International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 11, No.1 (Mar-2022)

http://dx.doi.org/10.12785/ijcds/110192

Using Association Rule Mining to Enrich User Profiles with Research Paper Recommendation

Lule Ahmedi[1], Edonit Rexhepi[1] and Eliot Bytyçi[2]

[1]*Department of Computer Engineering, FECE, University of Prishtina “Hasan Prishtina”, Prishtina, Kosovo*
[2]*Department of Mathematics, FMNS, University of Prishtina “Hasan Prishtina”, Prishtina, Kosovo*

Received 10 Jun. 2021, Revised 19 Nov. 2021, Accepted 3 Feb. 2022, Published 31 Mar. 2022

Abstract: Association rules are used in recommender systems to develop a model that enhances the profiles of users, as well as to address the cold start problem. Our approach proposes a model which is implemented in a system for recommending scientific papers called Collaborative Topic Regression (CTR). Collaborative Topic Regression consists of two matrices, U and V, where U represents the relationship between users and paper topics, while V represents the relationship between papers and the paper topics. The CTR model is focused on matrix V, by adopting it, and as the result, it is influenced by the paper’s textual content, leaving the content of matrix U essentially unchanged. The depicted model extracts association rules from matrix U, and then enriches it, with the information gleaned from the mining process. The outcomes are based on both, out-of-matrix prediction and in-matrix prediction. Our approach improved the quality of the results by up to 20% for out-of-matrix prediction in the best-case scenario. Unfortunately, the same cannot be said for in-matrix prediction, which will be further investigated in the future.

Keywords: Association Rule Mining, Prediction, Recommendation, Collaborative Topic Regression, In-matrix, Out-of-matrix

1. Introduction

Association Rule Mining (ARM) [1] was used initially in analyzing market basket data of point of sale registered transactions. An information gained can be used by the markets for further increase of sale. Even though there exists a clear connection between association rules and the aim of the Recommender Systems (RS), ARM was seldom used in the RS. Reasons for this is that the approach is similar to Collaborative Filtering (CF), but less flexible in the definition of the transactions. The transactions that are to be used with ARM, should be defined in more details. That served as a motivation to further research the inter-relation between the two subjects.

A general model that uses ARM to enrich user profiles, is based on resulted rules from mining those profiles. It is based on a system for recommendation of research papers named Collaborative Topic Regression (CTR) [2]. CTR algorithm uses information from other users’ profiles and works very well when we recommend common papers (in users’ profiles) but cannot be generalized also for new papers. In order to solve the problem and find abstract terms in the set of documents, CTR algorithm uses topic modeling [3]. An example of topic modeling is the Latent Dirichlet Allocation (LDA), which creates a topic from a document based on Dirichlet distribution.

Collaborative Topic Regression is based on applying LDA algorithm on matrix D, which holds the representation of m papers as a dictionary T, to draw k latent topics. Another matrix R, contains reviews of *n* users for *m* papers.

The next step is to apply matrix factorization to get two latent matrices from matrix R:

- U: latent matrix users / topics and
- V: latent matrix papers / topics.

After that, it’s time to alter matrix V so that it’s affected by textual paper content, allowing matrix to influence the matrix factorization method created by matrix V. Finally, a new matrix R’ is created by multiplying U with transposed V in order to generate recommendations.

The rest of the paper is organized as follows: Section 2 provides related work, while Section 3 describes the model and Section 4 provides experimental results. We conclude the paper with discussions and future work.

2. Related work

Authors in [4] have presented a collaborative recommendation technique based on a new algorithm specifically designed to mine association rules. They have divided the

E-mail address: lule.ahmedi@uni-pr.edu, edoniti@gmail.com, eliot.bytyci@uni-pr.edu - Corresponding author

mining process in two parts: mining of user association rules and mining of papers association rules. The strategy is to combine the results of both approaches, by checking the support of one and the other. If the support is lower than the threshold on the user's part, then ARM of the users is used, otherwise papers association rule mining is used. Our model is different from theirs, since their model uses only association rule for recommendation and thus not combining them with other RS.

