A Comparative Analysis of Memory-based and Model-based Collaborative Filtering on the Implementation of Recommender System for E-commerce in Indonesia : A Case Study PT X

P. H. Aditya, I. Budi, Q. Munajat

[1]*Faculty of Computer Science, University of Indonesia*
Depok, Indonesia
pandhu.hutomo@ui.ac.id, indra@cs.ui.ac.id, qoribmunajat@cs.ui.ac.id

***Abstract*— The increasing growth of e-commerce industry in Indonesia motivates e-commerce sites to provide better services to its customer. One of the strategies to improves e-commerce services is by providing personal recommendation, which can be done using recommender systems. However, there is still lack of studies exploring the best technique to implement recommender systems for e-commerce in Indonesia. This study compares the performance of two implementation approaches of collaborative filtering, which are memory-based and model-based, using data sample of PT X e-commerce. The performance of each approach was evaluated using offline testing and user-based testing. The result of this study indicates that the model-based recommender system is better than memory-based recommender system in three aspects: a) the accuracy of recommendation, b) computation time, and c) the relevance of recommendation. For number of transaction less than 300,000 in database, respondents perceived that the computation time of memory-based recommender system is tolerable, even though the computational time is longer than model-based.**

Keywords*— *e-commerce; collaborative filtering; recommender system; memory-based; model-based.

I. INTRODUCTION

It is important for e-commerce sites to provide innovative features to compete with others. There are three categories of e-commerce features which are Transactional, Relational, and Social [1]. Recommender system, as a relational feature, is one of important features that need to be implemented to improve the quality of e-commerce services. Some e-commerce sites that have implemented the recommenders are amazon.com and eBay [2].

Based on the method of implementation, recommender systems generally can be divided into two, memory-based and model-based. Memory-based method performs recommendation by accessing the database directly, while model-based method uses the transaction data to create a model that can generate recommendation [3]. By accessing directly to database, memory-based method is adaptive to data changes, but requires large computational time according to the data size. As for model-based method, it has a constant computing time regardless the size of the data but not adaptive to data changes.

McCarey, Cinneide, and Kushmerick [4] conducted a study to evaluate memory-based and model-based collaborative filtering on software library. The research results show that memory-based approach is superior on two aspects which are precision and recall. Robillard and Walker [5] states that the nature of recommender system on software engineering and e-commerce domain are different. In the domain of software engineering, the recommendations are made by the task context, whereas in the domain of e-commerce, the recommendations are very dependent on the user's profile [5]. Considering the differences, it is necessary to study the implementation of recommender systems in e-commerce domain. Currently, there is still lack of studies that conduct comparative studies of model-based and memory-based recommender systems on the domain of e-commerce in Indonesia.

Motivated by the growth of e-commerce industry in Indonesia, it is important for e-commerce sites in Indonesia to implement recommender systems to improve its service quality. However, there is still lack of studies that provide the best practices to implement recommender systems within the domain of e-commerce in Indonesia. Therefore, this study wants to give contribution by exploring two approaches of recommender system implementation which are memory-based and model-based collaborative filtering on e-commerce in Indonesia. In order to perform the study, one e-commerce company in Indonesia is selected as a case study. The performance of each method is evaluated based on the computation time, accuracy, and relevance of the recommendation.

To explain the conduct of the study, the paper is structured as follows. We first explained the context of e-commerce and the related theories in recommender systems followed by the research methodology explanation. Then, we explained the implementation of the recommender systems followed by the evaluation. The results and analysis is discussed in section 5 and finally section 6 conclude the finding of this research.

II. RECOMMENDER SYSTTEMS IN E-COMMERCE

This study applied recommender systems within the domain of e-commerce. Kalakota and Whinson [6] defines e-

 978-1-5090-4629-4/16/$31.00

commerce from various perspectives, one of which is online perspective. This study uses the online-perspective which defines e-commerce as an online system that can provide product information and enable users to perform a transaction [6]. In e-commerce domain, recommender systems contribute by improving its information and service quality which are two of the six dimensions in IS success model [7]. Recommender systems improves the information and service quality by providing personalized product recommendations to users.

