Discovering interesting patterns in MSNBC.com Anonymous Web Data using Frequent Pattern Mining Algorithms

Can Duran Unaldi and Recep Firat Cekinel, *Department of Computer Engineering, METU*

***Abstract*— Gathering information on user behavior is very crucial to learn the users' needs and to serve them better. In this study, we intended to find inherent regularities within a user's web page clicks using msnbc.com anonymous web data [7]. Within the dataset, each user's web navigations are presented and our goal is to discover interesting web page patterns and increase users' satisfaction point. For instance, we can reveal frequent paths to the target web pages and create some shortcuts to fasten the users' navigation. For these purposes, we performed association rule mining and sequential rule mining techniques to generate rules and evaluated the interestingness of these rules with some metrics.**

***Keywords*—Web Usage Mining, Association Rule Mining, Sequential Rule Mining**

I. INTRODUCTION

Web usage mining is a subfield of data mining that analyzes user activities on the web to understand their needs and to serve them better [11]. The main source of data is user logs that track the visited web pages with some additional information such as timestamp. These logs can be used to track pages that were regularly visited by the users and the companies can develop recommender systems to predict the users' behavior using these logs.

Association rules can discover the hidden relationships and find inherent regularities within data. It can be applied to web usage logs to increase user satisfaction and to increase the time spent on the company's website. As a result, personalized web pages can be designed according to user behavior logs. Additionally, the advertisements can also be personalized to attract the user's attention. Since the user activities can provide a lot about the user's interests, it can be successfully used for e-commerce. Another alternative usage of these logs can be caching. The companies can employ these logs to find frequently visited webpages and apply caching mechanisms to decrease the latency.

In this paper, we perform web usage mining on msnbc.com anonymous web data by generating association rules and sequential rules. We measure the interestingness of these rules using some metrics such as lift, interest and confidence. The paper is organized as follows: in the next section we introduce our dataset and give some statistics. In section 3, we present our methodology and the algorithms that we apply. Then, we describe the experimental setting and present the experimental results. Finally, the overview of the study and the future work are given in Section 5.

II. MSNBC.COM ANONYMOUS WEB DATASET

The data [7] comes from Internet Information Server (IIS) logs for msnbc.com and news-related portions of msn.com for the entire day of September, 28, 1999 (Pacific Standard Time). Each sequence in the dataset corresponds to page views of a user during that twenty-four hour period. Each event in the sequence corresponds to a user's request for a page. Requests are not recorded at the finest level of detail---that is, at the level of URL, but rather, they are recorded at the level of page category (as determined by a site administrator). The categories are "frontpage", "news", "tech", "local", "opinion", "on-air", "misc", "weather", "health", "living", "business", "sports", "summary", "bbs" (bulletin board service), "travel", "msn-news", and "msn-sports". Any page requests served via a caching mechanism were not recorded in the server logs and, hence, not present in the data. There are 989818 users' web page visit logs within the dataset.

III. METHODS

A. Association Rule Mining

Association rules implies that there is a strong association between a set of items I and item j such that they tend to exist together. Association rule mining consists of two subtasks:

i. Frequent Itemset Mining: It is for discovering subsets of items that occur in many transactions. Since we have page visit logs for each user, we consider each user's logs as a single transaction and consider each visited page as an item and applied frequent itemset mining to find frequently visited web page categories. The Apriori algorithm [1] is the conventional approach to mine frequent patterns but at each iteration it needs candidate generation at each pass and it is very costly and multiple scans of the database can be time-consuming. Therefore, FP-Growth algorithm [2] is developed to avoid these problems. Instead, it compresses the database into FP-tree (suffix tree) structure and divides the compressed database into conditional databases, each associated with one frequent item and mine each such database separately. It is shown that FP-growth algorithm is much faster than the Apriori algorithm since it does not require candidate generation and eliminates repeated database scans. Apache Spark provides a parallel implementation of FP-growth algorithm called PFP [3].

ii. Association Rule Generation: It is for finding all association rules that correlate the presence of one set of items with that of another set of items in the transaction database. From the business perspective, association rules are more valuable than the frequent patterns because we can discover hidden relationships such as the list of repeatedly co-purchased items using the rules. Therefore, we applied AssociationRules algorithm [4] of Apache Spark which generates specific rules that have a single item as the consequent.

