News Recommender System based on Association Rules @ CLEF NewsREEL 2017

Christián Golian and Jaroslav Kuchař

Web Intelligence Research Group, Faculty of Information Technology,
Czech Technical University in Prague
Thákurova 9, 160 00 Prague 6, Czech Republic
{goliachr, jaroslav.kuchar}@fit.cvut.cz

Abstract. Digital editions of newspapers cause information overflow and users have problems choosing what they want to read. Systems which recommend news articles are suitable to solve such problems. Nevertheless, they face challenges unknown to the systems recommending books or movies such as a frequency of producing the new content. CLEF NewsREEL challenge enables to compare and evaluate news recommendation systems in an online and offline task focused on recommending articles to real users and tuning of algorithms respectively. This paper deals with an approach based on association rules acting as a classifier. In our approach we experimented with settings that allows to reduce the amount of rules used for the classification and increase the performance that is crucial for real recommendations. We evaluated our approach in both tasks of the CLEF NewsREEL 2017 challenge.

Keywords: recommender systems, association rules, news recommendation, CLEF NewsREEL

1 Introduction

The enormous number of available news causes information overflow resulting in users having problems choosing what they want to read. News recommendation systems should solve this problem and offer them an article or a collection of articles which they could find worth a read. Systems which recommend news articles face challenges unknown to the systems recommending books or movies. Typical example of such a challenge is coping with the frequency with which the new content is produced. While a movie or a book is released once a few months (and can be read or watched multiple times), the news are produced every minute (and read once).

The *CLEF NewsREEL* [1] challenge enables to compare and evaluate news recommendation systems both offline and online. This challenge is split into two tasks: *NewsREEL Live*[7] which uses real-time information about interactions between users and items. It is realized by redirecting a part of internet

[1] www.clef-newsreel.org

traffic to a recommender system of a participant in the challenge. *NewsREEL Replay*[6] uses historical data containing recorded information about users, items and interactions between users and items. The provided system replays recorded articles' visits and compares them with the recorded clicks. Data used in both these tasks is provided by an advertisement company *plista*[8] - they are focused on providing recommendations for news portals in Germany.

Over the years, many different approaches to news recommendation have been developed. Frequently used algorithms include recommending the most read articles or recommending the most recent articles, including their modifications e.g. most read articles in their respective categories [4]. Other versions use data processing frameworks such as the Apache Storm or Apache Flink to make the system more scalable. These approaches were described by Domann et al [3] and Ciobanu et al [1] respectively. Approaches using a collaboration filtering for the recommending of articles, where users with similar tastes are used, have been also tried. For more details see Lommatzsch et al [10]. Other experimental algorithms can recommend articles containing images which have the potential to capture the user interest. This approach is described in more in detail by Corsini et al [2].

Recommending articles using association rules was already tried by Kliegr et al [9]. Given that this approach yielded promising results we decided to do further work in this area. The advantage of the rule-based approach is the possibility to easily explain the recommendations, since rules are considered as on of the most understandable representation of models. In our algorithm we focused on the rule-based classifier CBA [11] that is also focused on reducing amount of rules using a pruning of available rules. This setting allows to significantly reduce the time during the recommendation phase and compete with other algorithms.

This paper is structured as follows: our approach based on association rules is described in Section 2. Results of its offline and online evaluation are presented in Section 3, while Section 4 concludes the paper.

2 Approach

2.1 Tasks details

A news recommender system implementing the provided interface (Open Recommendation Platform - ORP[2]) interface has to handle four types of messages: *recommendation requests*, *item updates*, *event notifications* and *error messages*.

The recommendation request indicates that the participated recommender system should return recommendations from the same domain as mentioned in the request. The recommendations are in the form of a list of identifiers and only identifiers of valid articles should be returned. Item updates signal an addition of a new item or an update including in-validations of items. Error messages are used to inform the system about possible errors that could occur such as network problems or a wrong format of the provided recommendations. Event

[2] https://orp.plista.com/documentation

notification arrives when a user visits an article (*impression*) or when the user clicks on a recommendation (*click*).

With the exception of error messages and item updates, all of aforementioned message types contain contextual information about the reader or the article being read. Examples of such features include a user location, user income, user age, article keywords or article category.