In the context of e-commerce, recommender system is defined as software and method that provide suggestions about products to consumers [8]. In general, recommender system consists of several components which are databases, filtering algorithm, implementation method, and evaluation method. [3]. Among those components, filtering algorithm is the main component that defines how recommender systems generate suggestions. There are several filtering algorithms that can be used in recommender systems; which are content-based filtering, demographic filtering and collaborative filtering. In recommender systems domain, collaborative filtering is one of the most successful filtering algorithm [9][10] so that it is chosen as the filtering algorithm in this study.

The basic idea of collaborative filtering is that collaborative filtering make predictions based on the opinions of users with similar characteristics [9]. In a general scenario of collaborative filtering, there are a list of m user, i.e $U = \{u_1, u_2, .., u_m\}$, a list of n items, i.e $I = \{i_1, i_2, .., i_n\}$, as well as the opinion about the item which is also known as rating [9]. Collaborative filtering can be implemented using two approaches, model-based and memory-based. Memory-based collaborative filtering uses all the data in the database to generate a prediction while the model-based collaborative filtering uses the data in the database to create a model that can then be used to generate predictions [11].

A. Memory-based Collaborative Filtering

Memory-based collaborative filtering utilizes the entire user-item data to generate predictions. The system uses statistical methods to search for a set of users who have similar transactions history to the active user. This method is also called nearest-neighbor or user-based collaborative filtering [9]. Bobadilla et al. [3] explained that there are three processes in nearest neighbor method: (1) choosing other users that are similar to a user; (2) predicting rating of the item i to a user by calculating the results of aggregating similar users, and (3) providing recommendations based on the results predicted in stage 2.

Su and Khoshgoftaar [12] stated that the advantages of memory-based collaborative filtering are easy to implement and able to accommodate the new data with ease. However, memory-based collaborative filtering has decreasing performance in data with high sparsity and have limited scalability for large datasets [12].

B. Model-based Collaborative Filtering

Model-based collaborative filtering provides recommendations by developing a model from user ratings [9]. In addition to using explicit data such as ratings, collaborative filtering can also use implicit information by observing the habits of users, such as music played, applications downloaded, websites visited, or books read [3]. To develop a model, there are two approaches that can be used, which are probability approach or rating prediction [9]. The modeling process is conducted by machine learning techniques such as classification, clustering, and rule-based approach [9]. Based on its characteristics, model-based also has its advantages and disadvantages. Su and Khoshgoftaar [12] stated that model-based approach has better predictions than memory-based. It is also capable of handling the problem of sparsity and scalability better than memory-based. However, model-based approach requires a great resource, such as time and memory, to develop the model and may lose information when using dimensionality reduction [12].

III. Research Methodology

This study uses case study and quantitative analysis to evaluate the performance of memory-based and model-based recommender systems. The research process is depicted in Fig. 1.

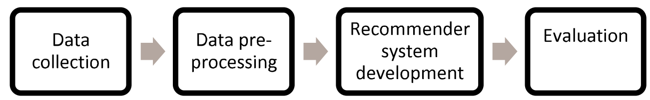

Fig. 1. Research Methodology

The data used in this research is data from PT X, one of the most popular e-commerce company in Indonesia. PT X officially launched its e-commerce on 1 March 2014. The e-commerce site of PT X offers products from eight categories, namely Fashion, Beauty / Health Babies / Kids, Home / Garden, Gadgets / Computers, Electronics, Sport / Hobby / Automotive and Service / Food. By 2016, e-commerce PT X already has 2.3 million subscribers and 4 million products [15]. The recommender system is developed using the data from PT X which is then evaluated using offline and online approach. The technique used in the evaluation process is explained in the evaluation section.

IV. Implementation

Model-based collaborative filtering uses learning techniques to create a model to generate recommendation. According to Sarwar et al. [9], learning techniques in recommender system can be categorized into two approaches: a) using a probability approach, for example, Bayesian Classifier, and b) using rating prediction of an item, for example, Singular Value Decomposition. In this study, we used the probability approach to construct the model since the dataset does not include rating information. The selected probability approach is Improved Naïve Bayes, a modified form of Naïve Bayes, developed for collaborative filtering application in e-commerce.