B. Sequential Rule Mining

Sequential rule mining is a task of discovering rules from a sequence database [8]. A sequential rule is in the form of A → B where A and B consist of itemsets. The left-hand side of the rule is called as antecedent, and the right-hand side of the rule is known as consequent. Note that, both A and B are frequent sequences found as sequential patterns by sequential pattern mining algorithms. Also, it implies that B happens after A, so there is a temporal relationship and it indicates a strong association between A and B. Since our data inherently consists of sequence of page visits, we consider users' logs as the sequence and each page navigation as a subsequence. The sequential rule mining can be decomposed into two subtasks:

i. Frequent Sequence Mining: It aims to discover sequential patterns that frequently exist in a sequence database. In our work, we used the PrefixSpan [5] algorithm to extract frequent sequences that are provided by Apache Spark. PrefixSpan is a pattern-growth based frequent sequential pattern mining algorithm that discovers the frequent sequences recursively starting from sequences that consist of a single item and discovers larger sequences using a depth-first search. PrefixSpan algorithm builds a projected database starting from a single frequent item (called as prefix). Projected databases keep shrinking at each iteration. In contrast to the Apriori algorithm, it does not generate candidates explicitly. However, construction of projected databases can be costly.

ii. Sequential Rule Generation: The second subtask performs the extraction of interesting sequential rules by using the frequent sequences that were discovered in the first stage. The rules are generated if the confidence of that rule is above the user-defined *minconf* value. It was shown that sequential rules can be more informative than the sequential patterns [10] because rules consider the confidence and sequential patterns do not. One can also employ additional metrics to eliminate the rules with respect to some other criteria such as lift. In this work, we use the RuleGen [6] algorithm, which extracts all sequential rules having confidence above a threshold, *minconf*, that is adjusted by users. The original RuleGen algorithm generates sequential rules such as A → B where A ⊂ B. However, our implementation discovers rules in the form of A → (B − A).

IV. EXPERIMENTAL RESULTS

A. Evaluation Metrics

- *Support* measures the relative frequency of a sequence that exists in a sequence database.
- *Confidence* measures the likelihood of occurrence of B after A. Higher values of confidence imply the high association between A and B.
- *Lift* is an indicator of interestingness. Rules with high lift values (lift > 1) are considered as more interesting than rules with low lift values (lift ≤ 1). Because high lift values imply that there exists a strong dependency between the antecedent and the consequent.
- *Interest* [9] is another metric to evaluate the interestingness of a rule. A rule is considered as interesting if its interest value is greater than 0.5.

$$Support\ (A) = \frac{frequency(A)}{Total\ transactions}$$

$$Confidence\ (A \rightarrow B) = \frac{Support\ (A\ \cup B)}{Support\ (A)}$$

$$Lift\ (A \rightarrow B) = \frac{Support\ (A\ \cup B)}{Support\ (A) * Support\ (B)}$$

$$Interest\ (A \rightarrow B) = |Confidence(A \rightarrow B) - Support(B)|$$

B. Experiments

Both association rule mining and sequential rule mining models are implemented using Spark Java API. To evaluate the interestingness of the association rules and the sequential rules we have conducted experiments with the values given in Table 1 for minimum support and minimum confidence.

Table 1: Parameter Settings

min_sup	0.00005	0.0001	0.0005	0.001
min_conf	0.8	0.7	0.6	0.5

Table 2 shows the number of explored rules vs minimum support values. The following abbreviations explains the table further:

- ARM-All: Association rule mining is performed and min_conf threshold is set to 0.001
- ARM-Conf: Association rule mining is performed and min_conf threshold is set to 0.6
- SRM-All: Sequential rule mining is performed and all rules are accepted.
- SRM-Conf: Sequential rule mining is performed and min_conf threshold is set to 0.6

Table 2: Number of rules for different support thresholds

min_sup	ARM-All	ARM-Conf	SRM-All	SRM-Conf
0.00005	363 417	301 585	2 132 699	122 581
0.0001	110 902	80 984	1 427 837	86 347
0.0005	21 558	14 737	68 963	4072
0.001	3246	582	22 039	1734

According to Table 2, using the sequential rule mining we discover more rules and as we decrease the support threshold the number of rules increases. Although the sequential rule mining model generates more rules, the rules should be processed in order to achieve meaningful results. Therefore, we employed some metrics to eliminate non-interesting rules.

