Temporal Rating Habits: A Valuable Tool for Rater Differentiation

Pedro G. Campos[1,2]
pedro.campos@uam.es

Fernando Díez[1]
fernando.diez@uam.es

Alejandro Bellogín[1]
alejandro.bellogin@uam.es

[1]Universidad Autónoma de Madrid
Francisco Tomás y Valiente 11
28049, Madrid, Spain

[2]Universidad del Bío-Bío
Av. Collao 1202
4081112, Concepción, Chile

ABSTRACT

In this paper, we describe the experiments conducted by the *Information Retrieval Group* at the Universidad Autónoma de Madrid (Spain) to tackle the Rater Prediction task (track 2) of the CAMRa 2011 Challenge. The experiments performed includes time-frequency probabilistic strategies, a simple kNN and a matrix factorization approaches. Results show that probabilistic classifiers based on temporal behavior of users have better performance than traditional recommendation based strategies, thus reflecting that temporal information is a valuable source for the identification or differentiation of users.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information Filtering, Retrieval Models, Selection Process*; I.5.1 [**Pattern recognition**]: Models

General Terms

Algorithms, Performance

Keywords

Context-Aware Recommender Systems, Movie Recommendation, Probability Models

1. INTRODUCTION

Contextual information can help improving personalization-related tasks [1]. The Challenge on Context-aware Movie Recommendation 2011 (CAMRa2011) provides an interesting opportunity to test recommendation approaches on real data. We focus on the *Rating Prediction Track*, which consists in, knowing which raters pertain to a *household* and given a set of movie ratings of households, determine which member of the household made particular “unidentified” ratings. In this case, there are two dimensions of contextual information. On the one hand, household information, which may allow to take advantage of knowing the existence of a relationship among some users (although which relation is it remains unknown), and on the other hand temporal data, as each rating has an associated time-stamp, which allows to track users' concept drift. However, other interesting information which have been used in recommendation strategies, e.g. movies features as *title* or *genre*, user demographics, other social relationships, etc., are unavailable, making it hard to define relations between the type of film in question and the users to be allocated.

Considering the above issues mentioned, we conducted a series of experiments with different models, in order to better predict whom each “unidentified” rating belongs to, which we describe in this work. The remainder of this paper is structured as follows: Section 2 describes the main characteristics of the available data for the competition. Section 3 details the models used for making the predictions. Section 4 presents the results obtained, in terms of the required challenge metrics. We finalize with some concluding remarks and devised additional approaches to experiment in section 5.

2. DATASET ANALYSIS

2.1 General Description

CAMRa 2011's MoviePilot Dataset consist of a training set of 4.536.891 time-stamped ratings from 171.670 users on 23.974 items on a timespan from July 11, 2009 up to July 12, 2010, and two test sets (one for each competition track): track 1 containing 4.482 ratings from 594 users on 811 items on a timespan from July 15, 2009 up to July 10, 2010 and track 2 containing 5.450 timestamped ratings from 592 users on 1.706 items on a timespan from July 13, 2009 up to July 11, 2010. As we are focused on track 2, from now on we analyze only track 2 related data.

Figure 1 shows the rating, community and catalog growth of training data (upper side) and testing data for the track 2 (lower side) through time. It may be seen that data growth is proportional on both data splits. Table 1 shows the size distribution of households in the dataset. 2-sized households represent the 93,8% of all households, whilst 3-sized and 4-sized households represent the 4,8% and 1,4% respectively.