A. Improved Bayesian Network

To get recommendations using Naïve Bayes, we need to calculate the probability of an item will be bought by a user given the user's and item's history of transaction. In Naive

Bayes technique, the probability can be determined by using the following equation [13]:

$$p(k_1|m_1, m_2, m_3, \dots) = p(k_1) \cdot \frac{p(m_1, m_2, m_3, \dots|k_1)}{p(m_1, m_2, m_3, \dots)} \quad (1)$$

Let k_1 is an item and $m_1, m_2, m_3, \dots$ are the transaction history of an active user, so $p(k_1|m_1, m_2, m_3, \dots)$ is the probability of the active user to buy item k_1. However, in Naïve Bayes, there is an assumption that the features (item bought) are independent. Meanwhile, the nature of transaction proved otherwise; therefore, there is a bias of result. The main idea of Improved Naive Bayes is by adding a constant to the Naive Bayes equation to reduce bias. The equation for improved Naïve Bayes is:

$$p(k_1|m_1, m_2, m_3, \dots) = p(k_1) \cdot \left(\frac{p(m_1, m_2, m_3, \dots|k_1)}{p(m_1, m_2, m_3, \dots)}\right)^{\frac{c_n}{n}} \quad (2)$$

The value of n is the number of item in transaction history and c_n is a constant value of 3 determined by Wang & Tan [13] experiment. Improved techniques Naive Bayes is selected because it has been adapted to the conditions of the nature of collaborative filtering [13].

B. Nearest Neighbor

In contrast to the model-based approach that conduct the most computation, i.e. developing model, before making actual recommendations; memory-based approach directly uses transaction data in the database to make recommendations. One of the methods in memory-based approach is nearest neighbor. Using nearest neighbor, to recommend items for user *U*, the system looks for other users that have similar transaction history to user *U*. Having obtained the list of users that are similar to user *U*, the system searches for products purchased by the users similar to user *U* and recommends products that haven't been purchased by user *U,* sorted by best-selling criteria. Fig. 2 illustrates how nearest neighbor method generates recommendations.

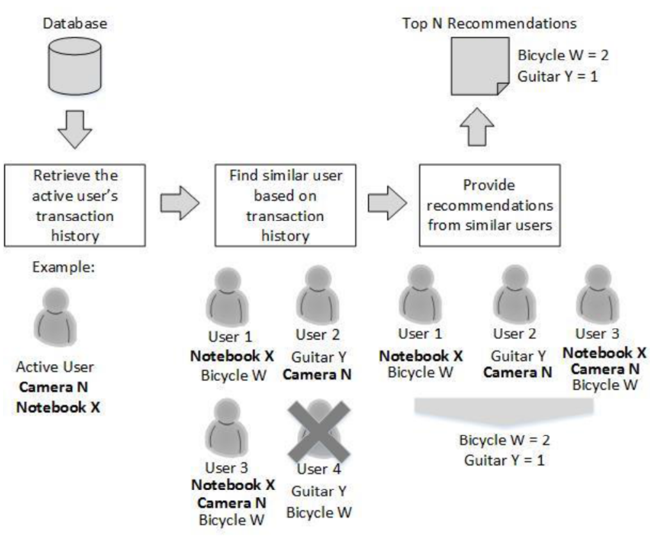

Fig. 2. Nearest Neighbor Process

In Fig. 2, an active user had two items in his transaction history, which are Camera N and Notebook X. The next process is finding users that had similar transaction history. In this case, the users are User 1 who had Notebook X, User 2 who had Camera N, and User 3 who had Notebook X and Camera X in their transaction history. User 4 is excluded since he did not purchase any item the active user had bought. The last process is getting the items from the similar users' transaction history that have not been bought by the active user. In this case, the items are Bicycle W and Guitar Y. The items are then sorted by best-selling criteria. In this case, Bicycle W is the first recommendation and Guitar Y is the second recommendation.

V. Evaluation

In recommender systems domain, evaluation process can be done using offline analysis, experiments on live users (online), and a combination of them [14]. In this study, we used both offline analysis and user-based testing to get comprehensive evaluation that can complement each evaluation method weaknesses. We conducted a case study on a real e-commerce company in Indonesia to demonstrate the implementation of recommender system on actual data of e-commerce in Indonesia.

