2007 that implements the algorithm that compares the users' data with the association rules and returns predicted videos. The application consists of 2 stored procedures and 1 function. It gives results for the real data for the users and their activity in the VOD system.

The association rules are generated using WEKA. Different sets of association rules are produced by supplying different values for the *minsupp* and *minconf* values. The association rules are imported in new tables of the MS SQL database.

The simulation method is consisted of 6 steps:

Step1: Extracting the data for the users, videos and rentals in a relational database. A period of 3 months is used. 7576 users were active in this period. 161 videos were available for rental. Total of 26077 rentals were performed in this period of time.

Step 2: A subset of users is selected as a valid group whose rentals will be used for the generation of the association rules. All users that have rented 10 or more videos in the selected period of 3 months were selected in this group. There are 502 such accounts.

Step3: The data from the rentals is prepared in a format suitable for performing the Apriori Algorithm.

Step 4: Association rules are generated using the Apriori algorithm.

Step 5: Running the algorithm that compares the users' sets of rented videos with the association rules.

Step 6: Prediction of 3 videos for every one of the 7576 users is done and written in a results table.

	Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5
Minconf	0,4	0,4	0,3	0,7	0,5
Minsupp	0,1	0,08	0,08	0,1	0,2
No. of Assoc.rules (AR)	156	333	487	18	5
No. of users for which results cannot be found from the Association rules	2289	1488	1487	6922	5809
% of users for which results cannot be found from the Association rules	30,17 %	19,64 %	19,63 %	91,37%	76,68%
No. of vids found with Assoc. rules	13802	16564	17284	753	1767
% of videos found with the Associat. rules	60,73 %	72,88 %	76,05 %	3,31%	7,77%
No. of users for which 1 result is found from the Assoc. rules	829	619	397	559	1767
% of users for which 1 result is found from the Assoc. rules	10,94 %	8,17%	5,24%	7,38%	23,32%
No. of users for which 2 results are found from the Assoc. rules	410	462	189	91	0
% of users for which 2 results are found from the Associat. rules	5,41%	6,1%	2,49%	1,2%	0%
No. of users for which 3 results are found from the Assoc. rules	4051	5007	5503	4	0
% of users for which 3 results are found from ARs	53,47	66,09 %	72,64 %	0,05%	0%

Fig. 6 Table of results of the significant iterations.

Over 100 iterations with different values for *minconf* and *minsupp* were conducted. Five iterations were chosen for this paper as the most relevant, to show the prediction percentage based on the association rules. The results from the 5 iterations are given in the table below. By analyzing the iterations' results, it is obvious that the values of *minsupp* that is around 0,1 and *minconf* - around 0.4 will give the most logical results. If these values are higher, the number of predicted videos is very low. On the other hand, by lowering these values the number of association rules and predicted videos is rising, but the strength of the prediction is dropping.

In any real case scenario, these values can be calibrated and the most appropriate values can be found by analyzing the impact that the offered videos will have on the amount of videos rented.

5 Conclusion

In this paper we have proposed a method for recommending videos to users that log on in a VOD system. The model is used for the prediction of videos that will be most suited for the logged-in user. The data used was extracted from a real existing and publicly available VOD system (T-home's MAX TV), for a period of 3 months. The basic data mining technology is the association rule generation using the Apriori algorithm. The Apriori algorithm for the creation of the association rules was implemented in WEKA. The system should give the best prediction for the videos that a user can rent according to the previous rentals of that particular user, compared to the rentals of all users in the system.

The model can be calibrated on the fly, by choosing different values for the minimal support and confidence of the association rules. The process can be divided in two parts, one that can be prepared in advance and a second that will do the calculation when the user logs in. This makes it plausible in a real case scenario due

to the fact that the time latency that will be caused with the calculation step is reasonably low. The simulation is realized in WEKA and in MS SQL 2007. The results show that the system will give predictions for a significant number of users.

6 References

1. RakeshAgrawal, RamakrishnanSrikant, „Fast Algorithms for Mining Association Rules", *IBM Almaden Research Center,* 650 Harry Road, San Jose, CA 95120, 1999
2. Mehmet AydinUlas, „Market Basket Analysis for Data Mining", (PhD thesis) Bogazici University, 1999.
3. Sotiris Kotsiantis, Dimitris Kanellopoulos, „Association Rules Mining: A Recent Overview", *GESTS Intern. Trans. on Computer Science and Engineering*, Vol.32 (1), 2006, pp. 71-82, 2006.
4. R. Agrawal, H. Mannila, R. Srikant, H. Toivonen& A.I. Verkamo, „Fast discovery of association rules", *Advan. in Knowledge Discovery and Data Mining*, pp. 307 - 328, 1996.
5. YaseminBoztuğ, Lutz Hildebrandt, „A Market Basket Analysis Conducted with a Multivariate Logit Model", *Schmalenbach Business Review* (sbr), Vol. 60(4), pp. 400-422, 2005.
6. Sally Jo Cunningham, Eibe Frank, „Market Basket Analysis of Library Circulation Data", Proc. of *6th International Conference on Neural Information Processing*, vol. II, *Perth, Australia*, pp. 825-830, 1999.
7. LuísCavique, „A Scalable Algorithm for the Market Basket Analysis", *Journal of User Modeling and User-Adapted Interaction*, Vol. 19 Issue 1-2, February 2009.
8. E.García, C.Romero, S.Ventura, C.de Castro, „An architecture for making recommendations to courseware authors using association rule mining and collaborative filtering", *Jour. of User Modeling and User-Adapted Interaction*, Vol. 19 Issue 1-2, 2009.
9. Troy Raeder, Nitesh V. Chawla, „Market Basket Analysis with Networks", *Social Networks Analysis and Modeling Journal*, vol. 1, No. 2, pp. 97-113, 2010.
10. Ian H. Witten &Eibe Frank, „Data Mining Practical Machine Learning Tool and Techniques", second edition, Morgan Kaufmann Publishers, 2005.
11. Xiaoyuan Su and Taghi M. Khoshgoftaar, „A Survey of Collaborative Filtering Techniques", *Advances in Artificial Intelligence*, Vol. 2009 (2009), Article 421425, 2009.