RECOMMENDATION WITH ASSOCIATION RULES: A WEB MINING APPLICATION

Alípio Jorge[1,2]*, Mário Amado Alves*[1]*, Paulo Azevedo*[3]
[1] LIACC - Laboratório de Inteligência Artificial e Ciência de Computadores
Universidade do Porto - R. do Campo Alegre 823, 4150 Porto, Portugal
E-mail: amjorge@liacc.up.pt, http://www.niaad.liacc.up.pt
[2] Faculdade de Economia da Universidade do Porto
[3] Departamento de Informática, Universidade do Minho, Portugal

ABSTRACT

Data mining tools can bring many new possibilities for the analysis of web access log files. In this paper we follow one case (the Infoline web site) and describe our study on how to build recommendation models in order to improve the usability of the site. Our recommendation models are sets of association rules. We measure the performance of the models with different metrics on different levels of detail of the dataset.

1 INTRODUCTION

Each time a page in your web site is requested, your web server drops a line on the access log. The access log is basically a relation (or table, if you like) with a few columns that record what is regarded as important information. This data can then be mined for different purposes. In this paper we describe one application of data mining to the web site Infoline of INE, the Portuguese National Institute of Statistics (www.ine.pt).

The business problem we address here is how to improve the usability of the web site. Our approach is to build recommendation models that can produce recommendations to each user, on the fly, as she traverses the site, according to the pages the user visits in a given session.

2 BUSINESS UNDERSTANDING

Infoline is the web site of INE. It is one of the distributions channels of statistical data to the Portuguese citizens. You can enter the site, look for the data category that you want, find the data that you want and download it. In some cases, the download must be payed. In that case you have to be a registered user, login any time before you get the data (it can be done just before download), and get the data. Infoline has more than 9000 registered users, but they are not all necessarily active currently.

In a continuous effort to provide better service, INE is studying the usability of the web site [5]. Exploring the web logs using data mining is one of the threads of this action. The web log data is complemented with data about the registered users.

2.1 Business problems

At the higher level, INE wanted to obtain results on the following problems:

1. Improve the **usability** of the site, in the sense that users could more easily find and retrieve the information they were looking for. More easily means: with less clicks, in shorter time.
2. Knowledge about the users. Who is visiting the site? Are there distinct groups of users?

At a lower level, by refinement of these problems, we obtain the following:

1.1. How to indicate to our users interesting links to follow, according to their personal interests.

1.2. How to identify pitfalls in the site, i.e., paths that lead commonly to failure from the user.

2.1. Which are the groups of users that we have?

2.2 Data mining problems

In this paper we address the problem of how to indicate users links that might be relevant to them. The corresponding data mining problem is:

- *"Based on the sites web log, build a recommendation model that, given a set of visited pages, indicates a list of interesting links to the user"*

The idea is to build the recommendation model by first generating association rules from the web data. Then the pages visited by a given user are matched with the antecedent of the rules. The consequents of the matching rules with the highest confidence become the recommendations.

3 DATA PREPARATION

The data that we need to build the association rules for one recommendation model is a set D of transactions or baskets, where each basket is of the form $B = \{< Id, Item >\}$, where Id is the user identification, and the Item is a retrieved document or a document category.

For this approach we considered three category levels: the *Theme* (tema) level with 9 items, the *Subtheme* (subtema) level with 27 items, and the *Topic* (Tópico) level with 173 items. All these numbers correspond to the items actually occurring in the web log for the period we considered.

In the following we describe the process of data understanding, data storing in a relational database and data preparation.

3.1 Data preparation

The main source data are the web access logs produced—independently—by INE's two HTTP servers (located in Lisbon and Oporto). Complementary source data are a database of registered users and a mirror of the web site on disk. All the data in compressed form occupy 5 CD-ROMs (c. 3 Gigabytes) covering the period from 1999 to 2001.
The logs are packaged periodically, with varying periodicity (from 1 day to 1 month) and structure. These conditions imposed a substantive effort in understanding, collecting and pre-processing the data. The standard set of field definitions used to harmonize the data is shown in Table 1. A standard database system (MySQL [6]) is used to collect and prepare the data. The varying source fields were mapped onto the standard set, and corresponding transformation procedures were developed to populate the database. Accesses to images (GIFs and JPEGs) were not loaded into the database.

