POLITECNICO DI TORINO
Repository ISTITUZIONALE

All you need is ratings: A clustering approach to synthetic rating datasets generation

Original
All you need is ratings: A clustering approach to synthetic rating datasets generation / Monti, D.M., Rizzo, G., Morisio, M.. - ELETTRONICO. - (2019). (13th ACM Conference on Recommender Systems Copenhagen (DK) 16th-20th September 2019).

Availability:
This version is available at: 11583/2749263 since: 2019-09-23T11:52:23Z

Publisher:
Organizers of REVEAL 2019

Published
DOI:

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

09 October 2026

All You Need is Ratings: A Clustering Approach to Synthetic Rating Datasets Generation

Diego Monti
Politecnico di Torino
diego.monti@polito.it

Giuseppe Rizzo
LINKS Foundation
giuseppe.rizzo@linksfoundation.com

Maurizio Morisio
Politecnico di Torino
maurizio.morisio@polito.it

ABSTRACT

The public availability of collections containing user preferences is of vital importance for performing offline evaluations in the field of recommender systems. However, the number of rating datasets is limited because of the costs required for their creation and the fear of violating the privacy of the users by sharing them. For this reason, numerous research attempts investigated the creation of synthetic collections of ratings using generative approaches. Nevertheless, these datasets are usually not reliable enough for conducting an evaluation campaign. In this paper, we propose a method for creating synthetic datasets with a configurable number of users that mimic the characteristics of already existing ones. We empirically validated the proposed approach by exploiting the synthetic datasets for evaluating different recommenders and by comparing the results with the ones obtained using real datasets.

KEYWORDS

Synthetic Dataset, User Clustering, Offline Evaluation

1 INTRODUCTION

It is widely known that novel recommendation approaches should be evaluated in the context of online experiments involving human subjects in order to obtain reasonably robust results about their performance [8]. Nevertheless, most of the studies available in literature support their conclusions with offline trails relying on the preferences of users collected without considering the algorithms under investigation [5]. Despite the possible weaknesses of this approach [14], offline experiments are extremely popular among researchers because of their limited costs and the theoretical reproducibility of their results. In industry, they are usually considered a powerful tool for pruning the number of possible recommender systems that need to be tested with real users, thus mitigating the economical impact of eventual failures.

It is necessary to rely on a collection of user preferences obtained in a particular domain to perform an offline experiment. For example, the MovieLens datasets represent a popular choice for conducting an offline evaluation in the field of movie recommender systems [6]. Nevertheless, the number and the variety of publicly available rating datasets is often limited, especially in less mainstream domains [15]. It is possible to identify different causes for this problem. For example, the companies capable of collecting rating datasets are usually reluctant to share them, because of the fear of violating the privacy of their users or of exposing commercially sensible data to their competitors. On the other hand, researchers often do not have the resources for obtaining a sufficient number of ratings that are worth to be publicly released.

Because of the shortage of public datasets, practitioners have started to rely on synthetic ratings in order to conduct their offline experiments [16]. An obvious advantage of such an approach is that it enables the creation of rating datasets with an arbitrary number of users and items at a limited cost of dataset acquisition. However, the results obtained from such experiments may be questionable, as the generated datasets are usually not capable of capturing the characteristics of a particular domain of interest [11]. For example, different generative approaches only rely on descriptive statistics, like mean and standard deviation, and, for this reason, they fail to mimic the individual behavior of a user.

In this work, we propose a novel approach for automatically generating synthetic datasets with a configurable number of users leveraging on a reference dataset that is used as the seed of the process and that encodes the peculiarities of a domain of interest. Such a generative method can be exploited to create different rating datasets containing users that exhibit behaviors similar to the ones available in the reference dataset. However, the synthetic users do not have a direct relation with the real users and, therefore, no private or commercially sensible information is leaked. At the same time, because the number of synthetic users is configurable, the generated dataset can be exploited to conduct scalability tests in a realistic way and to train recommendation algorithms using reinforcement learning approaches.

