Industry 3.5 International Symposium for Intelligent Manufacturing

Manual Book

SCAN ME

Organized by:

Artificial Intelligence for Intelligent Manufacturing Systems Research Center (AIMS), MOST, Taiwan
Industrial Engineering and Management Program (IEM), MOST, Taiwan
NTHU-TSMC Center for Manufacturing Excellence
Department of Industrial Engineering and Engineering Management, National Tsing Hua University
Center for Applied Artificial Intelligence Research, Soochow University

management consultants
Speaker: Yen-Po Liu, Management Consulting, Partner, Advisory Services Dept.
KPMG International Cooperative

17:20-17:30 Ending

September 26th (Thursday)

08:00-08:30 Registration (Engineering Building I, 1F)

08:00-09:30 Best Paper Award Finalist (Room 106, Engineering Building I, 9F)
Session Chairs: Chia-Yu Hsu, Anthony SF Chiu, Mahendrawathi ER

45 Analysis Framework for Next-Items Recommendation using Local Process Model on a Pairwise Comparison Dataset
Frans Prathama, Wenny Franciska Senjaya, Bernardo Nugroho Yahya, and Jei-Zheng Wu

14 Measuring efficiency of recycling systems with imprecise data
Cheng-Feng Hu and Hsiao-Fan Wang

34 Artificial Intelligence based Data Processing Algorithm for Video Surveillance
Linh H. Truong and Minh T. Nguyen

35 Design and Implementation of an IoT-Based Water Quality Monitoring System for Aquaculture in Mekong Delta
Luong Vinh Quoc Danh, Dang Vu Minh Dung, Tran Huu Danh and Nguyen Chi Ngon

37 A Novel Approach for Product Defect Prognostic and A Case Study
Nhat-To Huynh

08:00-09:20 Best Student Paper Award Finalist (Room 108, Engineering Building I, 9F)
Session Chairs: Che-Wei Chou, Yun-Jing Park, Biswajit Sarkar

15 Mixed Product Cost Estimation of a Cyber-Physical Production System – A Case Study
Rong-Shean Lee and Yi-Cheng Chung

19 The Development of an Expert System for the Control of Industrial Robots in Bicycle Manufacturing Welding Area
Tzu-Hsuan Huang and Taho Yang

29 Solving the Multi-Objective Software Next Release Problem using an Adaptive Metaheuristic Framework

18 Innovative data pre-processing methods to improve wafer pattern defect pattern classification
Yen- Han Lee and Ming-Chuan Chiu
Michael Manuel Smith and Yun Shiow Chen

44 Cross-Domain Human Activity Recognition in A Heterogeneous Setup with Feature Commonalities
Aria Ghora Prabono, Bernardo Nugroho Yahya and Seok-Lyong Lee

Analysis Framework for Next-Items Recommendation using Local Process Model on a Pairwise Comparison Dataset

Frans Prathama[1], Wenny Franciska Senjaya[1,3], Bernardo Nugroho Yahya[1*], and Jei-Zheng Wu[2]
[1]Department of Industrial and Management Engineering, Hankuk University of Foreign Studies, Korea
[2]Department of Business Administration, Soochow University, Taiwan
[3]Faculty of Information Technology, Maranatha Christian University, Indonesia
*bernardo@hufs.ac.kr

***Abstract*— Consumers often engage with comparison and attractive recommendations before making a decision to purchase. An intelligence approaches such as recommender system can be applied in order to provide recommendations to consumers while comparing products. While research works in recommendation system focused on transactional data with single item (i.e., market basket), there are some challenges on pairwise comparison, i.e., multiple items at the same time and order sequence of items. In addition, the next-items recommendation is a challenge on a pairwise comparison data due to its characteristics; sparsity and intransitivity. The mentioned challenges can influence consumers' decision during product search. To address the challenges, this study proposes a new framework by combining two different approaches, i.e., association rules and sequential pattern mining, to generate a recommendation on a pairwise comparison data. Using top-k association rules, the sparsity data problem could be overcome. The result from association rules is suitable for constructing the local process model, as a technique of process mining to find the frequent sequential patterns due to the intransitivity. The result of local process model gives reasonable insights as to the recommender system.**

Keywords: Association Rules, Sequential Pattern Mining, Process Mining, Recommender System.