A. Data Preprocessing

The data used in this study is the user, products, and transactions data of the PT X's e-commerce site. The product data has three attributes, namely ID, product names, and product categories which consists of three levels of categories. In this study, we used 95,468 records of product data. As for the user data, there are three attributes, namely ID, age range, and gender. In this study, we used 50,000 records of user data. Lastly, we used purchasing data (transaction data), which consists of month, year, user ID, product ID, product names, and product categories attributes. In this study, we used 290,060 records of purchasing data.

Before using the data, the data is pre-processed to remove the anomaly or outlier within the data. After discarding some irrelevant data, we obtained dataset as follows.

TABLE I. Dataset Before and After Preprocessing

Dataset	
Before Preprocessing	***After Preprocessing***
290.060 Transactions 95.468 Products 50.000 Users	290.060 Transactions 40.640 Products 26.672 Users

After that, the dataset was divided into three datasets in order to simulate the growing of data in e-commerce sites.

B. Evaluation Scenario

To perform the evaluation, dataset is divided into two sets, training and test set. Training set contains data that will be used as input to generate recommendations on memory-based approach and to create model on model-based approach. The data that is not included in training set is used as the test set to evaluate the recommendation result. The partition of the sets is

70% for the training set and 30% for the test set, as also performed by Godbole and Sarawagi [16].

In order to simulate the conditions of data growing in e-commerce site, we formed three datasets from the original dataset (namely Dataset 1, Dataset 2, Dataset 3). Dataset 1 has 33.33% of the total transaction data, Dataset 2 has 66.67% of the total transaction data and Dataset 3 has 100% of the total transaction data. The three dataset is formed based on transaction time in ascending. These three data sets simulate how the data size is growing overtime so that can be used to evaluate how the growing data size affects the performance of each recommender systems approach. The following table show the number of data for each set.

TABLE II. DATASETS FOR EXPERIMENTS

Dataset 1	Dataset 2	Dataset 3
100.016 Transactions 17.574 Products 22.329 Users 3 Avg. Transactions	200.016 Transactions 28.639 Products 25.217 Users 5 Avg. Transactions	290.060 Transactions 40.640 Products 26.672 Users 6 Avg. Transactions

C. *Offline Testing*

Precision and recall, metrics that is commonly used in the field of information retrieval, can be used to evaluate the recommendation accuracy. According to Sarwar, Karypsis, Constant, and Riedl [10], the definition of recall and precision in the context of the recommenders system is as follows:

- Recall defined as:

$$recall = \frac{\text{Size of Hit Set}}{\text{Size of Test Set}} \quad (3)$$

- Precision defined as:

$$precision = \frac{\text{Size of Hit Set}}{\text{Size of Top N Set}} \quad (4)$$

Precision and recall are often in conflict, for example, the addition of the value of N increases the recall value but reduces the precision value [10]. However, both of precision and recall are important to determine the quality of the system. Therefore, it is required a metric that can combine the precision and recall value, which is F1 metric. F1 Score can be calculated by using the following formula.

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (5)$$

D. *User-based Testing*

Online testing or live user experiments is conducted as an evaluation method for evaluating user performance, satisfaction, participation, and other measurements [14]. Online testing or user-based testing is an approach that is taken to cover the weaknesses of offline testing. In this study, user-based testing is performed to measure how relevant the products recommended by the system and to evaluate the user's tolerance toward computation time. To conduct this evaluation, we developed an online system that can be accessed by respondents. The scenario of user-based testing is as follow:

1. User access the system through a browser.
2. User fill out the basic information form (age and gender).
3. User chooses three items as transaction history
4. The system generates five recommended items using memory-based approach.
5. User chooses a number of recommended items that are considered relevant by user and determine whether the computation time can be tolerated.
6. The system generates five recommended items using model-based approach.
7. User chooses a number of recommended items that are considered relevant by user and determine whether the computation time can be tolerated
8. Repeat step 4-7 for each dataset (Dataset 1, 2, and 3)

There are 31 respondents with demographics of a) 42% female (13 people), 58% male (18 people), and b) aged between 18-24 years. All the respondent is familiar with e-commerce sites.

VI. RESULT AND ANALYSIS

A. *Offline Testing Result*

The offline testing evaluated the accuracy of the recommendation using F1 metric. The evaluation is conducted for all variety of N value (5, 8, 11, 14, 17, and 20) where N is the number of recommended item generated. The result of offline testing is displayed on Table III, as follows.