More formally, we aim to provide an answer to the following research questions.

- **RQ1** What is the impact of using a synthetic dataset instead of a real one on the results of an offline experiment in the context of recommender systems?
- **RQ2** Can a generative approach be exploited to create a synthetic dataset that exhibits properties similar enough to the ones of a real dataset?
- **RQ3** To what extend this method can be consistently applied to datasets from different domains and of different sizes?

The remainder of this paper is structured as follows: in Section 2 we review related works and we compare them to our proposal. In Section 3 we introduce the generative approach for creating synthetic datasets, while in Section 4 we describe the experimental setup designed to validate it. We present and discuss the results in Section 5 and, in Section 6, we provide the conclusions.

2 RELATED WORK

Synthetic datasets are commonly used in literature to assess the performance of database systems or to study the behavior of data mining algorithms. For example, Agrawal et al. [1] created a generator of retail transactions intended for the evaluation of association rule algorithms, while Houkjær et al. [9] introduced a software capable of creating relational data for benchmarking purposes. Such

tools can generate realistic data in terms of their statistical distributions, which can be empirically learned for existing datasets or provided by a researcher using specialized languages.

Similar approaches have been also explored in the field of recommender systems, usually because of the lack of public datasets with the required characteristics. Tso et al. [15] created a synthetic data generator for evaluating context-aware recommenders based on Dirichlet and Chi-square distributions. The metric of information entropy is then exploited to control the randomness of the synthetic data. A similar method has been discussed by Pasinato et al. [13]: their intuition is to represent the heterogeneous rating behaviors of the users with different statistical distributions.

Manouselis et al. [10] presented a tool, named CollaFis, capable of creating synthetic ratings for the evaluation of either single-criteria or multi-criteria recommender systems. The users of CollaFis need to specify the characteristics of the generated data, like the number of users, items, and criteria. A common aspect of all the previously mentioned methods is that researchers are required to choose and configure the statistical distributions that are exploited to generate the artificial datasets. However, the main problem of such an approach is that it is impossible to predict the real behavior of many different users with a few statistical distributions [11].

Another possible line of research is related to the imitation of a real collection of preferences. For example, CarmenRodríguez Hernández et al. [3] developed a software, DataGenCARS, for creating artificial ratings using a set of parameters provided by the user or inferred from a reference dataset. However, we argue that statistics computed at a global level are not informative enough to create a synthetic dataset, as they are not able to capture the different behaviors of the various groups of users.

3 DATASET GENERATION

Our approach for generating synthetic datasets starting from a reference dataset consists of two steps. In the first one, it is necessary to analyze an existing collection of user preferences in order to obtain an accurate representation of the domain of interest. Then, in the second one, it is possible to exploit such a representation for creating different generated datasets.

We argue that only relying on a few statistical distributions computed empirically at a global level from an existing dataset or specified by a researcher is not sufficient to realistically simulate the individual tastes of human beings [11]. Such methods would lead to the creation of datasets with users having no individual preferences, thus making the task of any recommender system nearly impossible.

For this reason, we included a preliminary clustering phase as part of the first step in order to group the users in a fixed number of communities. The individual rating behaviors, represented by different statistical distributions, are learned for each community and then exploited during the sampling phase.

For simplicity, we assume that each user can only express positive preferences about the items available in the system. However, this approach can also be exploited to simulate datasets with ratings expressed on a more complex scale by repeating these steps for each rating value and then by merging the results.

In the following, we detail the user clustering and distribution learning process and the rating sampling algorithm.

3.1 User clustering and distribution learning

We represent each user $v \in \mathcal{U}$ from the reference dataset as a vector with length equal to the number of items $|\mathcal{I}|$. The component $\hat{v}_i$ of such a vector is equal to 1 if the user v expressed a positive rating ρ about the i-th item of the catalog, otherwise it is equal to 0.

Given this data structure, we decided to apply the K-means clustering algorithm [7] to group together users who liked a similar set of items in K different clusters. The value of K needs to be empirically selected by the experimenter because, in general, it depends on the characteristics of the reference dataset.