I. INTRODUCTION

The attractiveness of recommender systems in Web 3.0 is highly associated with the hot topic of "Industry 4.0". Recommender system can be regarded as an intelligent approach to providing suggestions for the customer prior to the decision to purchase. Recommender system can support companies to improve their services to consumers. In addition, it can support consumer's decision such as choosing a relevant and preferred product or item. On the other hand, the recommender system can be employed by companies to discover the market demand, so the company can improve their service or product based on consumer's preferences. Many works on recommender system have been done in various applications such as financial, gaming, tourism, business, e-commerce [1-6].

Generally, a recommender system can be categorized into two types: Content-Based (CB) recommender system and Collaborative Filtering (CF)-based recommender system. Content-based (CB) recommender technique generates recommendation similar items based on a particular item. This technique uses items features, like user's age, gender, occupation, movie's genre, actor, description or etc. Meanwhile, Collaborative Filtering-based recommender offers recommendation based on user preferences that are generated from explicit or implicit characteristics [1]. In the literature on recommender system, they mentioned issues and challenges such as cold start problem, synonymy, shilling attacks, privacy, limited content analysis, sparsity, scalability, and context-awareness [7]. While there are a lot of works in recommendation systems, it is merely dealt with transactional data. As a matter of fact, the real-world environment enforces an attractiveness for consumer to compare products in pairs to determine which of each product is preferred, or whether or not the two products are identical. This refers to pairwise comparison. The intransitivity characteristics of pairwise comparison data, which has multiple items at the same time and disorder sequence of items, create many challenges on the recommendation and leads to the sparsity problem.

Association rule, a data mining approach, aims to discover interesting pattern from the data [5,6]. The approach has been widely used as a recommender system. However, the association rule approach neglects the sequence which is able to learn users' preferences with logical transitivity. In addition, association rule with a set of antecedent and consequence items is unable to accurately recommend the next item in a sequence. The recommender system works under sequential pattern mining techniques, works on count the *support* of the pattern. The sequential pattern mining attempts to find inter-session patterns such as the presence of a set of item followed by another item in a time-ordered set of session or episodes. Some existing works have proposed that sequential pattern mining can involve the recommender system problem, in particular, next-item recommender system [8,9]. When the data is sparse and the process is unstructured, the community proposed Local Process Model (LPM) as an extension of sequential pattern mining that allows for discovering local patterns of intransitive, including choices and loops. LPM is a recent frequent pattern mining technique that goes beyond the mining of sequential ordering relation to represent the frequent pattern using business process modeling concept. LPM techniques can generate strong relations among the activities when the sequential pattern cannot perform [10,11].

In this work, we proposed an analysis framework for next-items recommendation to handle the sparsity and intransitive of recommendation system on pairwise comparison data. The framework starts with Association Rule to prune the sparse and find the frequent items and follows with LPM to discover the proper next-items. This work aims to explore the potential and possibility of Local Process Model technique as a recommender system. This framework proceeds in four steps: data preparation process, association rule mining, LPM, and analysis. To verify the analysis framework, this study conducts an experiment on car pairwise comparison dataset. The contribution of this study is twofold; proposing an analysis framework on pairwise comparison data and verifying the potential of process mining, i.e., LPM, to enhance the quality of recommendation and overcome the sparsity problem.

The paper is organized as follows. Section 2 describes the related works in the field of recommendation system and process mining. Section 3 explains our proposed methods. Section 4 describes the dataset, experimental setting, and the result of the experiment. Section 5 concludes the paper.

II. RELATED WORKS

This section gives a brief background and explanation on existing recommender system techniques relevant to this work; association rule, sequential pattern, and recommender system for the next-item recommendation.