For each model, we have measured the interestingness of rules using interest and lift metrics. For simplicity, we only presented the top-5 interesting rules in the tables. We consider a rule R as interesting if the following criteria hold:

- R_confidence ≥ min_conf
- R_interest > 0.5
- R_lift > 1

We summarized the average statistics of each experimental setup in Table 3. For each setup, we took the average of all rules to measure these statistics. We can infer that by decreasing the support threshold, we are achieving better average confidence, average interest and average lift results.

Table 3: Average statistics for each experimental setup

Method - min_sup	Confidence	Interest	Lift
ARM-0.00005	0.8132	0.7033	13.8208
ARM-0.0001	0.7390	0.6258	10.7563
ARM-0.0005	0.7038	0.5818	8.9456
ARM-0.001	0.7606	0.5082	3.2821
SRM-0.00005	0.6848	0.5276	14.2850
SRM-0.0001	0.6754	0.5105	7.3751
SRM-0.0005	0.6716	0.4735	5.8160
SRM-0.001	0.6740	0.4756	5.5313

Top-5 rules with the highest confidence values for each setup are presented in Table 4 through Table 11. For each rule, the antecedent of the rule and consequent of it are presented with confidence, interest and lift metrics. For the consistency, we set the min_conf as 0.6 in all the following experiments. According to the tables, all of the top-5 highest confidence rules are also interesting so they can be significant from the business perspective. Moreover, as we decrease the min_sup, the confidence value for the top-5 rule increases so the association between the antecedent and consequent become more stronger.

V. CONCLUSION

In this study, we aim to discover interesting rules and patterns from msnbc.com web usage dataset. For this purpose, we have designed two models, association rule mining and sequential rule mining respectively. Although the models discover different kind of rules, we observed that both of the models are able to find interesting patterns on the dataset. For each model, we have created different setups using different evaluation metrics and discussed the success of these setups. Using different evaluation metrics, we tried to find the most interesting rules that may also be beneficial for the businesses.

Table 4: Association Rules, min-support = 0.001

Antecedent	Consequent	Confidence	Interest	Lift
{opinion, summary, misc}	{on-air}	0.9026	0.6832	4.1152
{living, misc, sports, news}	{frontpage}	0.9024	0.5860	2.8521
{health, living, business, sports}	{frontpage}	0.9	0.5835	2.8444
{living, misc, business, news}	{frontpage}	0.8967	0.5803	2.8342
{health, living, business, tech, news}	{frontpage}	0.8899	0.5735	2.8127

Table 5: Association Rules, min-support = 0.0005

Antecedent	Consequent	Confidence	Interest	Lift
{opinion, health, living, misc, weather, sports, on-air, frontpage}	{news}	1	0.8229	5.6468
{opinion, health, living, misc, weather, sports, frontpage}	{news}	1	0.8229	5.6468
{opinion, health, living, misc, weather, business}	{news}	1	0.8229	5.6468
{opinion, health, living, misc, weather, business, on-air, frontpage}	{news}	1	0.8229	5.6468
{opinion, health, living, misc, weather, business, sports}	{news}	1	0.8229	5.6469

Table 6: Association Rules, min-support = 0.0001

Antecedent	Consequent	Confidence	Interest	Lift
{opinion, summary, misc, weather, sports, on-air}	{news}	1	0.8229	5.6468
{bbs, living, weather, business}	{frontpage}	1	0.6835	3.1605
{travel, summary, health, living, misc, business, sports, tech, news, frontpage}	{on-air}	1	0.7806	4.5592
{opinion, summary, misc, weather, business, sports, local, news, on-air}	{frontpage}	1	0.6835	3.1605
{ravel, opinion, summary, health, sports, tech, news, on-air, frontpage}	{business}	1	0.8866	8.8232

Table 7: Association Rules, min-support = 0.00005

Antecedent	Consequent	Confidence	Interest	Lift
{summary, health, msn-sports, misc, msn-news, weather, business, local, tech}	{sports}	1	0.8796	8.3081
{summary, health, msn-sports, misc, msn-news, weather, business, local, tech}	{news}	1	0.8229	5.6468
{summary, health, msn-sports, misc, msn-news, weather, business, local, tech}	{on-air}	1	0.7806	4.5592
{summary, health, msn-sports, misc, msn-news, weather, business, local, tech}	{frontpage}	1	0.6835	3.1605
{bbs, summary, living, weather, business, local, on-air, frontpage}	{opinion}	1	0.9747	39.6133

