RED: a Rich Epinions Dataset for Recommender Systems

Simon Meyffret, Emmanuel Guillot, Lionel Médini, Frédérique Laforest

RED: a Rich Epinions Dataset for Recommender Systems

Simon Meyffret[1,2], Emmanuel Guillot[1,2], Lionel Médini[1,3], Frédérique Laforest[1,2]
[1]Université de Lyon, CNRS
[2]INSA-Lyon, LIRIS, UMR5205
[2]F-69621, France
[3]Université Lyon 1, LIRIS, UMR5205
[3]F-69622, France
{firstname.name, red}@liris.cnrs.fr

ABSTRACT

Recommender Systems require specific datasets to evaluate their approach. They do not require the same information: descriptions of users or items or users interactions may be necessary, which is not gathered in today datasets. In this paper, we provide a dataset containing reviews from users on items, trust values between users, items category, categories hierarchy and users expertise on categories. This dataset can be used to evaluate various Recommender Systems using Collaborative Filtering, Content-Based or Trust-Based.

Keywords

recommendation systems, evaluation, dataset

1. INTRODUCTION

Recommender Systems (RS) are now widely used. They aim at recommending items to users to ease items selection regarding user profile. The profile contains usually user's preference on items with possibly some additional information depending on the RS. A user preference on an item is generally a rate, *e.g.* a number between 1 and 5, but it can be a set of direct comparisons between items [JS08].

Main RS are Content-Based, Collaborative Filtering or Trust-Based. Content-Based RS focus on items similarity and therefore need items descriptions and hierarchy. Collaborative Filtering RS focus on users similarity computed from users profiles. Trust-Based RS focus on implicit or explicit trust relations between users. Whatever the chosen approach, one needs to evaluate it. Users evaluation campaigns based on questionnaires can be run; they often provide rich feedback but are very heavy to proceed. A more commun technique uses a dataset containing real information extracted from a recommendation application. Evaluation consists in comparing predicted ratings with real ones. However, the dataset must contain all data necessary for the RS, such as trust relations for Trust-Based RS or items description for Content-Based RS.

A dataset containing information for Content-Based and Trust-Based RS is required to define and evaluate hybrid approaches. To the best of our knowledge no such dataset is freely available. In this article we define and share a dataset that enhances the trust-oriented dataset proposed in [MB04]. It contains additional information on items categories, categories hierarchy and users expertise on those categories.

In the following, we first describe existing datasets, with their pros and cons. We present in section 3 the Epinions[1] website we have extracted. Section 4 describes the dataset structure and analyzes it. We also provide some additional elements we have calculated on the extracted dataset. We provide in section 5 tips on how to easily use this dataset for an evaluation purpose.

2. STATE OF THE ART

Many classical RS datasets [MAL+03, CKR+07, ZMKL05, GRGP01] contain ratings between user and items (movies, books, jokes...) and sometimes items description but no trust. Classical Collaborative Filtering RS can use them, but Trust-Based RS cannot.

Some trust datasets exist, coming from Advogato, Twitter, Delicious, Facebook... Those datasets contain some form of social relations (friendship, trust or intercommunication) but they do not contain explicit ratings on items.

Few datasets contain trust and ratings: [YZC+09] contains user ratings on music and social relations; [DGKS09] contains trust relations between users and films ratings. However no reference dataset has been publicly released. Some websites like LastFM provide an API to build such datasets, but in order to compare various approaches, we need to use exactly the same dataset.

[MB04] has released an anonymised Epinions dataset, with trust relations and item ratings, used to evaluate Trust-Based approaches [MA07, XKD09, MYLK08]. Unfortunately, there is no users profiles nor items descriptions. Then, no Content-Based approach can use this dataset.

3. EPINIONS WEBSITE EXTRACTION

3.1 Epinions website

The Epinions website contains reviews made by users on items. Items are any product or service. They have names and belong to one unique category. In a given category,

[1]www.epinions.com

items may show a common description structure. Categories are structured in a tree and may contain any number of items or subcategories.

Users build their web of trust within the community. A web of trust is a list of trusted or distrusted users. Anyone can trust or be trusted by anyone. Trusted users' reviews are promoted and distrusted users' reviews are less likely encountered. The web of trust may or may not be public, depending on the user settings.

A review contains a rating between 1 and 5 and a free text message. It may also contain some specific characteristics depending on the category (*e.g.* photo quality or shutter lag for cameras). Reviews can be commented and/or rated. A review rating is either "Not Helpful", "Somewhat Helpful", "Helpful", "Very Helpful", "Most Helpful" or "Off Topic". Express reviews are very short reviews that can only be tagged with "Show" or "Don't Show" whether they are valid or not.