2.2 Context-Aware Item Recommender

In our approach we decided to mainly use the contextual information about the reader or the article being read. During offline evaluation, we made several experiments to explore which features would perform best. Based on these experiments measuring influence of attributes on the quality of recommendations, we selected twelve attributes for the online evaluation (*category*, *keyword*, *income*, *age*, *geo_user*, *geo_user_zip*, *weather*, *device*, *isp*, *browser*, *recommended_id* and *position* - see ORP documentation [3] for more details about attributes).

The reasoning behind the use of rules was following: If a certain number of user interactions with an item often include values of attributes repeating themselves, they may be interesting either for users sharing these attributes or for users reading articles sharing these attributes. In order to provide an example: if an article was read frequently during evening hours, it may be interesting to someone reading news late at night.

Listing 1.1: Examples of association rules created from interactions

```
{browser: 40052, geo_user_zip: 61958} => {itemId: 341743113}
            supp = 0.02, conf = 1.0
{isp: 6, category: 420949} => {itemId: 367259468}
            supp = 0.01, conf = 1.0
{browser: 16064801, device: 504182} => {itemId: 315791779}
            supp = 0.1, conf = 0.7
{} => {itemId: 322334534}
            supp = 0.01, conf = 0.01
```

Each rule is composed from a *left-hand side (LHS)*, *right-hand side (RHS)* and its described by its quality measures: *support* and *confidence*. The process of extracting of these rules (algorithms Apriori, FP-Growth, Eclat [5]) requires minimum support and confidence as its parameters. In our approach we place all available contextual features to the *LHS* and the article identifier to the *RHS*. Examples of such rules are displayed on Listing 1.1.

To prefer certain rules over other, we sort rules in the same way as in CBA according to the confidence (decreasingly), support (decreasingly) and length of the LHS of each rule (increasingly - shorter is better). Since the amount of rules returned by the standard implementation can be huge, we use the rule pruning: it removes rules that can be never used for subsequent classification, usually due to their redundancy, lower significance etc. The advantage is that the amount of rules is significantly lower and thus the classifier can provide recommendations much faster.

[3] https://orp.plista.com/documentation

Our algorithm works as follows: an article with identifier equal to `itemId` is recommended only when values of attributes contained in the left hand side of a rule are equal to values of attributes in the recommendation request. If there are more matching rules, we use all unique article identifiers as a list of recommended articles. If no recommendation was made using a matching rule, implementation of baseline algorithm provided by organizers of CLEF NewsREEL was used. However, in every domain there was a rule with an empty left hand side called the default rule. This means that every recommendation request from this domain matched it, and so at least one recommendation was always made using association rules. The baseline algorithm is thus used only in very specific situations related to addressing the *cold start problem.* The overall complexity of the algorithm is influenced by existing algorithms for the rule mining, pruning and matching of rules with recommendation requests [9].

2.3 Technical details

To be able to communicate with the platform, we decided to use the Java SDK implementation of ORP interface provided by CLEF NewsREEL organizers[4]. Main implementations of rule mining algorithms and corresponding operations are available for the programming language R[5], we thus created a set of scripts for R. Binary server `Rserve`[6] is used to provide communication between Java and R.

The incoming interactions were stored in a thread-safe collection in Java. The size of collection was limited by a constant that is configurable via a configuration parameters. It is worth noting that this number should be neither too big nor too small. While big numbers cause taking into account the history and fail in situations the users are mostly interested in newer content, small numbers do not provide sufficient data for mining reliable association rules. After n minute interval elapsed (e.g. 30 minute), thread which mined the rules was notified. This triggered the start of a rule mining process. The combination of the number of stored interactions together with the interval of updates focuses on addressing issues with the short item lifecycle. To create association rules from interactions between users and news articles *arules* library for R was used [5]. This library provides interface for representing, manipulating and analyzing transaction data and patterns. The `rCBA`[11], a *classification based on association* classifier for R was used to prune and remove redundant association rules.

Following parameters: maximum number of interactions being stored, support, confidence, minimum and maximum rule length thresholds were available via a configuration file. This enabled to experiment with these values during run time end evaluations.