2.2 Frequency Based Analysis

Taken into account that we do not know whether the households relationships correspond to friends, siblings, cou-

CAMRa 2011 October 27, 2011, Chicago, IL USA

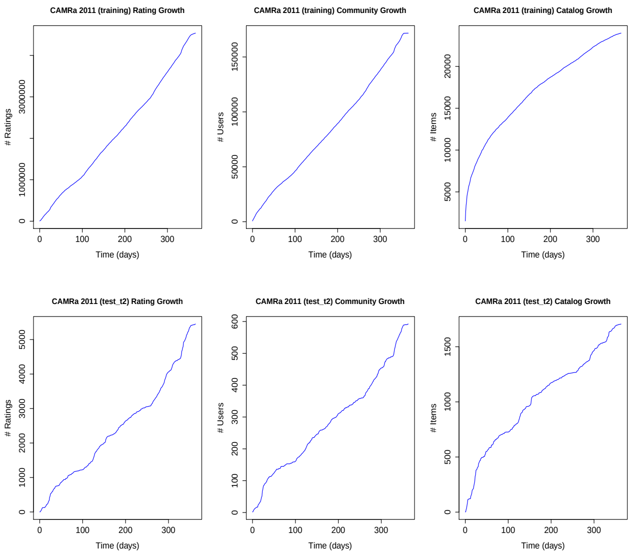

Figure 1: Training data growth through time

Table 1: Households' size frequencies

	All	Size-2	Size-3	Size-4
Frequency	290	272	14	4

ples, etc., and that no other information is provided, we focused our analysis on temporal trends which may help us on completing the task at hand. We performed a descriptive study of the given characteristics on training data and we observed a phenomenon repeated in several of the users belonging to different households. In Figure 2, it is shown the rating hour histogram of a couple of users in the first household. Here, we can observe there is a clear disparity between the hours employed by each of the household members in rating movies. The user u40426 has a probability near 1 (0.93) to rate movies in the period from 18:00 to 19:00. On the contrary, user u311738 rates movies from 20:00 on, that is, mostly by night. Circumstances similar to the one observed are repeated all along the data set.

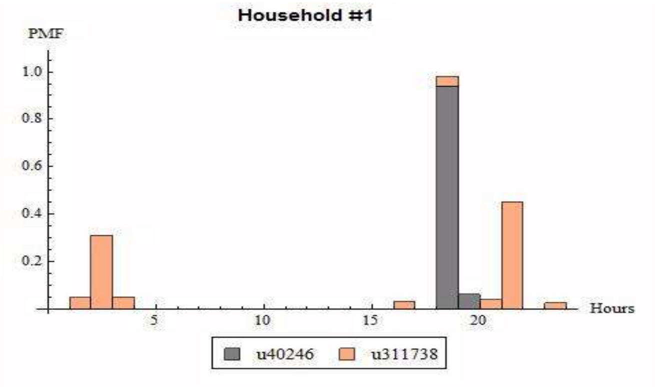

Figure 2: PMF of the user rating hours

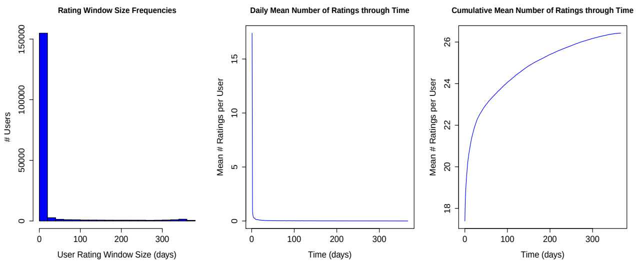

Figure 3: User's rating frequencies through time

When analyzing the date of rating from each user, it is also possible to detect some interesting facts. Figure 3 shows how many ratings are made by users through time. The left frame shows that the mean user rating window size (i.e. timespan at which users makes ratings) is very small (just a few days). The center and right frames also shows that vast majority of ratings are incorporated during the first days of participation of a user. Considering that users start their participation on different days, this information can be helpful in our task. We also noted that there are differences on which day of the week each user rates movies.

The analysis of rating frequency alone also give some clues about user behaviors. Figure 4 shows an example of two probability mass functions (PMFs from now on) of rating values, corresponding to two couple of users (in different households). The one on the left emphasizes the fact that user u322924 (thick lined) rarely gives ratings higher than 90 points. On the contrary, user u880228 (dashed lined) usually gives ratings higher than 90 points. The example on the right has a stronger differentiation. The dashed user rates less than 10 points most of the time. On the contrary, the thick lined user tend to rate over 60 points.