Epinions defines four kinds of users[2]:

- *Category leads* ensure high-quality review coverage of key items in their category and ensure that new reviews in their category are rated by a category lead or an advisor (see below).
- *Top reviewers* write high-quality reviews in their category of expertise. Their reviews have received the highest ratings from the Epinions community.
- *Advisors* rate reviews in their category.
- *Regular users* can review items, rate reviews and trust or block other users.

Orthogonally, any user can be a popular author: they are determined by the number of total visits to their reviews. Popular reviewers in specific categories are based on the users' total number of visits in that category. These users hold a top X rank (top 10, top 100...).

3.2 Extraction

Regarding the Epinions website structure, two strategies could have been used to do the extraction: extract all items, then for each item extract the relative reviews and the associated users; extract all users, then for each user extract the relative reviews and the associated items.

For the first strategy, Epinions proposes an easy way to browse items through items categories and subcategories. However there is no standardization between categories and parsing categories is not an easy job. Moreover, each category cannot show more than fifteen hundred items. And finally, many items do not have review, they are not useful regarding our purpose. We could also search all items through the search field with a dictionary approach. But the result list is also limited to fifteen hundred items.

We have then implemented the second strategy: search all users through the "members search" facility with a dictionary based approach. The fifteen hundred users limitation applies also here, but we have managed to extract a subsequent number of users with this approach: 240 000 users. Then, for each identified user, we have parsed his/her profile, reviews and web of trust, adding new users if any. This brought a total of about 307 000 users. For each users review, we have parsed the associated item if new and its category.

[2]http://www99.epinions.com/help/faq/?show=faq_recognition

This approach ensures that items in the dataset have been reviewed at least once. However it does not ensure that each user has reviewed at least one item. We then cleaned the dataset by removing all unnecessary users, *i.e.* users with no trust relation nor review. Those users were found with the dictionary based approach and are certainly users who wanted to try Epinions or use a read only access.

This extraction took two plain days of crawling in June 2011 on an Intel Core 2 Duo notebook with 3 Go of RAM.

We have encountered several problems during the dataset extraction. First of all, the Epinions website html structure is very particular, using a lot of table tags and very few CSS classes. This made the use of XPath very difficult. In addition, there are many exceptions in the pages structures, some pieces of information were missing sometimes whereas some others appeared not often. Moreover some special characters in users names were problematic. Categories breadcrumbs are not always consistent and made the category extraction pretty chaotic: we had to correct it manually.

4. DATASET

4.1 Structure

As shown in figure 1, the dataset is a relational database with the following tables:

- User: name (pseudo and profile url), location, top rank (may be null) and profile visits count
- Item: name, category and profile url
- Category: name, parent category, description url, lineage (path in the category tree) and depth (in the category tree)
- Review: a review associates a user with an item, it contains the rating, between 1 and 5, the review rating (mean of all review ratings associated with this review) and the review date
- Expertise: users who are experts in a category appear here with the expertise (category lead, top reviewer, advisor) associated with the considered category
- Trust: web of trust, *i.e.* a trust value (either -1 or 1) from one user to another, only positive trust values appear in the dataset
- Similarity: we have computed the similarity between all users couples using the Pearson coefficient correlation [BHKO98]. Since this operation may be long and is used in classical collaborative filtering, we provide it in order to ease recommendation; those values do not belong to the Epinions website

4.2 Statistics

The dataset contains 131 228 users, 317 755 items and 1 127 673 reviews, that is a 0.003 % sparsity. 113 629 users have at least one rating. 47 522 users have at least one trust relation. 31 000 users have at least one similarity computed toward another user. 21 910 users have at least one review, one trust relation and one computed similarity. 4 287 users have neither reviews nor trust relation.

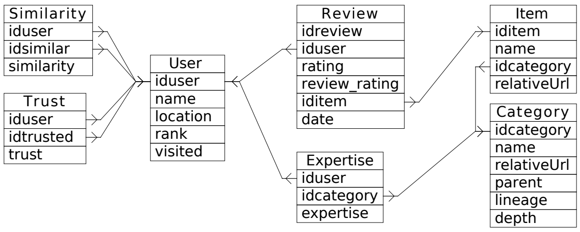


Figure 1: Database schema of the dataset

4.2.1 *Users and Trust*

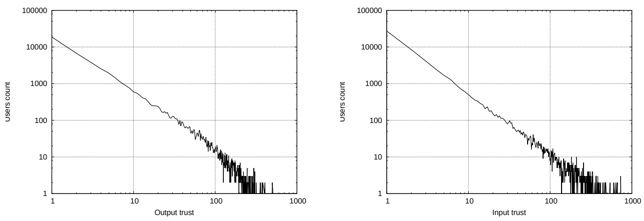
Table 1 provides statistics on four users sets: all users, users with at least one review, users with at least one review and one trust relation and users with at least one review, one trust relation and one computed similarity. We provide for each set its cardinality and the average count of reviews, trust and similarity per user.