TABLE III. OFFLINE TESTING RESULT

N	Dataset 1			Dataset 2			Dataset 3		
	Me	*Mo*	*Mo+*	*Me*	*Mo*	*Mo+*	*Me*	*Mo*	*Mo+*
5	1,57	5,53	5,57	2,07	2,85	5,32	1,87	2,21	4,74
8	1,51	5,16	5,15	1,90	2,97	4,97	1,79	2,30	4,52
11	1,38	4,63	4,65	1,83	2,76	4,61	1,70	2,16	4,29
14	1,13	4,18	4,19	1,73	2,56	4,29	1,59	2,03	4,06
17	1,20	3,80	3,80	1,62	2,40	4.01	1,50	1,92	3,83
20	1,13	3,49	3,49	1,52	2,26	3,76	1,42	1,80	3,64
Avg.	**1,32**	**4,47**	**4,48**	**1,78**	**2,63**	**4,59**	**1,65**	**2,07**	**4,18**

For each dataset, there are three result of recommendation which are (1) *Me,* the result from memory-based approach; (2) *Mo,* the result form Model-based approach without updating the model; and (3) *Mo+*, the result from Model-based approach with updated model for each data change. The third result, *Mo+,* is used as a benchmark for other models. In this evaluation, the higher the F1 Score means better performance. Based on Tabel III, we can conclude that model approach has better performance than memory-based approach in term of accuracy. We can also conclude that N=5 is the optimal number of recommendations that need to be generated.

B. User-based Testing Result

The user-based testing evaluated the relevance of the recommendation according to user and the user tolerance towards computation time for each method. The relevance testing is conducted to reconfirm the accuracy (F1 Score) obtained from the offline testing.

An online system was developed as an evaluation tool for both of model-based and memory-based recommender systems. Respondent were requested to use the systems as if it were a real e-commerce sites. The systems generate 5 recommended items that need to be evaluated by respondents. Five (5) is chosen as the number of N (the number of recommendation generated) based on the offline testing result. The offline testing result shows that 5 has the highest average F1 Score, so that it is chosen as the optimal number of N.

From the user-based testing, three information are collected: (1) the computation time, as shown in Tabel IV; (2) user's tolerance toward computation time, as shown in Tabel V; and (3) the relevance of recommended items as shown in Table VI.

TABLE IV. COMPUTATION TIME

Method	Dataset 1	Dataset 2	Dataset 3	Average
Memory-based	415	789	1220	808
Model-based	54	80	97	77

All data is presented in millisecond

TABLE V. USER TOLERANCE TOWARD COMPUTATION TIME

Method	Dataset 1	Dataset 2	Dataset 3	Average
Memory-based	95,77	87,1	90,32	91,40
Model-based	93,55	87,1	83,87	88,17

All data is presented in percent

TABLE VI. RELEVANCE OF RECOMMENDED ITEMS

Method	Dataset 1	Dataset 2	Dataset 3	Average
Memory-based	18,71	18,71	14,20	17,20
Model-based	17,42	20,65	16,13	18,07

All data is presented in percent

The value in Table V shows the percentage of respondent that considered the computation time can be tolerated. It means the higher the score is better. Meanwhile, the value in Table VI shows the percentage of item that were considered relevance to the recommendation given the user transaction history. It also means that the higher the score is better. The computation time is recorded to examine how long the time to generate recommendation for each method. Together with the user tolerance data, the computation time data can be used to analyze the limit of user tolerance.

C. Analysis

The graph in Fig. 3 shows that the computation time of memory-based is longer than model-based approach. As can be seen in the graph, the data growth affects the computation time of memory-based approach. This is because memory-based approach includes all the data, including the new one, in the calculation process. Meanwhile, the increasing of computation time on model-based was not significant because the model that used on each dataset is the same model. New data didn't affect computation time significantly.