Table 8: Sequential Rule, min-support = 0.001

Antecedent	Consequent	Confidence	Interest	Lift
{on-air} {misc}{misc}	{misc}	0.8812	0.7999	10.8337
{msn-news} {opinion} {opinion} [opinion}	{opinion}	0.8797	0.8544	34.8492
{on-air} {misc}	{misc}	0.8776	0.7963	10.7897
{tech} {on-air} {misc} {misc}	{misc}	0.8746	0.7933	10.7530
{weather} {misc} {local} {misc}	{local}	0.8662	0.7432	7.0439

Table 9: Sequential Rule, min-support = 0.0005

Antecedent	Consequent	Confidence	Interest	Lift
{on-air} {misc} {misc}	{misc}	0.8812	0.7999	10.8337
{msn-news} {opinion} {opinion} {opinion}	{opinion}	0.8797	0.8544	34.8492
{on-air} {misc}	{misc}	0.8776	0.7963	10.7897
{tech} {on-air} {misc} {misc}	{misc}	0.8746	0.7933	10.7530
{weather} {misc} {local} {misc}	{local}	0.8662	0.7432	7.0439

Table 10: Sequential Rule, min-support = 0.0001

Antecedent	Consequent	Confidence	Interest	Lift
{health} {local} {weather} {summary}	{news}	0.9591	0.7820	5.4163
{summary} {sports} {weather} {opinion}	{news}	0.9533	0.7762	5.3832
{local} {weather} {opinion} {summary}	{news}	0.9531	0.7760	5.3821
{summary} {local} {weather} {opinion}	{news}	0.9513	0.7742	5.3721
{local} {opinion} {weather} {summary}	{news}	0.9510	0.7739	5.3706

Table 11: Sequential Rule, min-support = 0.00005

Antecedent	Consequent	Confidence	Interest	Lift
{weather} {tech} {business} {msn-sports}	{msn-sports}	1	0.9222	12.8634
{travel} {sports} {living} {summary}	{news}	0.9852	0.8082	5.5638
{living} {misc} {tech} {bbs}	{bbs}	0.9811	0.9790	466.4467
{business} {living} {mis} {bbs}	{bbs}	0.9803	0.9782	466.095
{sports} {bbs} {news} {tech}	{news}	0.9803	0.8033	5.5361

REFERENCES

[1] Agarwal, R., & Srikant, R. (1994, September). Fast algorithms for mining association rules. In *Proc. of the 20th VLDB Conference* (pp. 487-499).

[2] Han, J., Pei, J., & Yin, Y. (2000). Mining frequent patterns without candidate generation. *ACM* sigmod *record*, *29*(2), 1-12.

[3] Li, H., Wang, Y., Zhang, D., Zhang, M., & Chang, E. Y. (2008, October). Pfp: parallel fp-growth for query recommendation. In *Proceedings of the 2008 ACM conference on Recommender systems* (pp. 107-114).

[4] (n.d.). Retrieved January 17, 2021, from https://spark.apache.org/docs/2.2.0/api/scala/index.html#org.apache.spark.mllib.fpm.AssociationRules

[5] J. Han, J. Pei, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu, "Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth," in proceedings of the 17th international conference on data engineering, pp. 215–224, Citeseer, 2001.

[6] M. J. Zaki, "Spade: An efficient algorithm for mining frequent sequences," Machine learning, vol. 42, no. 1-2, pp. 31–60, 2001.

[7] (n.d.). Retrieved January 17, 2021, from http://archive.ics.uci.edu/ml/datasets/msnbc.com+anonymous+web+data

[8] Fournier-Viger, P., Lin, J. C. W., Kiran, R. U., Koh, Y. S., & Thomas, R. (2017). A survey of sequential pattern mining. *Data Science and Pattern Recognition, 1*(1), 54-77.

[9] Rajaraman, A., & Ullman, J. D. (2011). *Mining of massive datasets*. Cambridge University Press.

[10] Fournier-Viger, P., Gueniche, T., & Tseng, V. S. (2012, December). Using partially-ordered sequential rules to generate more accurate sequence prediction. In *International Conference on Advanced Data Mining and Applications* (pp. 431-442). Springer, Berlin, Heidelberg.

[11] Srivastava, J., Cooley, R., Deshpande, M., & Tan, P. N. (2000). Web usage mining: Discovery and applications of usage patterns from web data. *Acm Sigkdd Explorations Newsletter*, 1(2), 12-23.

[12] Géry, M., & Haddad, H. (2003, November). Evaluation of web usage mining approaches for user's next request prediction. In *Proceedings of the 5th ACM international workshop on Web information and data management* (pp. 74-81).