Name	*Type*
Id	int(11)
Server_Id	char(1)
Date	varchar(19)
IP	text
User_Id	text
Method	text
URI	text
Status	decimal(3,0)
Request_Volume	decimal(10,0)
Response_Volume	decimal(10,0)
Processing_Time	decimal(10,0)
Referer_URI	text

Table 1: Standard web access fields

Next, sessions were identified. A session is a sequence (in time) of accesses with the same Session_Owner, which is either the IP or the User_Id. Not all accesses have User_Id. To identify and represent sessions, five additional fields were used (Table 2). Unix_Time is the time in seconds elapsed since 1970-01-01 00:00:00. Sessions were identified by traversing all accesses ordered by the composite key (Session_Owner, Unix_Time): each new Session_Owner or a Unix_Time more than 30 minutes greater than the previous starts a new session.

Name	*Type*
Unix_Time	decimal(10,0)
IP_Session_Id	int(11)
Order_Number_In_IP_Session	int(11)
User_Session_Id	int(11)
Order_Number_In_User_Session	int(11)

Table 2: Additional access fields, for sessions.

Each page, either static or dynamic, has associated categories e.g. Tema (theme), Subtema (subtheme), Tópico (topic), which are organizing concepts of the site—and are the attributes for data mining in the currently reported study. In order to enrich the data with these categories, each access entry is mapped onto the corresponding categories. Such a mapping is mainly based on the character string pattern of the URI. A document prepared by INE provides most of this knowledge, in semi-formal style. Currently, approximately one third of the visited pages have as associated category. The unmatched URIs are assumed to be irrelevant to the current study.
Sessions are also enriched, with information derived from accesses the forementioned mapping (Table 3).

Name	*Type*
Owner	enum('ip','user')
Id	int(11)
Date	char(19)
Number_Of_Accesses	int(11)
Duration	bigint(13)
Volume	int(11)
Number_Of_Visualizations	int(11)

Table 3 Enriched sessions

4 MODELING

4.1 Association Rules

An association rule $A \rightarrow B$ represents a relationship between the sets of items A and B. Each item I is an atom representing a particular object. The relation is characterized by two measures: support and confidence of the rule. The support of a rule R within a dataset D, where D itself is a collection of sets of items (or itemsets), is the number of transactions in D that contain all the elements in $A \cup B$. The confidence of the rule is the proportion of transactions that contain $A \cup B$ with respect to the transactions with A. The most common algorithm for discovering AR from a dataset D is APRIORI [1].

4.2 Recommendation models with association rules

In the context of this paper, a recommendation model M outputs a set of items as recommendations R, given a set of observable items O. In our case, the model M is a set of association rules with support and confidence. To produce the recommendations, we build the set R as follows:

$$R = \{consequent(r_i) \mid r_i \in M \text{ and } antecedent(r_i) \subseteq O \text{ and } consequent(r_i) \notin O\}$$

If we want the N best recommendations (top N), we select from R the recommendations corresponding to the rules with highest confidence. This process for using association rules to generate top N recommendations is very similar to the one described in [7].

4.3 Evaluating the recommendation models

To evaluate the recommendation models produced, we used the All But One protocol described in [3]. In this protocol, the baskets in the dataset are split randomly into train and test (we chose an 80%/20% split). The training set is used to generate the recommendation model. From each basket in the test set we randomly delete one pair <id,item>. The set of deleted pairs is called the hidden set (*Hidden*). The set of baskets with the remaining pairs is called the observable set (*Observable*).
One model is evaluated by comparing the set of recommendations it makes (*Rec*), given the observable set,

against the items (we call a pair an item for the sake of simplicity) in the hidden set. The set of recommendations $\{r_1, r_2, ..., r_N\}$ for a given user ID = *id* is represented as $\{<id,r,>, <id,r_2,>, ..., <id,r_N>\}$. *Rec* is the union set of all the sets of recommendations over all the users. The number *N* of recommendations produced for each test basket can vary. Each recommendation model is used with different values of *N* and for each case we measure Recall, Precision and the F1 metric as defined below [7], [9].

$$Recall = \frac{|Hidden \cap Rec|}{|Hidden|}$$

This is a global measure for the whole set of users in the test. Recall corresponds to the proportion of correct answers and is an estimate of the probability of having at least one relevant recommendation. It tends to increase with *N*.

$$Precision = \frac{|Hidden \cap Rec|}{|Rec|}$$

Precision is also an average for all the test users. It gives us the quality of each individual recommendation. As *N* increases, the quality of each recommendation decreases.

$$F1 = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

F1 has been suggested as a measure that combines Recall and Precision with an equal weight. It ranges from 0 to 1 and higher values indicate better recommendations. It is useful as a summary of the other two measures and can be used to find the best combination (according to its own criterion) of Recall and Precision.