Recommender systems play an important role in suggesting relevant items to users and have been successfully applied in many areas, such as financial, tourism, e-commerce, games, and education. Most of the general recommender systems are represented in traditional Collaborative Filtering-based and Association rule-based recommender system, which focus on mining the relevancy between user and item [1-6]. However, most of these works provide recommender system by mining the relevancy between user and item. Despite its success, the existing works nevertheless have limitations such as sparsity and sequence problem. Many researchers have attempted to alleviate the sparsity problem such as the Collaborative Filtering-based recommender system using association retrieval technology to explore the transitive association based on users' feedback data. The proposed method computed similarity matrix through the relative distances between users' rating to alleviate the sparsity problem [12]. Other works proposed multi-level association rule which reaches a better performance than Collaborative Filtering due to the sparseness of data [13].

In recent years, many researchers focus on sequential-based recommender systems, such as next-basket or next-item recommender system [14,15]. In general, most of the sequential recommender works are based on sequential pattern mining and the applications have made successfully benefit for recommender system advancement. The basic idea of sequential pattern mining is finding the pattern relevancy between data where the values are delivered in sequence. Some early works on next-item recommendation use the approach of sequential pattern mining [14,15]. A hybrid method by mining the sequential pattern to avoid sparseness problem and by factorizing product and customers' matrices from customer purchase data provide a better recommendation (i.e., next-basket) [14]. In other work, a novel personalized sequential pattern mining-based next-item recommendation has been proposed to improve the accuracy of next-item recommendation [15]. Although the sequential pattern mining has been proposed and applied in a recommender system to improve the performance metrics, most of these methods rely on transactional data. The implementation of pairwise comparison data and intransitive property is not fully addressed.

While such existing works tried to overcome the sparsity and sequence issue, the transitivity issue in sequence data has fewer attentions. Alternatively, there are several works of recommendation system which tried to incorporate the intransitivity issue [16,17]. Konigsberg and Asherov (2014) [16] introduced a recommender system which sensitive to intransitive choice and preference reversals by computing the utility values of the items. In other work, a model approach using Bayesian probabilistic method was proposed from intransitive pairwise comparison data [17].

Although these works have taken the sparsity problem and transitive property into consideration, there has been a limited exploration for next-item recommendation system which incorporating intransitivity issue, especially using pairwise comparison dataset. This paper proposes a novel analysis framework for next-item recommender system by combining Association rule and Local Process Model approach, which is learned from intransitive user's pairwise comparison data and handling the sparseness.

III. METHODS

This section describes the proposed analysis framework for the next-items recommendation in detail. This framework adopts both data mining and process mining method. First, data preprocessing takes place prior to the analysis. The pairwise comparison data has been preprocessed by transforming the comparison data into transactional data. In addition, we filter the irrelevant data due to some issues, e.g., system errors. Second, the concept of association rule to handle the data noise (e.g., redundant data caused by a system error, dissatisfy user experience) is addressed by utilizing the Top-k Association Rule mining. Finally, a next-items recommendation which adopted the process mining technique called by Local Process Model (LPM) is described.

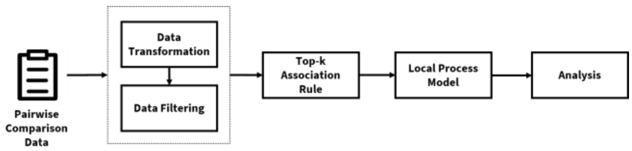


Figure 1. Analysis Framework for the next-items recommendation

A. The Sparsity and Intransitive Problem

Despite the success of a recommender system approach in many application setting, there are still some limitations, such as sparsity, scalability, and cold start. The data sparsity problem arises from the phenomenon that active users, generally, compare only a limited number of items while the number of items is extremely large (i.e., cars). In a large-scale application, both of number of consumers and products are large. The interaction between consumer and product can be represented by a matrix. When many events have been recorded, the consumers-product matrix can be extremely sparse so that there are few elements in the matrix whose value is not 0. This problem referred to the sparsity problem [12]. If the sparsity of data is high, there might be many unseen items and inactive users. Moreover, we are unable to accurately generate the recommendation.