Every cluster identifies a different community of users. For generating a dataset similar to the reference one, it is necessary to know how many users belong to each community and what are the item preferences associated with them. More in detail, we create the following empirical distributions from the reference ratings:

- P^C, how users are distributed in K clusters;
- P^U_k, how ratings are distributed in $|\mathcal{U}|$ users for each cluster;
- P^I_k, how ratings are distributed in $|\mathcal{I}|$ items for each cluster.

Note that only the first distribution is global, while the second and the third ones are associated with a cluster.

The distribution P^C represents the probability of assigning a user to a certain cluster and it is computed by counting the number of users per cluster. The distribution P^U_k represents the probability of finding a certain number of ratings per user in the cluster k and it is computed by counting the number of ratings per user. Finally, the distribution P^I_k represents the probability of finding a certain number of ratings per item in the cluster k and it is computed by counting the number of ratings per item.

The user clustering and distribution learning process is formalized in Algorithm 1. Its output is represented by the previously mentioned empirical distributions.

Algorithm 1 User clustering and distribution learning, given a reference dataset and the number of clusters.

Require: $\mathcal{U} \neq \{\emptyset\} \wedge K > 0 \wedge K \leq |\mathcal{U}|$
1: $\mathcal{C} \leftarrow$ K-means$(\mathcal{U}, K)$
2: $P^C \leftarrow P(v \in \mathcal{C}_k)$
3: **for all** $k \in \{1, \ldots, K\}$ **do**
4: $\quad P^U_k \leftarrow P(\rho_v | v \in \mathcal{C}_k)$
5: $\quad P^I_k \leftarrow P(\rho_\imath | \imath \in \mathcal{I}_v \wedge v \in \mathcal{C}_k)$
6: **end for**
7: **return** $P^C,\ P^U_k,\ P^I_k$

3.2 Rating sampling

Starting from the empirical distributions obtained from Algorithm 1, it is possible to generate a synthetic dataset by applying to them a sampling function σ. In the following, we assume that σ is the weighted random sampling function.

As discussed in Section 1, the experimenter can select the number of users available in the generated dataset. This value, called U, is an input of the rating sampling algorithm, together with the

probability distributions. The synthetic dataset can also have the same number of users in the reference dataset, that is $U = |\mathcal{U}|$.

Firstly, each generated user u is assigned to a cluster k from the reference dataset, according to the distribution of users per cluster. Then, the number of ratings I for that user is selected considering the distribution of ratings per user in the cluster k. Finally, I items are sampled without replacement ($\hat{\sigma}$) from the distribution of ratings per item in the cluster k. Thus, the number of user ratings and her liked items are associated with a particular community of users.

The rating sampling procedure is formalized in Algorithm 2.

Algorithm 2 Rating sampling, given the required number of users and the distributions computed in Algorithm 1.

Require: $U > 0, P^C, P^U_k, P^I_k$
$\mathcal{R} \leftarrow \{\emptyset\}$
for all $u \in \{1, \ldots, U\}$ **do**
  $k \leftarrow \sigma(P^C)$
  $I \leftarrow \sigma(P^U_k)$
  for all $i \in \{1, \ldots, I\}$ **do**
    $\rho_{u,i} \leftarrow \hat{\sigma}(P^I_k)$
    $\mathcal{R} \leftarrow \mathcal{R} \cup \{\rho_{u,i}\}$
  end for
end for
return $\mathcal{R}$

4 EXPERIMENTAL SETUP

We compared the results obtained from the evaluation of different recommenders conducted on popular datasets typically exploited in literature with the ones computed in the same experimental conditions using various collections of synthetic preferences generated starting from them using multiple techniques.

In fact, we claim that a synthetic dataset can be successfully used during an evaluation campaign if the behavior of the recommender systems under analysis is similar to one that it would be possible to observe with the reference dataset. Thus, almost all the possible pairs of recommenders should exhibit the same relation of order for a given dimension and lead to similar conclusions.