Users set	count	review	trust	similarity
all users	131 228	9	4	28
with review	113 629	10	4.5	32
with review and trust	34 410	25	15	95
with review, trust and similarity	21 910	38	20	149

Table 1: Statistics depending on user characteristics

In average, a user has less than one trusted user with a computable similarity: intersection between trusted users and similar users is very small. However, experts have an average of 41 trusted users with a computable similarity.

The output and input trust are equally distributed and follow a power law (fig.2). This is common to main social network datasets. In average, users trust as many users as they are trusted.



(a) Output trust count (b) Input trust count

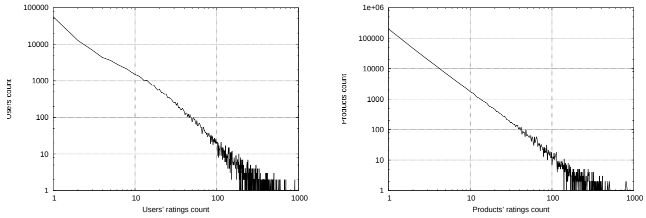
Figure 2: Trust distribution

4.2.2 *Categories and Expertise*

587 categories and sub-categories are provided. Among them, 21 root categories contain experts. 261 users are "experts", *i.e.* category leads, top reviewers or advisor in at least one category. Some of them have several expertises: the dataset contains 556 expertises. Only 261 experts in 131 228 users seem very low, but those experts made 488 217 reviews, *i.e.* almost half reviews. If we take experts with trust (respectively trust and similarity), they are 245 (resp. 241) and have made 463 991 (resp. 463 886) reviews.

4.2.3 *Ratings*

The ratings distribution is as follow: 7.2 % of 1, 7.4 % of 2, 12 % of 3, 30 % of 4 and 43.4 % of 5. We can see the particular distribution of the dataset. It is similar to [MB04] and seems to be the real distribution of the Epinions website.



(a) Users' (b) items'

Figure 3: Ratings count distribution

The ratings count distribution follows a power law (fig.3), a few users made a lot of ratings whereas most users made few ratings. Similarly, a few items has been reviewed many times whereas most items were reviewed a few times.

4.3 Views

In their evaluation, [MA07] introduce views, *i.e.* parts of the dataset grouping particular users or items, that point out the advantages and drawbacks of the evaluated scorers regarding specific contexts. We have adapted them in table **??**.

Category		users/items %	ratings %
Rateness	No raters	13.4	0
	Cold start users	61.3	10.6
	Medium raters	15	15
	Heavy raters	10.3	74.4
Trustness	No trusters	62.6	17.8
	Cold start trusters	24.3	16.3
	Medium trusters	6.7	14
	Heavy trusters	6.4	51.9
Sheepness	Sheep	30.6	37.7
	Gray sheep	15.4	30
	Black sheep	54	32.3
Controversy	Unanimous	72.1	23.9
	Cold controversial	13.9	23.2
	Medium controversial	7.1	25.2
	Heavy controversial	6.9	27.7

Table 2: Views distribution

We define three categories of views for users. Each category defines three disjoint partitions of the users. The "rateness" category considers the number of ratings given by users. *No raters* provide no ratings. *Cold start users* provide between 1 and 4 ratings. *Medium raters* provide between 5 and 15 ratings. *Heavy raters* provide more than 15 ratings. The "trustness" category considers the number of trust values given by users. *No trusters* have no trust relations. *Cold trusters* have between 1 and 4 trust relations. *Medium trusters* have between 5 and 15 trust relations. *Heavy trusters* have more than 15 trust relations. The "sheepness" category considers the rating behaviour of users, it denotes the ability to rate more or less differently from the others: d is the average distance from users ratings to items mean (for each rated item), the bigger d, the more the actor gives ratings different from the majority. *Sheep* are users with $d \leq 0.5$. *Gray sheep* are users with $0.5 < d \leq 0.7$. *Black sheep* are users with $d > 0.7$.

Users categories are not orthogonal: cold start users tend to be cold trusters and heavy raters tend to be heavy trusters.

We define one category of view for items. The "controversy" category considers the standard deviation σ of items ratings. *Unanimous items* have $\sigma = 0$, all users rate them the same. *Cold controversial* are items with $0 < \sigma \leq 0.75$. *Medium controversial* are items with $0.75 < \sigma \leq 1.1$. *Heavy controversial* are items with $1.1 < \sigma$.

We have balanced the two last categories regarding ratings ratio.

4.4 Dataset release

We have released an anonymised version of this dataset at http://liris.cnrs.fr/red. All names and relative urls have been removed, as well as users' location. However, removed data can be provided on demand for non commercial use.