The transitivity property is fundamental to consumers preferences. In recommender systems, users' preferences are assumed to be consistent. For example, the user where *x* is preferred to *y*, *y* is preferred to *z*, and *x* is preferred to *z*. This assumption is quite reasonable since the users behave consistently by upholding the transitivity. Some consumers do have their intransitive preference in moment-to-moment choice or daily life. A simple example of intransitive preference, acting on Monday is less preferable to acting Tuesday, which less preferable to acting on Wednesday, which less preferable to acting on Monday, because we were too late to get the report done. This condition may arise for some reasons, not only because of the user ambiguity in consumers preferences, multiple users in same account, but also a profound societal influence through economic, psychology, and benefit for personal live [15,18]. These situations are so common in pairwise comparison data are in fact not always transitive.

B. Top-k Association Rule

For the implementation of Association Rule mining, we use the Apriori algorithm. Generally, there are two steps in association rule mining: to generate frequent itemset and to generate strong association rule for each frequent itemset. The generated rules can express the patterns in a complete dataset. Association rule is generally expressed in form $X \rightarrow Y$, where X and Y are different sets of items, X is called by antecedent and Y is called by consequent.

There are three metrics to measure the strong criterion of generated rules, i.e., *support*, *confidence*, and *lift*. *Support* measures the frequency of item sets repeated in the data. *Confidence* describes how often Y (consequents) appears in the transactions that it also contains the X (antecedents). *Lift* metrics commonly used to measure the correlation between X and Y. In other words, the *confidence* measures the degree of the correlation between item sets, while *support* performs the significance of the correlation between item sets.

$$Support = \frac{Number\ of\ Transaction\ X\ and\ Y}{Total\ Number\ of\ Transaction} \quad (1)$$

$$Confidence = \frac{Number\ of\ Transaction\ X\ and\ Y}{Number\ of\ Transaction\ with\ X} \quad (2)$$

C. Next-items recommendation using Local Process Model

Process mining aims to construct a process model to represent the flow of activities. The difficulties to handle unstructured process encourage researcher in the community to develop various approaches to capture the relevant knowledge from the event log. The Local Process Model (LPM) is one of the approaches to discover frequent pattern from event logs which extend the ability of sequential process mining technique. LPM can be positioned in-between the process discovery on the one hand and frequent pattern mining on the other hand that focuses on extracting a local pattern from data expressed as Petri Nets [11]. LPM works by calculating an alignment between patterns and event log. Some metrics are used to measure the quality of the pattern (i.e., *support*, *confidence*, *language fit*, *determinism*, and *coverage*). The support relates to the number of times that the behavior that is described by the Petri net pattern is found in the event log. Then, Confidence, a pattern has high confidence when a high ratio of the events in the event log of the activities that are described in the pattern belong to instances of the pattern. It should be noted that the support and confidence of LPM are different from Association rule. The average number of enabled transitions during replay of the pattern instance on the pattern measures by determinism metrics. Coverage relates how many events in the event log are described by the pattern. The pattern will be ranked overall in those criteria [11].

Due to the sparsity problem, this approach could disregard unnecessary patterns and find the local relevant patterns with some reliable parameters such as time gaps, event distance, ranking. The use of LPM has an advantage in comparison to Association rule by finding the next-items in accordance with the local relevant patterns.

IV. RESULT

A. Data Introduction

The experiment used data from the pairwise comparison records from the cars database application (i.e., NewCarDB) developed in Taiwan *(www.newcarsdb.com)*. The dataset consists of the product (i.e., car) pairwise comparison records during 2015/1/30 to 2015/4/2 with 30,867 records were processed. The dataset includes detail information and features for various cars. All the datasets are recorded in the cloud database which becomes a treasure mine for big data analysis in the future [6].