Furthermore, we investigated what is the impact of the parameter K on the results of the evaluation, in order to understand how to empirically select the most appropriate value for it.

In our experiments, we utilized Random, Most Popular, User KNN, BPRMF, and WRMF recommendation algorithms and the metrics of precision, recall, and NDCG as defined in the evaluation framework RecLab [12]. Regarding the user preferences, we exploited the binarized versions of the MovieLens 100K, MovieLens 1M, and LastFM [2] datasets. We considered as positive all ratings with a value higher than 3 for MovieLens and than 0 for LastFM. We relied on the default values of the evaluation framework for all other experimental parameters: we followed a random splitting protocol with a test set size equal to the 20% of all available ratings and we recommended 10 items for each test user.

From the aforementioned reference datasets we generated their synthetic versions exploiting the procedure described in Section 3. We considered U equal to the number of users originally available, in order to compare datasets of similar size. Furthermore, we also created three baseline synthetic collections with the same number of ratings by not applying the user clustering phase. All the users of such baselines exhibit the same rating behavior, similarly to the approach described in [3]. In Table 1, we report different statistics regarding the baseline, generated, and reference datasets.

Table 1: The total number of users, items, and ratings available in the datasets under consideration.

Dataset	Version	Users	Items	Ratings
MovieLens 100K	Baseline	942	1,374	55,375
MovieLens 100K	Generated	942	1,332	53,915
MovieLens 100K	Reference	942	1,447	55,375
MovieLens 1M	Baseline	6,038	3,463	575,281
MovieLens 1M	Generated	6,038	3,457	584,101
MovieLens 1M	Reference	6,038	3,533	575,281
LastFM	Baseline	1,888	13,342	92,834
LastFM	Generated	1,892	13,442	92,510
LastFM	Reference	1,892	17,632	92,834

Table 2: The values of precision obtained with the generated versions of the MovieLens 100K dataset by varying K.

Dataset	Most Popular	User KNN	BPRMF	WRMF
K = 5	0.088449	0.099890	0.078768	0.091749
K = 10	0.095793	0.124595	0.102805	0.111974
K = 50	0.098378	0.133946	0.103243	0.133838
K = 100	0.102415	0.150494	0.115587	0.149945
K = 200	0.099672	0.154158	0.122538	0.164114

5 RESULTS

In this section, we first discuss the impact of the number of user communities on the evaluation results, then we present a comparison between exploiting the synthetic and the reference datasets.

5.1 Number of user communities

For studying what is the impact of the value K on the results of an evaluation conducted with a synthetic dataset, we computed the measure of precision on different synthetic versions of the MovieLens 100K dataset created with $K = \{5, 10, 50, 100, 200\}$. We report the numerical outcomes of this experiment in Table 2.

We also observed that it is possible to obtain similar results by considering other datasets and metrics. As expected, the values of precision for all the algorithms but the Random and Most Popular approaches improve by increasing the number of available clusters. However, this relationship is not linear, as doubling its value from 100 to 200 only slightly improves the results.

We empirically observed that reasonable values for K could be 100 or 200. In Section 5.2, we will assume that $K = 200$.

Therefore, we can provide an answer to **RQ1** by observing that the impact of using a synthetic dataset in an evaluation campaign can be mitigated if we are able to simulate a sufficient number of heterogeneous user communities.

Table 3: The results obtained with the baseline, generated, and reference versions of MovieLens 100K.

	Baseline dataset			Generated dataset			Reference dataset		
Algorithm	Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
Random	0.009416	0.008877	0.009841	0.009847	0.008977	0.010022	0.007743	0.006300	0.008183
Most Popular	0.060065	0.053209	0.064384	0.099672	0.083875	0.110229	0.112759	0.102804	0.130632
User KNN	0.055952	0.050587	0.058744	0.154158	0.135917	0.169499	0.205234	0.221684	0.233362
BPRMF	0.045346	0.033628	0.048740	0.122538	0.106186	0.129742	0.182770	0.186838	0.198869
WRMF	0.047078	0.042876	0.048104	0.164114	0.144272	0.173916	0.221592	0.233235	0.250386

Table 4: The results obtained with the baseline, generated, and reference versions of MovieLens 1M.