In another similar paper [5], authors developed a model that separates papers in favorite ones and the non-favorite ones. Through a pre-defined threshold, papers above the threshold were considered favorites, and the ones below as non-favorite. Model recommends papers based on two approaches: using ARM for favorite papers and content-based filtering for non-favorite ones. The difference with our model is that even though it is considered as hybrid approach, it does not combine the approaches but uses them separately for different papers.

One of the most similar approaches to our model [6], enriches user profiles by using ARM but with the distinction from our work, it takes user profiles only as binary values. Therefore, if a user has a specific paper in her/his profile, that paper does not have a specific weight for the user. As such, the model cannot be implemented in RS where papers have different weight for every user, as in the case of CTR, where our model has been implemented.

The most comparable research model to our approach is the model presented in [2]. The main aim of the approach is to recommend research papers to users of an online community. The approach combines traditional CF merits and papers probabilistic modeling. The CTR algorithm used, has shown to be more effective than traditional CF. Our approach is distinct from that of the authors in [2], where only matrix V-papers are used, by incorporating also the other matrix U-users.

3. Description of the model

The user profiles and the content of the paper are both used in the Collaborative Topic Regression model. The system can locate previous key publications for each user, as well as fresh papers with material that reflects the user's individual interests. Every recommender system with user profiles can benefit from our concept. This could be a matrix comprised of [users X papers], where every cell would present the relation between a user and a paper. For the purpose of testing the model, it is implemented in a CTR recommender system for recommending scientific articles.

The model includes six key processes: discretizing, one-hot encoding, zero probability case removal, mining association rules, identifying new terms and enriching user profiles.

A. Abbreviations and Acronyms

Discretization is the process of transforming continuous variables into discrete ones [7]. The procedure is necessary because the domain of matrix values that represent user profiles can be quite large. As a result, discretization allows us to narrow down the values. Only one parameter, the number of bins, is required for discretization.

B. Units

One-hot encoding process converts categorical values in vectors that contain binary values [8]. This is necessary since mining association rule accepts only transactions with binary values. In the case when the user profile contains real value ranging from 0 to 1, then a conversion of those profiles in acceptable transactions should be done, for mining association rules. In that case, values bigger than 0 are converted to 1 and 0 values would be left as 0. It is evident that we would lose in the quality of the data, since the difference between the first user (U1) that has a paper: A2 with probability 0.03 and the second user (U2) that has paper: A2 with probability 0.8, is wide.

In the case when the user profile contains other values (not real values from 0 to 1) it would be impossible to convert them directly into values 0 and 1. Therefore, the process one-hot encoding is very useful. If it is the case that a paper could have three possible categorical values (S1, S2, S3 derived from the discretization step), then paper value could be converted to a vector with three numerical values:

- S(X)1 = [1, 0, 0]
- S(X)2 = [0, 1, 0]
- S(X)3 = [0, 0, 1]

As we can see from Table 1 and Table 2, after the one-hot encoding has been applied, the dimensions of term vector has increased, and a ready matrix for mining association rules is gained.

	T1	T2	T3	T4
U1	S(T1)1	S(T2)2	S(T3)1	S(T4)1
U2	S(T1)2	S(T2)1	S(T3)1	S(T4)2
U3	S(T1)2	S(T2)3	S(T3)2	S(T4)3

TABLE I. User profile before applying one-hot encoding

	P1	P2	P3	P4
U1	100	010	100	100
U2	010	100	100	010
U3	010	001	010	001

TABLE II. User profile after applying one-hot encoding

C. Zero probability cases removal

Since in the cases when the probability of a paper is zero, the discretization process gathers them in a bucket.

After that, the one-hot encoding process is performed, providing thus a numerical value vector. These cases have to be treated in order to not damage the quality of the rules. Treatment is done through 2 steps:

- first additional step is performed after discretization, when the values with probability zero are divided into an additional bucket. For example, if the separation is done in 5 buckets, then those values with probability zero are put in the bucket 6. For the process of one-hot encoding it is easier than to see which values of the vector represent these cases.

- second additional step is performed after the process of one-hot encoding, when the vector representing cases with probability zero, [0 0 0 0 0 1], is represented as [0 0 0 0 0 0] and therefore will not be treated during the association rule mining process.