[4] https://github.com/plista/orp-sdk-java
[5] https://www.r-project.org/
[6] https://rforge.net/Rserve/

Table 1: Offline evaluation - offline CTR (first dataset)

(a) Baseline algorithm

Domain	**Offline CTR**
418	0.07%
694	0.00%
1677	0.04%
3336	0.00%
13554	0.00%
35774	1.30%
all	1.04%

(b) Association rules

Domain	**Offline CTR**
418	0.14%
694	0.00%
1677	0.05%
3336	0.00%
13554	0.00%
35774	1.62%
all	1.29%

Table 2: Default values of parameters for the algorithm

Parameter	**Value**
Maximum number of items	30 000
Maximum rule length	6
Confidence	2%
Support	5
Pruning	enabled

3 Evaluation

3.1 Offline evaluation

Description In our scenario both news recommender system and the environment replaying data stream and sending recommendation requests ran on the same machine. This set-up did not enable to detect possible errors caused by any network problems. It is also important to underline that the stream of recorded data is being replayed in the offline evaluation. This means that specific situations can occur: e.g. when recommender system provides recommendation that was not provided in original interaction. This recommendation counts as an unsuccessful one, but it is also possible that the user would find it interesting if he had seen it.

Offline evaluation was done on a laptop with processor Intel Core i3 3217U with 8GB RAM and Ubuntu 16.04 as operating system.

We selected two subsets of the full dataset provided by organizers for the offline evaluation. First subset contains first 100 000 lines from the dataset and the second one contains first 500 000 lines. We also experimented with other subsets from middle regions of the dataset, but no significant changes in results were discovered. We did not experimented with the whole dataset due to the performance and time requirements of experiments for multiple runs of our algorithm with many settings. However, even this limited evaluation allows to get insights into the quality of our algorithm and also allows tuning of parameters.

Table 3: Offline evaluation - offline CTR (second datase)

(a) Baseline algorithm

Domain	Offline CTR
418	0.10%
694	0.00%
1677	0.51%
3336	0.00%
13554	0.00%
35774	1.12%
all	0.99%

(b) Association rules

Domain	Offline CTR
418	0.14%
694	0.00%
1677	0.49%
3336	0.00%
13554	0.00%
35774	1.39%
all	1.16%

Table 4: Offline evaluation - different values of parameters (first dataset)

Parameter/Domain	418	1677	35774	all
conf: 1%, supp: 0.5%	0.14%	0.45%	1.62%	1.29%
conf: 1%, supp: 1%	0.16%	0.43%	1.59%	1.27%
conf: 2%, supp: 0.5%	0.13%	0.45%	1.60%	1.28%
conf: 5%, supp: 2%	0.18%	0.46%	1.67%	1.34%
max. clicks: 5 000	0.14%	0.41%	1.65%	1.31%
max. clicks: 10 000	0.17%	0.42%	1.61%	1.28%
max. clicks: 50 000	0.12%	0.43%	1.65%	1.31%
max. clicks: 100 000	0.14%	0.40%	1.65%	1.31%
max. rule length: 2	0.16%	0.42%	1.66%	1.32%
max. rule length: 8	0.15%	0.43%	1.58%	1.26%
max. rule length: 10	0.13%	0.42%	1.60%	1.30%
max. rule length: 12	0.13%	0.43%	1.62%	1.29%
pruning: disabled	0.18%	0.53%	1.75%	1.40%

The datasets contain interactions from six different domains. It is worth noting that out of three possible types of ORP messages (*event_notification*, *recommendation_request* and *item_update*) recommendation request is represented most prominently. For our first dataset, this unusual distribution may be caused by the fact that most of the interactions come from late evening hours.

Results *Prediction accurracy* (also known as the offline click-through rate) was used as a metric to determine the overall quality of partial experiments. The prediction accuracy is defined as a proportion of the number of recommended items that have been clicked by the user and number of all recommendations. The prediction accuracy of baseline algorithm provided by organizers of CLEF NewsREEL can be seen in Table 1a. Comparing these results to results of our algorithm (Table 1b with parameter values in Table 2), it can be seen that our algorithm performed better in every domain. Results of the baseline algorithm and our algorithm for the second dataset can be seen in Table 3a and in Table 3b. Once again, our algorithm performed better than the baseline.