	Baseline dataset			Generated dataset			Reference dataset		
Algorithm	Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
Random	0.004989	0.002733	0.004803	0.005712	0.002678	0.005580	0.005589	0.002862	0.005657
Most Popular	0.066483	0.037723	0.068991	0.101570	0.061565	0.107356	0.131982	0.082978	0.142782
User KNN	0.064708	0.035976	0.066573	0.129113	0.078716	0.136779	0.232082	0.172290	0.262018
BPRMF	0.057459	0.027145	0.059226	0.106948	0.057729	0.111058	0.199633	0.136378	0.218727
WRMF	0.053022	0.027763	0.055035	0.133723	0.080045	0.140670	0.227878	0.154425	0.252999

Table 5: The results obtained with the baseline, generated, and reference versions of LastFM.

	Baseline dataset			Generated dataset			Reference dataset		
Algorithm	Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
Random	0.000691	0.000656	0.000869	0.000532	0.000520	0.000548	0.000797	0.000825	0.000791
Most Popular	0.046281	0.046513	0.048636	0.052844	0.054597	0.056285	0.067906	0.068970	0.075406
User KNN	0.042614	0.043088	0.044308	0.101435	0.104549	0.113433	0.156057	0.160451	0.189487
BPRMF	0.039957	0.040543	0.041804	0.062307	0.064720	0.066835	0.075877	0.077336	0.087066
WRMF	0.032731	0.032974	0.033975	0.090324	0.092860	0.099598	0.160202	0.164468	0.193937

5.2 Synthetic and reference datasets

As anticipated in Section 4, we compared the evaluation results obtained when relying on the reference dataset and two synthetic datasets created with different approaches. We repeated this experiment with datasets of different sizes and from different domains in order to assess the generalizability of the results.

The results obtained with MovieLens 100K, MovieLens 1M, and LastFM are available in Table 3, Table 4, and Table 5 respectively.

We observe that in all experiments and for almost all the possible pairs of recommenders the relative order of the measures is the same between the generated and the reference datasets.

As expected, their values are lower when exploiting the synthetic ratings, as they do not represent real preferences. Nevertheless, they are still useful to identify the most promising recommendation techniques in a certain domain, while the results obtained with the baseline datasets cannot be exploited for such a purpose.

With respect to **RQ2**, we can conclude that a generative approach capable of replicating the behaviors of different groups of users can be used for creating realistic datasets. We also discovered, as an answer to **RQ3**, that our approach can be potentially applied to datasets from different domains and of different sizes.

6 CONCLUSION AND FUTURE WORK

In this paper, we have discussed a method for generating synthetic datasets with an arbitrary number of users starting from existing collections of preferences. Differently from the approaches already available in literature, we propose to first model user communities in order to generate more realistic ratings that can be successfully exploited during an evaluation campaign.

We empirically verified that the outcome of an offline comparison among different recommender systems conducted exploiting the generated datasets is consistent with the results obtained when using the reference datasets, provided that a sufficient number of user clusters is selected. This finding could encourage private companies to publicly release synthetic datasets created from internally available data without the fear of violating the privacy of their users or of exposing commercially sensible information.

As future work, we would like to explore additional methods for creating synthetic datasets. We believe that Generative Adversarial Networks (GANs) could be successfully exploited for this task, as they are already used to generate fake images starting from real ones [4]. Such approaches would require the definition of a way for representing the preferences of a user similarly to an image.

REFERENCES

[1] Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB '94)*. Morgan Kaufmann Publishers, Burlington, MA, USA, 487–499.