D. Association rule mining

After the process of one-hot encoding, a ready matrix has been gained for the mining of association rules. This matrix is provided to any standard algorithm for ARM and in our case the algorithm chosen is FP-growth since according to research in [9] [10] it secures better results than other algorithms.

During the process, standard parameters for mining association rules are provided, such as support and confidence.

E. Identifying new terms

By using the rules gained from the process, new terms for the user profiles are identified. During this process, our algorithm accepts following parameters:

- Sequence of the rules: before we test them for their validity. This could be done according to trust or elevation, where as a result only top X rules are chosen.

- Number of rules: how many top rules should be chosen for each profile.

F. Enriching the profiles

Enrichment of profiles represents last process in the model, where user profiles are enriched based on the results from other processes and especially from the process of finding new terms. This process accepts the following parameters:

- Type and number of threshold: in order to get the profiles to be enriched. In our implementation there exist two thresholds: topic and paper, but we could leave one threshold and therefore enriching all profiles. For example, if we choose only the topic as a threshold and the number 10, then our algorithm will enrich only those profiles with less than 10 topics.

- Enrichment result: through this parameter a final result could be chosen to be added to a profile, such as in our case: trust, average of the term probability and average of trusted term probability.

4. Experimental result

We have conducted experiments on same data used by CTR, in order to make it easier to compare results obtained from our model with CTR results. CTR uses user data and their work profiles obtained from CiteULike, where registered users create their own profiles consisting of scientific papers [2]. From each paper we will use title and abstract but discard other information such as authors, publications and keywords. The authors of CTR have deleted duplicate papers, blank papers, and users with less than 10 papers, thus resulting in a data set consisting of 5,551 users, 16,980 papers and 204,986 user-paper combinations. On average, each user has 27 papers in their profile, ranging from 10 to 403 papers, with 93% of users having less than 100 papers. For each article, the title and the abstract are merged. Stop words have been deleted and TF-IDF has been used to select the 8,000 unique key-words as a dictionary. This has created a corpus of 1.6 million words. It should be noted that papers were added to CiteULike between 2004 and 2010. On average, each paper appears in 12 users, ranging from 1 to 321 times, where 97% of papers appear in less than 40 profiles.

A tool developed by Anas Alzogbi, research assistant at the Albert Ludwig University of Freiburg was used to do the sharing and testing of the results. This tool has the following functionalities:

1) Split user ratings into training and validation data. This partitioning is done based on several methods of partitioning, in this paper we used in-matrix and out-of-matrix partitions. The partition model used in this tool is K-fold Cross-validation.
2) Evaluation of recommendation results in validation data based on two metrics: Recall and Mean Reciprocal Rank (MRR).

Cross-validation as a statistical method for evaluating and comparing algorithms, divides data into two segments, one used to train the model and the other one to validate it[8]. The basic form of cross-validation is cross-validation with K folds. In this validation, data are initially divided into K-folds. Then K replicates are made, so that in each iteration a different fold is used for model validation while the remaining K - 1 folds are used for model training.

Metrics for evaluating the results are: Recall and Mean Reciprocal Rank (MRR). Recall [4] represents the rapport between true positives and the sum of true positives with false negatives, mathematically represented in formulae 1:

$$recall = \frac{TP}{TP + FN} \tag{1}$$

Reciprocal Rank (RR) is a measurement that calculates

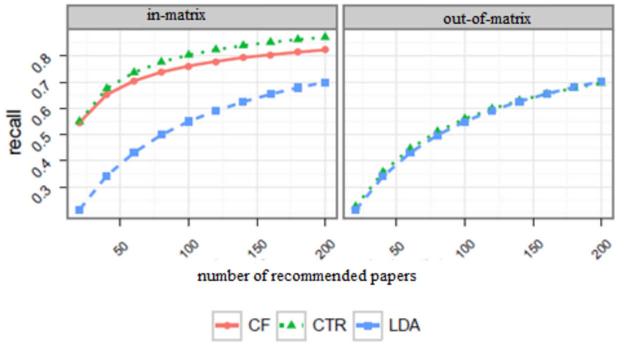


Figure 1. Example of in-matrix and out-of-matrix recommendations

the rank where the first document is appropriately to be found[8]. The mean of RR, represents the mean of all the document request. Since in our case, we are not sure about the papers that are not in user profiles, we could not use another metric called precision, since it is difficult to calculate it.