Table 5: Offline evaluation - different values of parameters (second dataset)

Parameter/Domain	**418**	**1677**	**35774**	**all**
conf: 1%, supp: 0.5%	0.19%	0.49%	1.41%	1.17%
conf: 1%, supp: 1%	0.21%	0.48%	1.35%	1.11%
conf: 2%, supp: 0.5%	0.17%	0.49%	1.39%	1.16%
conf: 5%, supp: 2%	0.19%	0.56%	1.34%	1.12%
max. clicks: 5 000	0.18%	0.46%	1.38%	1.15%
max. clicks: 10 000	0.18%	0.48%	1.36%	1.14%
max. clicks: 50 000	0.17%	0.47%	1.40%	1.18%
max. clicks: 100 000	0.16%	0.51%	1.41%	1.16%
max. rule length: 2	0.17%	0.49%	1.37%	1.13%
max. rule length: 8	0.16%	0.49%	1.37%	1.14%
max. rule length: 10	0.16%	0.49%	1.37%	1.14%
max. rule length: 12	0.16%	0.49%	1.37%	1.14%
pruning: disabled	0.28%	0.90%	1.75%	1.50%

We decided to experiment with settings of *rules updating frequency*, *confidence*, *support*, *maximum rule length*, *maximum number of stored interactions* and *attributes selected for rule mining*. Results of experiments to fine tune our algorithm and find values of parameters with the best prediction accuracy can be seen in Table 4 and 5 (the default values are in Table 2).

Several conclusions were drawn from these experiments. The prediction accuracy increased together with increasing of maximum length of association rule only up to a certain length. Increasing the maximum number of clicks did not increase the prediction accuracy. Disabling pruning of rules can bring better results, but at a cost of higher number of rules leading to higher number of erroneous responses and longer computing time.

Table 6: Online evaluation - results from the second test period

Team	**CTR**	**clicks**	**impressions**
WIRG	0.55%	122	21849
BL2beat	0.48%	89	18505
A	0.37%	116	31032
B	0.35%	87	24591
C	0.33%	56	16715
D	0.30%	106	34224
E	0.30%	52	17029
F	0.27%	18	6647
G	0.25%	56	22007
H	0.23%	56	24305
I	0.00%	0	0
J	0.00%	0	89
K	0.00%	0	0

3.2 Online evaluation

Description In the online evaluation, news publishers forward their users' article interactions to the recommendations provider who in turn forwards it to participant recommender systems taking part in the NewsREEL challenge. Open Recommendation Platform serves as an interface between different publishers and participants. Publishers get from ORP recommended articles within partial participants recommendation requests.

Our algorithm was deployed on a dedicated server and communication with the Open Recommendation Platform was established using the ORP protocol and the provided SDK.

Table 7: Online evaluation - results from the evaluation period

(a) CTR in time

Date	**CTR**
24.04.2017	0.45%
25.04.2017	0.35%
26.04.2017	0.35%
27.04.2017	0.39%
28.04.2017	0.43%
29.04.2017	0.42%
01.05.2017	0.40%
02.05.2017	0.40%
04.05.2017	0.40%
05.05.2017	0.40%
06.05.2017	0.40%
08.05.2017	0.40%

(b) Official results

Team	**clicks**	**impressions**	**CTR**
BL2Beat	726	193014	0.0037613852
A	58	11856	0.0048920378
B	879	244334	0.0035975345
C	817	191647	0.0042630461
D	166	72599	0.0022865329
WIRG	600	154419	0.0038855322
E	810	214406	0.0037778794
F	813	249492	0.0032586215
G	2	1146	0.0017452007
H	747	192207	0.0038864349
I	764	236322	0.0032328772
J	875	199547	0.0043849319
K	813	184052	0.0044172299
L	925	214922	0.004303887
M	1139	230896	0.0049329568
N	1268	255663	0.0049596539
O	896	130221	0.0068806107
P	6	2610	0.0022988506
Q	12	1380	0.0086956522

Results Online evaluation in CLEF NewsREEL consisted of two test periods and an evaluation period. Our algorithm took part in the second test period and in the evaluation period. The *click-through rate* was used as a metric to evaluate the quality during the online evaluation. Click-through rate is defined as a number of recommended items that the user visits divided by the total number of recommendations.

Results from the 10th of April, when second test period was in progress, are presented in Table 6. *WIRG* is the name under which our algorithm was registered. *BL2beat* is to the best of our knowledge the baseline algorithm provided by the organizers. Names of other teams were anonymized. Click-through rate from evaluation period is presented in Table 7a and official results from organizers are presented in Table 7b. Names of other teams were anonymized too.

Drop in of the click-through rate during the first days of the evaluation was caused by multiple interruptions of the traffic from ORP. We presume that these interruptions were caused by network problems and not by erroneous content of recommendations.