[2] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. 2011. Second Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec2011). In *Proceedings of the Fifth ACM Conference on Recommender Systems (RecSys '11)*. ACM, New York, NY, USA, 387–388. https://doi.org/10.1145/2043932.2044016

[3] María del Carmen Rodríguez-Hernández, Sergio Ilarri, Ramón Hermoso, and Raquel Trillo-Lado. 2017. DataGenCARS: A generator of synthetic data for the evaluation of context-aware recommendation systems. *Pervasive and Mobile Computing* 38 (jul 2017), 516–541. https://doi.org/10.1016/j.pmcj.2016.09.020

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, Vol. 2. MIT Press, Cambridge, MA, USA, 2672–2680.

[5] Asela Gunawardana and Guy Shani. 2015. Evaluating Recommender Systems. In *Recommender Systems Handbook* (2 ed.). Springer Publishing, New York, NY, USA, Chapter 8, 265–308. https://doi.org/10.1007/978-1-4899-7637-6_8

[6] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Transactions on Interactive Intelligent Systems* 5, 4 (2015), 1–19. https://doi.org/10.1145/2827872

[7] John A. Hartigan and Manchek A. Wong. 1979. Algorithm AS 136: A K-Means Clustering Algorithm. *Applied Statistics* 28, 1 (1979), 100. https://doi.org/10.2307/2346830

[8] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* 22, 1 (2004), 5–53. https://doi.org/10.1145/963770.963772

[9] Kenneth Houkjær, Kristian Torp, and Rico Wind. 2006. Simple and Realistic Data Generation. In *Proceedings of the 32nd International Conference on Very Large Data Bases (VLDB '06)*. VLDB Endowment, Los Angeles, CA, USA, 1243–1246.

[10] Nikos Manouselis and Constantina Costopoulou. 2008. Preliminary Study of the Expected Performance of MAUT Collaborative Filtering Algorithms. In *The Open Knowlege Society. A Computer Science and Information Systems Manifesto*. Springer Publishing, New York, NY, USA, 527–536. https://doi.org/10.1007/978-3-540-87783-7_67

[11] Miquel Montaner, Beatriz López, and Josep Lluís de la Rosa. 2004. Evaluation of recommender systems through simulated users. In *Proceedings of the 4th International Workshop on Pattern Recognition in Information Systems*. SciTePress, Setúbal, Portugal, 1–6. https://doi.org/10.5220/0002622703030308

[12] Diego Monti, Giuseppe Rizzo, and Maurizio Morisio. 2018. A Distributed and Accountable Approach to Offline Recommender Systems Evaluation. In *Proceedings of the Workshop on Offline Evaluation for Recommender Systems at the 12th ACM Conference on Recommender Systems*. REVEAL 2018, Vancouver, BC, Canada, Article 6, 5 pages. https://arxiv.org/abs/1810.04957

[13] Marden Pasinato, Carlos Eduardo Mello, Marie-Aude Aufaure, and Geraldo Zimbrao. 2013. Generating Synthetic Data for Context-Aware Recommender Systems. In *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*. IEEE, New York, NY, USA, 563–567. https://doi.org/10.1109/brics-cci-cbic.2013.99

[14] Alan Said and Alejandro Bellogín. 2014. Comparative Recommender System Evaluation: Benchmarking Recommendation Frameworks. In *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys '14)*. ACM, New York, NY, USA, 129–136. https://doi.org/10.1145/2645710.2645746

[15] Karen H. L. Tso and Lars Schmidt-Thieme. 2006. Empirical Analysis of Attribute-Aware Recommendation Algorithms with Variable Synthetic Data. In *Studies in Classification, Data Analysis, and Knowledge Organization*. Springer Publishing, New York, NY, USA, 271–278. https://doi.org/10.1007/3-540-34416-0_29

[16] Hsiang-Fu Yu, Cho-Jui Hsieh, Si Si, and Inderjit Dhillon. 2012. Scalable Coordinate Descent Approaches to Parallel Matrix Factorization for Recommender Systems. In *2012 IEEE 12th International Conference on Data Mining*. IEEE, New York, NY, USA, 765–774. https://doi.org/10.1109/icdm.2012.168