All the above suggest us to take into account the following dimensions in order to identify raters:

- The **hour of the day** in which a user rates movies more frequently (H).
- The **day of the week** in which a user rates movies more frequently (W).
- The **date of rate** (D).
- The **number of ratings** given by users (R).

3. PREDICTIVE MODELS

This section describes the models used for the challenge. We begin with the probabilistic models which gave the best performance. Then, we describe other more traditional recommendation models which were used to compare our results.

3.1 Probability Based Models

The findings observed from the dataset analysis motivated us to use probability based models to infer which users were the ones who evaluate each movie as required by the challenge. We used a discriminant function based on the PMFs obtained, giving more probability to users depending on the probabilities of the previously mentioned dimensions of information. Below we describe the two approaches used.

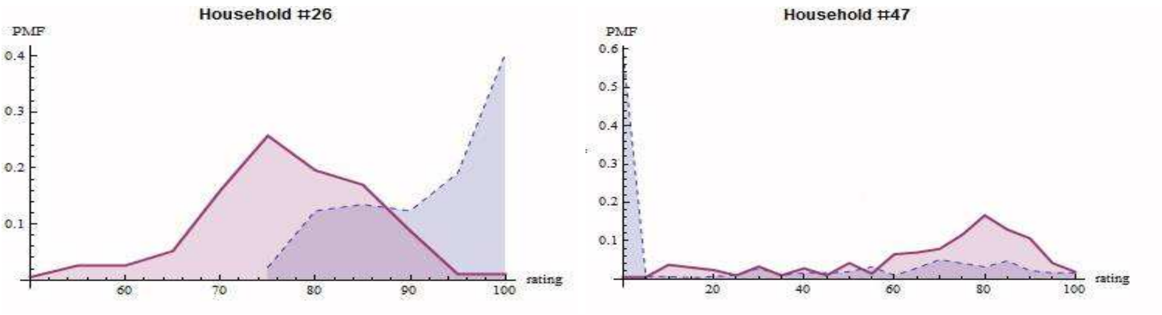

Figure 4: Histogram of the users rating values in households #26 (left) and #47 (right)

3.1.1 A-priori Model

Let us consider a set of objects $O = \{o_1, o_2, ..., o_m\}$ and a set of classes $\Omega = \{\omega_1, \omega_2, ..., \omega_c\}$, such that each object o_i is member of one, and only one, class ω_j. In addition, consider that these objects are described by means of the value of some numerical quantity feature, called X. Now, the question we want to answer herein is whether it is possible to determine which class an object o_i belongs to or not, once the value x_i of its feature X is already known? If we assume that we know the *a priori* probabilities of the respective classes, a simple classification rule can be:

$$\text{Assign } o_i \text{ to } \omega_j = \arg\max_{\omega_j \in \Omega} P(X = x_i|\omega_j) \tag{1}$$

Bringing this model to our case, let U_h be the set of users from household h, and let $\check{R}_h = \{\check{r}_1, \check{r}_2, ..., \check{r}_m\}$ be the set of unidentified ratings from h, that is, ratings that are known to be given by a user u_j from U_h, but not knowing which particular user u_j gave it. We define, based on the a-priori PMFs of feature X, $P(X|u_j)$ (where X can be any of the information dimensions described in section 2.2):

$$score(\check{r}_i, u_j) = P(X = x_i|u_j) \tag{2}$$

Once the scores given to each pair $(\check{r}_i, u_j)$ are determined, the *a-priori* based discriminant function assigns the rating $\check{r}_i$ to the user that reached the highest probability. That is:

$$\text{Assign } \check{r}_i \text{ to } u_j = \arg\max_{u_j \in U_h} P(X = x_i|u_j) \tag{3}$$