B. Data Preparation

The data preparation follows the step before association rule and local process model performed. Data preparation consists of several activities, such as data preprocessing, data transformation, and data filtering. Data preprocessing performed by several activities, such as handling missing values, deleting outlier values, and resolving inconsistent or duplicates data from the raw dataset. Then, the remaining data is transformed into a transactional form. Transactional data comprises consumer's ID (DID), Date, Timestamp, Vehicle's Factory, Vehicle's Series, Vehicle's Style, Vehicle's Type, Vehicle's ID, Vehicle's Type, and Price. For experiment analysis, we used filtered dataset based on the exploratory data analysis result. As a result, it was identified that most events are in May 2015 with 33,704 events. Then, filtered data used to generate the experiment result.

TABLE I. DATA INTRODUCTION

Events	33,704
Cases	3,744
Activities	954
Start	2015/05/02
End	2015/05/08

C. Experiment Setting

This study conducted several scenarios of minimum support setting. The changes in minimum support will affect the number of generated rules and the number of items. There are four scenarios of minimum support setting, start from 0.008 (0.8%) to 0.01 (1%), the result for each scenario described in Table II. Based on the experiment result, the minimum support set by 0.01 (1%) and the system generates 24 association rules and 15 vehicles for the recommendation. In addition, it was not suitable if we set the minimum support lower than 0.01 (1%), the support will become a smaller number. As mentioned in the previous section, the support expressed how frequent recurred the data. When minimum support value less than 0.01 there is no pattern gathered at all.

For the implementation of Association Rule mining, we use the Apriori algorithm. As a parameter, this implementation accepts the minimum support for an item set to be identified as frequent itemset. To gather as many association rules as possible, we conducted some scenarios by setting the minimum support values. In this experiment, we used filtered dataset based on exploratory data analysis process. As a result, it was identified that the most active consumer activity (i.e., access application) is on May 2015 with 33,704 records.

TABLE II. ASSOCIATION RULE SETTING

ID	Min Support	Min Threshold	Rules Counts	Vehicle Count
1	**0.01**	**0.1**	**24**	**15**
2	0.009	0.1	32	21
3	0.0085	0.1	48	33
4	0.0080	0.1	52	36

D. Association Rule Result

Using the first scenario (minimum support = 0.01 and minimum threshold = 0.1), the generated rules are 24 with 15 vehicles for the recommendation. As we mentioned in the previous section, we only use top-k association rules. We used 10 association rules sorted by support values and the result describes in Table III. The result shows the highest support can be achieved is 0.018 with the comparison of vehicle MZ5G 1268 and vehicle MZ5T 1266.

TABLE III. TOP-10 ASSOCIATION RULES

ID	Association Rules	Supp	Conf	Lift
1	{MZ5G 1268} => {MZ5T 1266}	0.0190	0.4897	8.1479
2	{MZ5T 1266} => {MZ5G 1268}	0.0190	0.3156	8.1479
3	{MZ4T 1265} => {MZ5T 1266}	0.0152	0.3701	6.1590
4	{MZ5T 1266} => {MZ4T 1265}	0.0152	0.2533	6.1590
5	{MZ5SDA 831} => {MZ5SGA 830}	0.0150	0.3944	11.9073
6	{MZ5SGA 830} => {MZ5SDA 831}	0.0150	0.4516	11.9073
7	{MZ4T 1265} => {MZ4G 1267}	0.0144	0.3506	13.8193
8	{MZ4G 1267} => {MZ4T 1265}	0.0144	0.5684	13.8193
9	{SO16M 1387} => {SO12T 1386}	0.0134	0.6944	18.3099
10	{SO12T 1386} => {SO16M 1387}	0.0134	0.3521	18.3099

In detail, the vehicle id: MZ (i.e., MZ5G, MZ5T, and MZ4T) is the most frequent in association rules. It means these vehicles type are the most preferred by consumers. The consumer's behavior can be implicit discovered shown at generated rules. The recommendation can help the consumer to easily and effectively find their preferred car.

E. Local Process Model Result

This section addresses the LPM analysis. We performed LPM using a plugin in ProM 6.9 [10]. The discovered local process model used Markov Clustering and was sorted by Ranking. The parameter settings are as follows; minimum support = 0.1, minimum confidence = 0.4, and minimum activities in LPM = 2. The problem related to data sparsity can be mitigated either by setting the time gap constraint and number of activities in LPM filter. The time gap constraints used to specify an upper bound on the time difference between two consecutive events that fit the behavior of an LPM. While the number of activity filter allows users to filter the number of the local process from the result, this filtering process will determine the *n* number of next-item recommendation.