There exist two types of predictions: in-matrix prediction and out-of-matrix prediction. In-matrix prediction refers to the problems where recommendations are provided by at least one user and can be taken care of also with traditional methods of collaborative filtering. Out-of-matrix prediction refers to the problem of not being able to provide recommendations for new papers, if none of the users provided recommendations. This is a problem for traditional methods.

As we can see from Figure 1, where an example is provided suggesting that the CF cannot even provide recommendations for out-of-matrix model.

A. *In-matrix prediction*

All results from the experiments performed are compared with CTR model results. In the results of the evaluation measurements, the number indicates how much the first highest evaluation papers were taken into account. For example, Rec @ 20 means the result of the Recall measurement when considering the first 20 papers with the highest rating. Due to the large volume of experiments, it is worth noting that that we will be presenting only a few experiments which have been considered more reasonable to present.

First experiment is done by using data parameters as presented in Table 3, parameters that are valid for division of the data used by our model and by the CTR. After that, few changes have been made to the specific parameters of our model, to see the effect those parameters have in results.

By using these parameters, results presented in Table 4 are gained from the CTR model. These results are same for all cases of first experiment since the parameters have not changed during the whole first experiment.

Parameter	Value
Division process	
Division method	in-matrix
Number of folds	5

TABLE III. Division data parameters for first experiment

Fold	Rec@5	Rec@10	Rec@20	Rec@40	Rec@60	Rec@80	Rec@100	Rec@160	Rec@200	MRR@10
1	0.229	0.334	0.458	0.587	0.661	0.707	0.740	0.798	0.823	0.427
2	0.228	0.333	0.462	0.590	0.658	0.705	0.737	0.800	0.823	0.431
3	0.233	0.341	0.467	0.593	0.663	0.708	0.738	0.797	0.819	0.438
4	0.235	0.346	0.474	0.599	0.667	0.712	0.744	0.801	0.824	0.437
5	0.176	0.277	0.407	0.556	0.641	0.694	0.732	0.805	0.834	0.515
Average	0.220	0.326	0.454	0.585	0.658	0.705	0.738	0.800	0.825	0.450

TABLE IV. Results from measurements of CTR model in first experiment

It should be noted that for each experiment in in-matrix prediction we have used parameters presented in Table 5, but for each experiment with different values.

	Parameter	Value Exp1	Value Exp2	Value Exp3	Value Exp4
Division process	Division method	in-matrix	in-matrix	in-matrix	in-matrix
	Number of folds	5	5	5	5
Discretization process	Number of bins	5	10	10	10
ARM process	Minimal support	40%	40%	30%	30%
	Minimal confidence	50%	50%	40%	40%
Process of finding new terms	Sequence of rules	Elevation	Elevation	Elevation	Elevation
	Number of rules	10	10	10	10
Enrichment of profiles	Type of threshold	Nr. of papers	Nr. of papers	Nr. of papers	Nr. of papers
	Number of threshold	avg. nr. papers per profile / 3	avg. nr. papers per profile / 3	avg. nr. papers per profile / 3	avg. nr. papers per profile / 3
	Enrichment result	Trust	Trust	Trust	Probabilistic average

TABLE V. Parameters of the model for in-matrix prediction experiments 1.1-1.4

Following the experiments, we discovered that our model failed to improve the CTR model's in-matrix prediction performance, as shown in Table 6. The next stage is to look for out-of-matrix predictions in the experiment.

Average	Rec@5	Rec@10	Rec@20	Rec@40	Rec@60	Rec@80	Rec@100	Rec@160	Rec@200	MRR@10
Model used	0.220	0.326	0.453	0.584	0.657	0.704	0.737	0.799	0.824	0.448
CTR	0.220	0.326	0.454	0.585	0.658	0.705	0.738	0.800	0.825	0.450
Difference	0.000	0.000	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.002
Difference %	0.00%	0.00%	-0.22%	-0.17%	-0.15%	-0.14%	-0.14%	-0.13%	-0.12%	-0.44%
Average	0.220	0.326	0.454	0.585	0.658	0.705	0.738	0.800	0.825	0.450

TABLE VI. Comparing the average values between the model used and CTR for experiment 1.4

B. *Out-of-matrix prediction*

The Table 7 presents parameters used for out-of-matrix prediction experiments 2.1 until 2.3. In all of the cases, starting from experiment 2.1 we see improvement as presented in Table 8.