3.1.2 Bayesian Model

Now, considering we know the PMFs of the feature X and each class, i.e., $P(X)$ and $P(\omega_j)$, and applying the Bayes' theorem, we compute the corresponding probabilities of each class provided the feature X:

$$P(\omega_j|X = x_i) = \frac{P(X = x_i|\omega_j)P(\omega_j)}{P(X = x_i)} \tag{4}$$

Then, the previous classification rule improves in:

$$\text{Assign } o_i \text{ to } \omega_j = \arg\max_{\omega_j \in \Omega} P(\omega_j|X = x_i) \tag{5}$$

Therefore, in our case we compute again the previously defined scores as:

$$score(\check{r}_i, u_j) = P(u_j|X = x_i) \tag{6}$$

Then, we apply the same decision rule as defined in the previous model (3). These models can be easily extended to consider a set of features $\mathcal{X} = \{X_1, X_2, ..., X_n\}$ describing each object o_i by computing the combined probability $P(X_1 = x_{1_i}, X_2 = x_{2_i}, ..., X_n = x_{n_i}|\omega_j)$. Using the conditional independence (a.k.a. naïve) assumption that each feature X_k is conditionally independent of every other feature X_l for $k \neq l$, we can compute it by $\prod_{k=1}^{n} P(X_k = x_{k_i}|\omega_j)$[5].

3.2 Recommendation based Models

Another discriminant can be build by computing a prediction of the rating $\widehat{r}_{i,j}$ that a user u_j would give to a movie m_i. Thus, if we compute rating predictions for m_i for each user in U_h, and knowing the actual rating value of $\check{r}_i$ (as provided by the challenge), we can assign the rating to the user with the lowest difference:

$$\text{Assign } \check{r}_i \text{ to } u_j = \arg\min_{u_j \in U_h} J = |(rating\_value(\check{r}_i) - \widehat{r}_{u,i}| \tag{7}$$

In order to compute rating predictions, we used two state-of-the-art recommendation methods, which are described below:

3.2.1 k-Nearest Neighbors

k-Nearest Neighbors (kNN) model [2] has been a widely used recommendation method due to its simplicity and good performance. It determines users most similar (a.k.a. nearest neighbors) to the target user, and considering ratings that they have given to the target item, it extrapolates the rating that the target user would give to the item, using the similarity value as a weighting factor:

$$\hat{r}_{u,i} = b + \sum_{u' \in N(u)} sim(u, u') \times r_{u',i} \tag{8}$$

Here b is a normalizing factor, usually computed as $b = 1/\sum_{u' \in N(u)} sim(u, u')$, $r_{u,i}$ is the rating value given by user u to item i, $N(u)$ is the set of k nearest neighbors of u computed by:

$$N_k(u) = \bigcup_{j=1}^{k} u'_j : u'_j = \arg\max_{u' \in \mathcal{U} - N_{j-1}(u), u \neq u'} sim(u, u') \tag{9}$$

with $N_0 = \emptyset$ and $sim(u, u_j)$ is the similarity between u and u_j, usually computed as the correlation among co-ratings, e.g. Pearson Correlation:

$$sim(u,v) = \frac{\sum_{i \in \mathcal{I}_{uv}} (r_{u,i} - \overline{r}_u)(r_{v,i} - \overline{r}_v)}{\sqrt{\sum_{i \in \mathcal{I}_{uv}} (r_{u,i} - \overline{r}_u)^2 \sum_{i \in \mathcal{I}_{uv}} (r_{v,i} - \overline{r}_v)^2}} \tag{10}$$

where $\mathcal{I}_{uv} = i \in \mathcal{I} : r_{u,i} \neq \emptyset \wedge r_{v,i} \neq \emptyset$. The above method is known as a *user based* collaborative filtering algorithm, as it is based on rating information of similar users. Similarly, this model can be computed on items similar to the target item, in which case it is called *item based* collaborative filtering.