Table IV shows the result of LPM when we apply a time gap constraint of 20 minutes and the number of activity in LPM filter of 2 activities. For illustration, we choose four groups of local process based on the highest frequency values that each group consist of several vehicles. Based on experiment results, vehicle id 1373, 1374, and 1375 achieved the highest frequency among the others.

TABLE IV. LOCAL PROCESS MODEL RESULT

Group ID	Vehicle ID	Score	Freq	Conf
18	1373, 1374, 1375	0.55	45	0.43
22	1374, 1375, 1297	0.57	24	0.30
17	1374, 1373	0.56	21	0.272
28	241, 1343, 221, 218	0.56	16	0.65
20	221, 1343, 218	0.57	13	0.54

All vehicle ID that appears in LPM result is different from the association rule result. The flow of activities in LPMs can explicitly shows the user's behavior when comparing cars. The next-items recommendation can be clearly seen and investigated (Figure 2-5). The experiment result showed that LPM could discover local frequent patterns from pairwise comparison data in two aspects; *n* next-items recommendation and parallel (i.e., comparison) activity. Figure 2 shows that when users select vehicle id 1375 or 1373, either vehicle id 1374 is selected afterward. Meanwhile, figure 5 shows a more complex pattern, when users select vehicle 241 or 1343 or 221, the next-items recommendation is vehicle 218 based on the user's frequent pattern. The experiment result let us identified until the next three item recommendation.

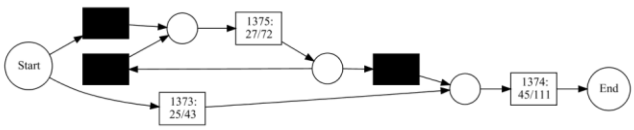


Figure 2. Local Process Model from Group ID: 18

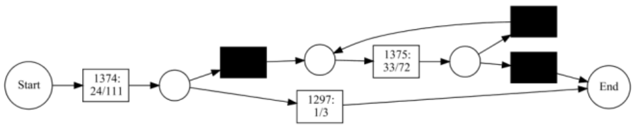


Figure 3. Local Process Model from Group ID: 22

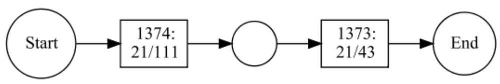


Figure 4. Local Process Model from Group ID: 17

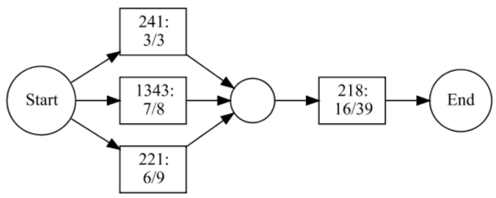
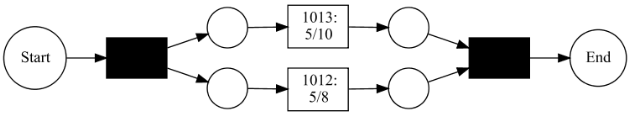


Figure 5. Local Process Model from Group ID: 28

LPM can also identify the parallel (i.e., comparison) activity. Figure 6 shows a parallel pattern on which vehicle ID 1013 and 1012 are simultaneously compared by users. The result of LPM is more beneficial than the Association Rule since the comparison process is well detected. This parallel activity occurs due to some additional parameter settings on LPM (i.e., time gap). Setting a particular value on the parameter time gap could identify the directly followed or transitive sequence in a specific time range. In this experiment, we use 20 minutes of the time gap.



Figures 6. Local Process Model (Parallel Activity)

V. CONCLUSION

This work proposed an analysis framework for the next-items recommendation on pairwise comparison data. The analysis framework started with data preprocessing, followed by handling sparsity data using association rule, and end with LPM to construct the local patterns. Association Rule had been used as a technique to prune the sparse data. Subsequently, local frequent patterns can be discovered using LPM and be used for generating *n* next-items in accordance to particular metrics. Our experiment showed that LPM could also discover and detect parallel patterns with time-gap parameter. In future, the use of hybrid recommender system techniques can be explored to enhance the quality of recommendation result.