In all of the other experiments for out-of-matrix prediction, our model achieved to improve the CTR model result, at best for 20% (from 0.005 to 0.006) as seen in Table 9. It is worth noting that in cases where we reduce the number of gained association rules or reduced the profiles selected for enrichment, we find that our model approximates to the CTR model. It may be thought that we have achieved a better model in these cases, but the truth is that in these cases our model has less impact on the CTR model (in some cases it does not affect it at all), therefore the results are more related.

	Parameter	Value Exp1	Value Exp2	Value Exp3
Division process	Division method	out-of-matrix	out-of-matrix	out-of-matrix
	Number of bins	5	5	5
Discretization process	Number of bins	5	5	5
ARM process	Minimal support	40%	40%	40%
	Minimal confidence	50%	50%	50%
Process of finding new terms	Sequence of rules	Elevation	Elevation	Elevation
	Number of rules	10	10	10
Enrichment of profiles	Type of threshold	Nr. of papers	Nr. of papers	Nr. of papers
	Number of threshold	avg. nr. papers per profile / 3	avg. nr. papers per profile	no threshold
	Enrichment result	Trust	Trust	Trust

TABLE VII. Parameters of used model for out-of-matrix prediction experiments 2.1-2.3

Average	Rec@5	Rec@10	Rec@20	Rec@40	Rec@60	Rec@80	Rec@100	Rec@160	Rec@200	MRR@10
Model used	0.005	0.011	0.028	0.081	0.145	0.209	0.268	0.419	0.496	0.022
CTR	0.005	0.011	0.027	0.078	0.141	0.203	0.263	0.414	0.490	0.021
Difference	0.000	0.000	+0.001	+0.003	+0.004	+0.006	+0.006	+0.005	+0.006	+0.001
Difference %	0.00%	0.00%	+3.70%	+3.85%	+2.84%	+2.96%	+1.90%	+1.21%	+1.22%	+4.76%

TABLE VIII. Comparing the average values between the model used and CTR for experiment 2.1

In order to analyze why our model did not perform better than the CTR model for in-matrix prediction, the recommendation process was manually recreated for certain users, where not so good results were observed in our model. According to the findings, our model discovered several subjects in the validation data for those users that were not related to the publications. For example, for a given user our model added topic T20, to that user's profile, with a probability of 0.625, while of the three papers that that user had, the validation data had a probability 0 in the topic T20. This has caused the similarity between the user profile and the paper in question to be smaller in our model than in the CTR model. We determined from the analysis that the quality of data mining had an impact in this case.

5. Discussion and Future Work

A method of merging associative rules with recommendation systems has been demonstrated using the model described in this study. By supplementing user profiles with associative rules, this paradigm facilitates the introduction of associative rules into existing recommendation systems.

This model can also be used to analyze other articles and in recommender systems that cope with non-binary variables. This feature has not been seen in any other model. The main contributions include the development of a novel model for merging associative rules with recommendation systems, as well as the use of associative rules to enhance user profiles. Another contribution is that the model has been integrated in an existing recommendation system that promotes scientific papers in order to test its effectiveness. Our investigations and tests revealed that this model increased the recommendation system score by at most 20% (11.57 percent on average) for out-of-matrix and failed to improve the result for within-matrix.