Contrarily to [7], we calculate the F1 measure from the global values of Recall and Precision, instead of calculating F1 for each user and then averaging.

5 EXPERIMENTS

5.1 Experimental setup

From the web log data we built three data sets, one for each level of detail of the web page category (theme, subtheme, topic).

A recommendation model is a set of association rules produced by Caren [2], a java implementation of Apriori [1]. The reference parameters were 0.02 for minimum support and 0.1 for minimum confidence. We explicitly indicate whenever different values are employed.

5.2 Experimental results

For the Topic level of detail, recall is around 16% when only one recommendation is made (Table 4). For the other two levels of detail, recall has the values of 21.6% and 27.4%. If we compare these recall values with the estimated results of a random guess, we see that in the case of the level Theme, we would get recall rates about twice as high (see Figure 1). This means that it is worthwhile making recommendations even at the Theme level (with only 9 themes), and this holds true independently of *N*. For the Subtheme and Topic levels, the model recall deviates considerably from the random guess recall (5.8 and 26 times , respectively). We also note that the most frequent items, for each level of detail, have the probabilities of 0.21 (theme), 0.13 (subtheme) and 0.05 (topic).

	Theme (9 items)				Subtheme (27 items)				Topic (173 items)			
N	*Recall*	*Prec.*	*F1*	*Rnd*	*Recall*	*Prec.*	*F1*	*Rnd*	*Recall*	*Prec.*	*F1*	*Rnd*
1	0.274	0.274	0.274	0.111	0.216	0.216	0.216	0.037	0.157	0.157	0.157	0.006
2	0.462	0.231	0.308	0.222	0.254	0.127	0.169	0.074	0.197	0.098	0.131	0.012
3	0.557	0.186	0.279	0.333	0.358	0.119	0.179	0.111	0.232	0.078	0.116	0.017
5	0.774	0.157	0.261	0.556	0.455	0.091	0.152	0.185	0.311	0.062	0.104	0.029
10					0.660	0.066	0.121	0.370	0.417	0.046	0.082	0.058
20									0.504	0.038	0.071	0.116

Table 4: Results for each level of detail and different number of recommendations (*N*).

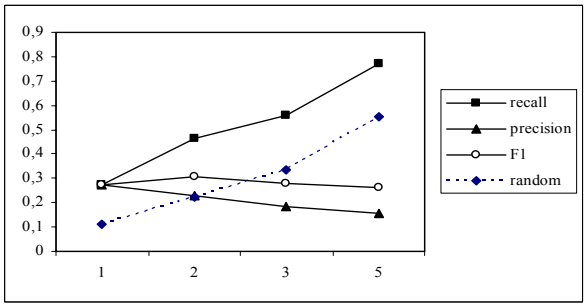

Figure 1: Results for the level of detail Theme.

As we can observe in Figure 2 and Figure 3, although increasing, recall values become relatively less interesting, with N. The Topic level, having more items (173) and larger baskets (5.5 items per basket, against 3.1 (Subtheme) and 2.1 (Theme)), is the more adequate of the three for automatic recommendation. More items give more fine grained rules, with higher confidences. Larger baskets have the same effect.

In the case of precision, it drops smoothly as the level of detail (number of items) increases. For the Topic level, when 10 recommendations are given (N=10), each one of them has a 4.6% chance of being relevant.

In Figure 4 we see how variations in the minimal support, when generating the association rules, affect Recall for the

Topic level. We observe that the best results are obtained with minsup= 0.02.

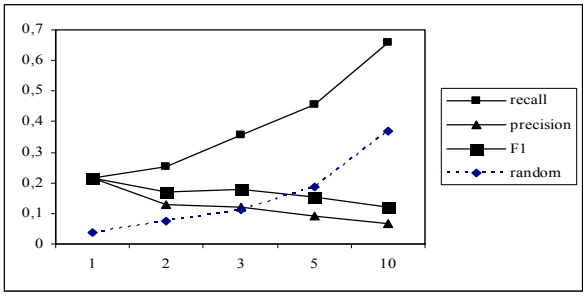

Figure 2: Results for Subtheme

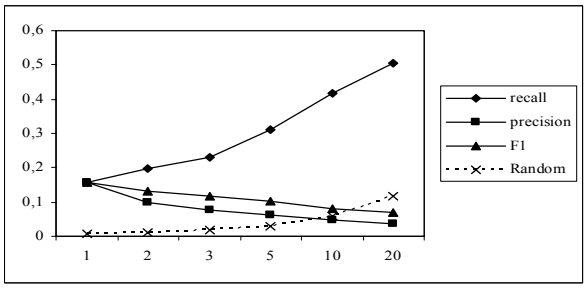

Figure 3: Results for Topic.