3.2.2 Matrix Factorization

Matrix Factorization is an adaptation of the Singular Value Decomposition approach that is gaining increasing interest in the field of Recommender Systems due to its good performance [3]. In this technique, the known rating values, represented as a rating matrix R, are iteratively approximated by user and item factor matrices P and Q (f user and item factors) such that:

$$\hat{r}_{u,i} = \sum_{j=0}^{f} P_{u,j} \cdot Q_{j,i} = p_u^T q_i \tag{11}$$

One advantage of this approach is that P and Q values may by computed for all users and items using only the known values R, minimizing and estimation of the difference, e.g. the Frobenius Norm: $\min \|R - PQ\|^2$. Overfitting can be alleviated using regularization, i.e. penalizing the magnitude of the approximated vectors [3]. The common regularized formulation for collaborative filtering is inspired in minimizing the squared error on the set of ratings:

$$\min_{p*,q*} J = \sum_{u,i \in R} (r_{u,i} - p_u^T q_i)^2 + \lambda(||p_u||^2 + ||q_i||^2) \tag{12}$$

Different algorithms exist to compute this kind of factorization. A widely used implementation of stochastic gradient descent was published by Simon Funk[1] in the context of the Netflix Prize. In this implementation, for each known rating, the parameters are optimized by updating them in the opposite direction of the gradient of the optimization criterion, using a *learning rate* parameter γ which controls the amount of update [3, 4]:

$$\begin{array}{rcl} p'_u & \leftarrow & p_u - \gamma \cdot \frac{\partial J}{\partial p_u} \\ q'_i & \leftarrow & q_i - \gamma \cdot \frac{\partial J}{\partial q_i} \end{array} \tag{13}$$

4. RESULTS

4.1 Implementation details

Table 2 shows the parameter values used in the implementation of the recommendation based models described in section 3.2. Note that we used an item based kNN algorithm. In the case of the Probabilistic based models, we ran out several trials combining the different features previously defined. In the next section are shown the best results obtained with all the described algorithms.

Table 2: Parameter values

Model	Param.	Value
kNN	k	200 items
MF	f	10 factors
	λ	0,001
	γ	0,02

4.2 Results

Table 3 shows results obtained with the tested models (bold indicates best column value). It may be seen that the best performing algorithm is the A-priori model when using the combination of **hour of the day** and **date of rate** features (HD). It is also interesting to note that, in general, A-priori models have superior performance than Bayes models, independently of the features considered. A possible explanation for this is that the independence assumption is violated. Deeper analysis is required in order to verify if the independence assumption between features is acceptable or not.

All the results involving the H feature, considered alone or combined with other features, present a value up to 0.9 except for the case of Bayes (RH) within all the households (2nd column in Table 3). No other algorithm grows up to this value. This fact give us a strong evidence of the importance of this feature. Among the three time-aware features studied (H, D and W), H is the one with higher discriminant capabilities for the task required.

It is also remarkable the poor performance of the number of ratings feature (R). It gives the lower values for the metric considered, even when taking into account the results of the recommendation based models, considered as baselines.

Regarding the classical recommendation models, which are based on the extrapolation of rating values, both of them present poorer results than most of the probabilistic ones. The only probabilist models that they are able to rival are the ones based on the rating value feature. This seems to remark that differentiating users based only on rating values is hard, and other features (as the temporal ones) are better suited for this task.

Table 4 shows the best results using an additional set of metrics, based on precision such as P@5, P@10, and MAP, and AUC (area under the curve), computed on each user's recommendation list and averaged on all test users (not on a per-household basis). As it may be seen, results are consistent with classification accuracy rate outcome, regarding the best performing models.