Average	Rec@5	Rec@10	Rec@20	Rec@40	Rec@60	Rec@80	Rec@100	Rec@160	Rec@200	MRR@10
Model used	0.006	0.012	0.031	0.090	0.158	0.227	0.290	0.443	0.518	0.023
CTR	0.005	0.011	0.027	0.078	0.141	0.203	0.263	0.414	0.490	0.021
Difference	+0.001	+0.001	+0.004	+0.012	+0.017	+0.024	+0.027	+0.029	+0.028	+0.002
Difference %	+20.00%	+9.09%	+14.81%	+15.38%	+12.06%	+11.82%	+10.27%	+7.00%	+5.71%	+9.52%

TABLE IX. Comparing the average values between the model used and CTR for experiment 2.3

There is further work to be done in the future with a specific focus on the data mining process, as the data in this study has been converted to function on traditional data mining platforms. As a result, research should concentrate on specialized algorithms for mining associative rules where transactions have weighted values (non-binary values).

References

[1] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Washington D.C., May 1993, pp. 207–216.

[2] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011, pp. 448–456.

[3] S. Li, "Topic modeling and latent dirichlet allocation (lda) in python," *: https://towardsdatascience. com/topic-modeling-and-latent-dirichletallocation-in-python-9bf156893c24*, 2018.

[4] W. Lin, S. A. Alvarez, and C. Ruiz, "Collaborative recommendation via adaptive association rule mining," *Data Mining and Knowledge Discovery*, vol. 6, no. 1, pp. 83–105, 2000.

[5] A. Alsalama, "A hybrid recommendation system based on association rules," 2013.

[6] G. Shaw, Y. Xu, and S. Geva, "Using association rules to solve the cold-start problem in recommender systems," in *Pacific-Asia conference on knowledge discovery and data mining*. Springer, 2010, pp. 340–347.

[7] A. Burkov, *The hundred-page machine learning book*. Andriy Burkov Quebec City, Can., 2019.

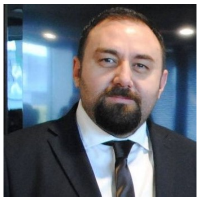
[8] P. Refaeilzadeh, L. Tang, and H. Liu, "Cross-validation." *Encyclopedia of database systems*, vol. 5, pp. 532–538, 2009.

[9] T. A. Kumbhare and S. V. Chobe, "An overview of association rule mining algorithms," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 1, pp. 927–930, 2014.

[10] K. Garg and D. Kumar, "Comparing the performance of frequent pattern mining algorithms," *International Journal of Computer Applications*, vol. 69, no. 25, 2013.

Lule Ahmedi is a full professor at University of Prishtina, Faculty of Electrical and Computer Engineering since 2005. She received her PhD in 2004 in computer science from University of Freiburg, Germany, where she worked from 1999 to 2004 in teaching, and as a researcher in a German Research Foundation (DFG) project "Spontaneous Integration of Heterogeneous Information from Web". From 2005 to 2010 she was also affiliated to the South East European University as visiting lecturer. Since 2010 she occasionally provides guest teaching to Linnaeus University and Norwegian University of Science and Technology. Her research interests relate always to data or the web, to mention: data science, machine learning, recommender systems, semantic web, and social network analysis (DBLP, Google Scholar). She is author and has managed numerous ICT projects through fundings by EU, and German, Norwegian, Swedish, and Kosovo national funds. She continuously serves to consultancy boards related to research, and is recipient of several awards. Check Prof. Ahmedi's Homepage for more information, GitHub for source code, Mendeley for datasets.

Edonit Rexhepi has a Master's degree in Computer Engineering since 2019 from University of Prishtina "Hasan Prishtina" in Kosovo. He works in the industry as a Software Engineer for over 10 years, mainly on PHP and MySQL stack, currently involved in tyres and rims industry. Previously, he worked for two years in banking industry as a Business Intelligence and Customer Relationship Management Specialist. His research interests are big data, data analytics and recommender systems.



Eliot Bytyçi is a Professor Assistant in the Department of Mathematics, Faculty of Mathematical and Natural Sciences, University of Prishtina "Hasan Prishtina" in Kosovo. His current research lies in the area of Data Mining, where he uses different kinds of algorithms in relation to the Semantic Web, in order to gain new insights. He is an active member of the research community in Kosovo, taking part in different projects, national and international, related to the ICT in general. As part of the University staff, he took part in creation of curricula and proposal for new study programs in Computer Science. Eliot is a board member of renowned journal on Semantic Web and has been a reviewer for some of the most renowned scientific journals in the field of Data Mining and Semantic Web.