The F1 measure indicates that the best combination of Recall and Precision tends to occur for *N*=1.

6 RELATED WORK

The most popular technique used to produce recommendation models is Collaborative Filtering (CF) [3],[7]. The term collaborative filtering has allegedly been coined [7] by David Goldberg, David Nichols, Brian M. Oki and Douglas Terry in 1992 for the first recommender system Tapestry [4] for electronic mail filtering. The term intends to distinguish content-based filtering, where e-mails are selected depending on the occurrence of some string, and a filtering process based on the preferences of other users that have similar selection patterns, the so-called Collaborative Filtering.

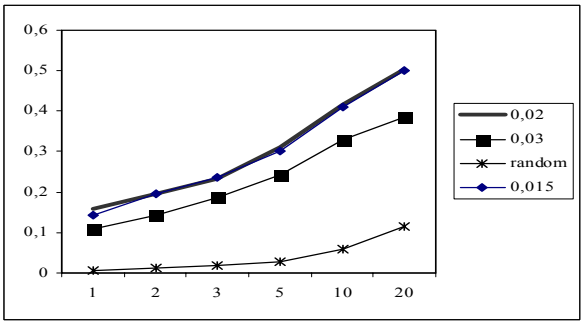

Figure 4: Recall for different *minsup* (Topic level).

Collaborative Filtering relies on the collection of data resulting from the activity of a large set of users. This data may contain votes/ratings from the users on a set of items (like movies or books) or simply binary information like bought/didn't buy. Collaborative filtering is commonly reduced to distance-based recommendation systems, working in a way similar to a nearest neighbor approach [8]. In [3] the term CF has a more general meaning, and these authors make a distinction between *memory-based* (akin to lazy classification) and *model-based* (akin to eager classification).

7 CONCLUSIONS

Although association rules are not the most common recommendation system, they have been used in the past and have been adopted for the work described in this paper. The performance of the AR-based recommendation models on the datasets resulting from the Infoline application is satisfactory, in the sense that they deviate considerably from the random recommendation. Recall values indicate that a top 10 recommendation may work more than 40% of the times. However, more experiments are needed in order to get to more tangible conclusions. Comparison with other collaborative filtering systems has not been done.

In terms of the application as a whole, data preparation has been a very laborious task especially because of the dynamics of the site structure and the difficulty in obtaining definite answers to our business understanding and data understanding questions.

Acknowledgements: This work is supported by the European Union grant IST-1999-11.495 Sol-Eu-Net and the POSI/2001/Class Project sponsored by Fundação Ciência e Tecnologia, FEDER e Programa de Financiamento Plurianual de Unidades de I & D

REFERENCES

1. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A. I., Fast Discovery of Association Rules. *Advances in Knowledge Discovery and Data Mining*: 307-328. 1996.
2. Azevedo, P. J., home page, http://www.di.uminho.pt/~pja.
3. Breese, J.S., Heckerman, D., and Kadie, C. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings *of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence*, pages 43--52, July 1998. http://citeseer.nj.nec.com/article/breese98empirical.html
4. Goldberg, D., Nichols, D., Oki, B.M., Terry, D., Using collaborative filtering to weave an information tapestry, Communications of the ACM, v.35, n.12, p.61-70, Dec. 1992.
5. Instituto Nacional de Estatística, Nova versão para o infoline 2002, (in Portuguese), *INEWS*, nº 5, March 2002.
6. MySQL, The World's Most Popular Open Source Database, http://www.mysql.com
7. Resnick, P., Varian, H. R., Recommender Systems, Communications of the ACM, Vol. 40, No. 3, March 1997.
8. Sarwar, B., Karypis, G., Konstan, J., and Riedl, J.. Analysis of recommendation algorithms for e-commerce. In *Proceedings of ACM E-Commerce*, 2000. http://citeseer.nj.nec.com/article/sarwar00analysis.html.
9. Yang, Y. and Liu, X. A re-examination of text categorization methods. In *Proceedings of ACM* SIGIR-99 *conference*, 1999. http://citeseer.nj.nec.com/yang99reexamination.html