5. CONCLUSIONS AND FUTURE WORK

This paper has described methods used for identifying users that made particular ratings. We focused the analysis on the study of PMFs of available features describing ratings, thus developing ad-hoc probability based models. Results obtained, compared with performance of recommendation based models adapted for the task, show that an adequate combination of features allows probability models to obtain an interesting classification accuracy rate (>90%). Also, it is notable the good performance of the feature **hour of the day** combined with **date of rating** or **day of the week**, showing that users have “temporal habits” when rating movies. It is thus expectable that the addition of time data awareness into recommendation based models improve their results. Furthermore, this finding could help on other

[1] `http://sifter.org/~simon/journal/20061211.html`

Table 3: Results on Rater Identification Task

	Classification Accuracy			
Model	All	Size-2	Size-3	Size-4
A-priori (R)	0,618	0,6336	0,3998	0,3255
A-priori (H)	0,9056	0,9074	0,8976	0,8065
A-priori (W)	0,8683	0,8706	0,8257	0,8615
A-priori (D)	0,8784	0,88	0,8269	0,9527
A-priori (RH)	0,9097	0,9115	0,9033	0,806
A-priori (RW)	0,8852	0,8877	0,8514	0,8383
A-priori (RD)	0,8975	0,8991	0,8526	0,9456
A-priori (HW)	0,9365	0,935	0,9586	0,9567
A-priori (HD)	**0,9392**	**0,9375**	**0,9604**	**0,9803**
A-priori (DW)	0,8825	0,8837	0,8419	0,9487
A-priori (HRDW)	0,9374	0,9358	0,9572	0,9773
Bayes (R)	0,6839	0,6979	0,5175	0,3145
Bayes (H)	0,9049	0,9084	0,9033	0,6688
Bayes (W)	0,8597	0,8654	0,7766	0,7673
Bayes (D)	0,8543	0,8627	0,7111	0,7915
Bayes (RH)	0,8858	0,8883	0,889	0,7067
Bayes (RW)	0,8726	0,8767	0,8276	0,7537
Bayes (RD)	0,8579	0,8652	0,7093	0,8837
Bayes (HW)	0,9211	0,9214	0,9315	0,8675
Bayes (HD)	0,914	0,9154	0,8908	0,8968
Bayes (DW)	0,8651	0,8707	0,7472	0,8994
Bayes (HRDW)	0,9191	0,9188	0,9192	0,9376
kNN	0,6467	0,658	0,4865	0,4399
MF	0,6412	0,6525	0,5016	0,3668

Table 4: Additional metrics for the task

Model	P@5	P@10	MAP	AUC
A-priori (HD)	0,9392	0,9375	0,9604	0,9803
Bayes (HW)	0,9211	0,9214	0,9315	0,8675
kNN	0,6287	0,4541	0,5509	0,8217
MF	0,6098	0,4468	0,5482	0,8279

interesting recommendation-related tasks, e.g. detecting the best hour of the day to send recommendations to users (via mobile devices, for example).

Regarding future work, we will test additional discriminants, based on clustering, SVMs, etc. Moreover, we think that the usage of classifiers specific for binary classes may improve performance on 2-sized households, whereas multiclass classifiers should be used on 3 and 4-sized households. Finally, a mixture of classifiers can be considered for further improvements on classification accuracy. We also want to study the independence assumption of the considered features using, for example, Fisher's independence analysis based on contingency tables.

6. ACKNOWLEDGMENTS

This research was partially supported by the Spanish Ministry of Science and Innovation (TIN2008-06566-C04-02) and by the Dirección General de Universidades e Investigación de la Comunidad de Madrid and Universidad Autónoma de Madrid (CCG10-UAM/TIC-5877). The authors acknowledge support from the Scientific Computing Institute at UAM (CCC at UAM). The first author acknowledges support from the Chilean Government through *Becas-Chile* scholarship program and from the Universidad del Bío-Bío through *Beca de Perfeccionamiento.*

7. REFERENCES

[1] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans.Inf.Syst.*, 23(1):103–145, 2005.

[2] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237, New York, NY, USA, 1999. ACM.

[3] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[4] G. Takács, I. Pilászy, B. Németh, and D. Tikk. Investigation of various matrix factorization methods for large recommender systems. In *Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition - NETFLIX '08*, pages 1–8, 2008.

[5] I. H. Witten, E. Frank, and M. V. Hall. *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann, 2nd edition. edition, 2005.