Using the click-through rate as the main metric, our algorithm took 13th place of 21 contestants. We were able to beat the baseline. Since we do not know any details about other participating algorithms, it is not possible to state any conclusions yet. The important fact is that our algorithm was able to handle all incoming messages, process data and provide recommendations. The recommendations are at the same time easily explainable.

4 Conclusion

In this paper our news recommender system based on association rules was examined. The main idea of the algorithm is to build a rule based classifier, use the pruning to significantly reduce the amount of rules while the recommendations can be provide faster and quality of recommendations remains comparable. The advantage is that the recommendations are explainable if needed. Our algorithm took part in both tasks of the CLEF NewsREEL 2017 challenge. The algorithm showed better results in the offline evaluation with higher rate of prediction accuracy than the provided baseline algorithm. During the offline evaluation we also experimented wit several setting of algorithm parameters. In the online evaluation, our algorithm managed to beat baseline. Our algorithm finished above baseline and above most of other contestants in the second test period. During the evaluation period, results of our algorithm were not that impressive. While still finishing slightly above baseline, its overall results were on the 13th place.

In our future work we would like to address the detected issues and limitations of our solution. We will focus on the evaluation of the approach using larger subsets of the dataset and on recommendations in other domains. One aspect that we would like to also try to overcome is the need for repetitive computation of models on the background using stream-based version of the rule mining algorithm.

Acknowledgements

This research was supported by Faculty of Informatics, Czech Technical University in Prague.

References

1. Alexandru Ciobanu and Andreas Lommatzsch. Development of a news recommender system based on apache flink. In *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016.*, pages 606–617, 2016.
2. Francesco Corsini and Martha Larson. CLEF newsreel 2016: Image based recommendation. In *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016.*, pages 618–827, 2016.
3. Jaschar Domann, Jens Meiners, Lea Helmers, and Andreas Lommatzsch. Real-time news recommendations using apache spark. In *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016.*, pages 628–641, 2016.
4. Doychin Doychev, Aonghus Lawlor, and Rachael Rafter. An analysis of recommender algorithms for online news. In *Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014*, pages 825–836, 2014.
5. Michael Hahsler, Bettina Gruen, and Kurt Hornik. arules – A computational environment for mining association rules and frequent item sets. *Journal of Statistical Software*, 14(15):1–25, October 2005.
6. Frank Hopfgartner, Torben Brodt, Jonas Seiler, Benjamin Kille, Andreas Lommatzsch, Martha Larson, Roberto Turrin, and András Serény. Benchmarking news recommendations: The CLEF newsreel use case. *SIGIR Forum*, 49(2):129–136, 2015.
7. Frank Hopfgartner, Benjamin Kille, Andreas Lommatzsch, Till Plumbaum, Torben Brodt, and Tobias Heintz. Benchmarking news recommendations in a living lab. In *CLEF'14: Proceedings of the 5th International Conference of the CLEF Initiative*, LNCS, pages 250–267. Springer Verlag, 09 2014.
8. Benjamin Kille, Frank Hopfgartner, Torben Brodt, and Tobias Heintz. The plista dataset. In *NRS'13: Proceedings of the International Workshop and Challenge on News Recommender Systems*, ICPS, pages 14–22. ACM, 10 2013.
9. Tomás Kliegr and Jaroslav Kuchar. Benchmark of rule-based classifiers in the news recommendation task. In *Proceedings of the Sixth International Conference of the CLEF Association, CLEF'15*, pages 130–141, 2015.
10. Andreas Lommatzsch and Sebastian Werner. Optimizing and evaluating stream-based news recommendation algorithms. In *Proceedings of the Sixth International Conference of the CLEF Association, CLEF'15*, LNCS, vol. 9283, pages 376–388, Heidelberg, Germany, 2015. Springer.
11. Stanislav Vojir, Vaclav Zeman, Jaroslav Kuchar, and Tomas Kliegr. Easyminer/r preview: Towards a web interface for association rule learning and classification in r. In *Proceedings of the RuleML 2015 Challenge, the Special Track on Rule-based Recommender Systems for the Web of Data, the Special Industry Track and the RuleML 2015 Doctoral Consortium hosted by the 9th International Web Rule Symposium (RuleML 2015), Berlin, Germany, August 2-5, 2015.*, 2015.