REFERENCES

[1] Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. Recommender system application developments: a survey. *Decision Support Systems*, 74. pp. 12-32. 2015

[2] Pereira, N., & Varma, S. L. Financial Planning recommendation system using content-based collaborative and demographic filtering. *Smart Innovations in Communication and Computational Sciences*, pp. 141-151. Springer, Singapore, 2019.

[3] Lu, Z., Dou, Z., Lian, J., Xie, X., & Yang, Q. Content-based collaborative filtering for news topic recommendation. *Twenty-ninth AAAI conference on artificial intelligence*. 2015

[4] Hanke, L., & Chaimowicz, L. A recommender system for hero line-ups in MOBA games. *Thirteenth Artificial Intelligence and Interactive Digital Entertainment Conference*. 2017.

[5] Osadchiy, T., Poliakov, I., Olivier, P., Rowland, M., & Foster, E. Recommender system based on pairwise association rules. *Expert Systems with Applications*, 115, pp. 535-542. 2019.

[6] Wu, J.-Z., Liu, H.-W., & Wu, F.-L. A recommender system based on car pairwise comparisons on a mobile application using association rules. *2016 IEEE International Conference on Industrial Technology (ICIT)*, pp. 1344-1346. IEEE, 2016.

[7] Khusro, S., Ali, Z., & Ullah, I. Recommender systems: issues, challenges, and research opportunities. *Information Science and Applications (ICISA) 2016*. pp. 1179-1189. Springer, Singapore. 2016

[8] Desai, N. A., & Ganatra, A. Buying scenario and recommendation of purchase by constraint based sequential pattern mining from time stamp based sequential dataset. *Procedia Computer Science*, 45, 166-175. 2015

[9] Yap, G. E., Li, X. L., & Philip, S. Y. Effective next-items recommendation via personalized sequential pattern mining. *International conference on database systems for advanced applications*. pp. 48-64. Springer, Berlin, Heidelberg. 2012

[10] Tax, N., Sidorova, N., Haakma, R., & van der Aalst, W. M. Mining local process models. *Journal of Innovation in Digital Ecosystems*, 3(2). pp. 183-196. 2016

[11] Tax, N., Sidorova, N., van der Aalst, W. M., & Haakma, R. Local Process Model Discovery: Bringing Petri nets to the pattern mining world. *International Conference on Applications and Theory of Petri Nets and Concurrency*. pp. 374-384. Springer, Cham. 2018

[12] Chen, Y., Wu, C., Xie, M., & Guo, X. Solving the sparsity problem in recommender systems using association retrieval. *Journal of Computers*, 6(9), pp. 1896-1902. 2011

[13] Kim, C., & Kim, J. A recommendation algorithm using multi-level association rules. *Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003)*. pp. 524-527. 2003

[14] Wang, P., Chen, J., & Niu, S. CFSH: Factorizing sequential and historical purchase data for basket recommendation. *PloS one*, 13(10). 2018

[15] Crispino, M., Arjas, E., Vitelli, V., & Frigessi, A Recommendation from intransitive pairwise comparisons. *Poster Proceedings of the ACM Recommender Systems Conference*. 2016

[16] Konigsberg, A., & Asherov, R. A recommender system sensitive to intransitive choice and preference reversals. *Proceedings of the Fourth International conference on Computer Science & Information Technology*. pp. 409-419. 2014

[17] Crispino, M., Arjas, E., Vitelli, V., & Frigessi, A. Recommendation from intransitive pairwise comparisons. *Poster Proceedings of the ACM Recommender Systems Conference*. 2016

[18] Saarinen, S., Tovey, C. A., & Goldsmith, J. A model for intransitive preferences. *In Workshops at the Twenty-Eighth AAAI Conference on Artificial Intelligence*. 2014