5. EVALUATION WITH THIS DATASET

There exists multiple evaluation approaches. The leave one out approach removes only one rating from the dataset, tries to predict it thanks to other ratings and compare the predicted rating with the removed one and so on with all ratings. This approach is reproductible with a given dataset and easy to implement. However this approach does not take into account the sparsity dimension of the dataset.

Another approach splits the dataset into two disjoint parts: the training part and the evaluation part. The training part is used by the RS to predict the ratings contained in the evaluation part. Those parts need to be split randomly. The size of the training dataset measures the robustness of the RS against sparsity. However this approach needs to split the dataset, which will influence the results. This evaluation must be made several times and its results aggregated, with the same training set size but with different shuffles.

In order to ease evaluation, we have introduced the *ReviewEval* table. This table contains five random values for each review. Those five values *orderField1* to *orderField5* can be used to build five different evaluation shuffles. In order to run evaluation number one, one just needs to sort reviews with the first random value and to split the result set into the training and the evaluation dataset. Here is a sample SQL query in order to shuffle the dataset using the first random coefficient. One can build the 20 % training and 80 % evaluation dataset by appending "`LIMIT 225 534`" for the training dataset and "`LIMIT 225 534, 1 127 673`" for the evaluation dataset:

```
SELECT Review.* FROM ReviewEval
  INNER JOIN Review
  ON (ReviewEval.idreview = Review.idreview)
  ORDER BY orderField1;
```

6. CONCLUSION

This paper presents a dataset usable for the evaluation of many kinds of RS. It aims at providing enough information for Content-Based, Collaborative Filtering and Trust-Based RS. It provides reviews from users to items, items description and users description. The comparaison between the different approaches will be eased using this commonly usable dataset.

7. ACKNOWLEDGEMENT

We thank Epinions for providing an open access to their reviews content through the Epinions website.

8. REFERENCES

[BHKO98] J.S. Breese, D. Heckerman, C. Kadie, and Others, *Empirical analysis of predictive algorithms for collaborative filtering*, Proceedings of the 14th conference on Uncertainty in Artificial Intelligence, Madison: Morgan Kaufmann, 1998, pp. 43–52.

[CKR+07] Meeyoung Cha, Haewoon Kwak, Pablo Rodriguez, Y.Y. Ahn, and Sue Moon, *I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system*, Proceedings of the 7th ACM SIGCOMM conference on Internet measurement, ACM, 2007, pp. 1–14.

[DGKS09] T. DuBois, Jennifer Golbeck, John Kleint, and Aravind Srinivasan, *Improving recommendation accuracy by clustering social networks with trust*, ACM RecSys'09 Workshop on Recommender Systems & the Social Web, Citeseer, 2009.

[GRGP01] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins, *Eigentaste: A constant time collaborative filtering algorithm*, Information Retrieval **4** (2001), no. 2, 133–151.

[JS08] Srikanth Jagabathula and Devavrat Shah, *Inferring rankings under constrained sensing*, Advances in Neural Information Processing Systems (NIPS) (2008), 1–11.

[MA07] Paolo Massa and Paolo Avesani, *Trust-aware recommender systems*, Proceedings of the 2007 ACM conference on Recommender systems (New York, New York, USA), ACM, 2007, pp. 17–24.

[MAL+03] B.N. Miller, Istvan Albert, S.K. Lam, J.A. Konstan, and John Riedl, *MovieLens unplugged: experiences with an occasionally connected recommender system*, Proceedings of the 8th international conference on Intelligent user interfaces, ACM, 2003, pp. 263–266.

[MB04] Paolo Massa and Bobby Bhattacharjee, *Using trust in recommender systems: an experimental analysis*, Trust Management **2995/2004** (2004), 221–235.

[MYLK08] H. Ma, Haixuan Yang, M.R. Lyu, and Irwin King, *Sorec: social recommendation using probabilistic matrix factorization*, 2008, pp. 931–940.

[XKD09] Xin Xin, Irwin King, and Hongbo Deng, *A social recommendation framework based on multi-scale continuous conditional random fields*, Proceeding of the 18th ACM conference on Information and knowledge management (2009), 1247—-1256.

[YZC+09] Quan Yuan, Shiwan Zhao, Li Chen, Yan Liu, Shengchao Ding, Xiatian Zhang, and Wentao Zheng, *Augmenting collaborative recommender by fusing explicit social relationships*, Recommender Systems & the Social Web (2009), 49–56.

[ZMKL05] C.N. Ziegler, S.M. McNee, J.A. Konstan, and Georg Lausen, *Improving recommendation lists through topic diversification*, Proceedings of the 14th international conference on World Wide Web (New York, New York, USA), ACM, 2005, pp. 22–32.