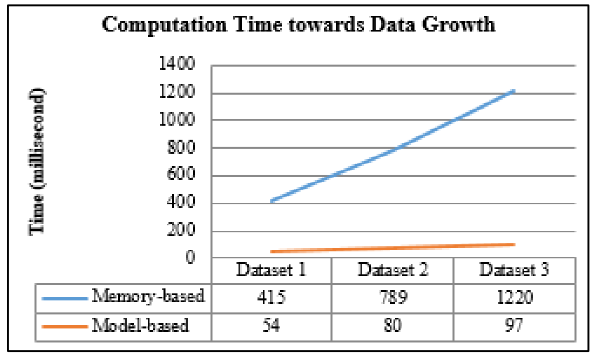

Fig. 3. Computation Time towards Data Growth

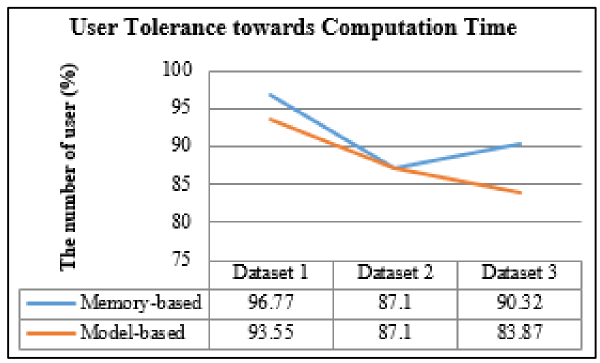

Fig. 4. User Tolerance towards Computation Time

The graph in Fig. 4 shows that in general, the user tolerance towards computation time from dataset 1 to dataset 3 is decreasing. There is an anomaly in which the user tolerance on memory-based better than model-based, despite the growth of memory-based computation time is always at the top of the model-based. This may happen because of several factors such as the internet connection, bias occured due to the order of testing or other factors. This data shows the need for further research in the perspectives of user behaviour.

The graph in Fig. 5 shows that in general, the number of relevant products perceived by user is decreasing when the number of data is increasing.

The average number of relevant products for each dataset in memory-based and model-based are 17.20% and 18.07% respectively. This indicates that model-based is better in terms of recommending products that are relevant to the user. There is an anomaly in the experiment on dataset 1 which shows that the relevance of the recommended products on memory-based is better than the model-based despite the better accuracy of model-based (see Table III). This finding indicates that there are many factors determining the user acceptance towards generated recommendations that need to be explored further.

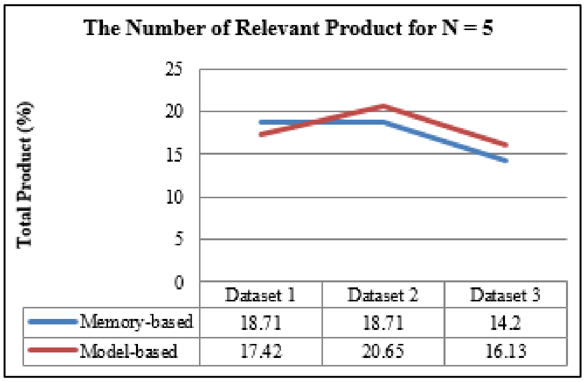

Fig. 5. The Number of Relevant Product for N = 5

VII. CONCLUSION

Based on the memory-based method and model-based collaborative filtering experiment, there are several conclusions:

a) In term of accuracy of recommendation, the offline evaluation shows that model-based accuracy is better than memory-based accuracy.
b) Based on the computation time, model-based has an average computation time 10 times faster than memory-based. This makes model-based better than memory based in terms of computational speed in recommending products..
c) Although memory-based computation time is slower than model-based, it was found that the respondents considered memory-based computing time is still as tolerable as the model-based within this number of data (less than 300,000 transaction in database).
d) Based on the average number of relevant products perceived by user, model-based is better than memory-based in generating relevant recommendation.

VIII. LIMITATIONS AND FUTURE WORK

The following are suggestions that can be used for future work:

a) This study used Improved Naïve Bayes and Nearest Neighbor techniques in the implementation of model-based and memory-based collaborative filtering. In the future studies, more techniques should be experimented to get more comprehensive analysis.
b) It is essential to conduct testing on other performance aspects of recommender system such as diversity, novelty, serendipity, and coverage.
c) The finding related to the user tolerance towards computational time and user perception towards recommended products shows an opportunity to do further research to explore the factors that influence user tolerance toward computational time and user acceptance toward recommendations in e-commerce domain.
d) It is also suggested in the future work to increase the number and the variety of respondent on user-based testing to improve the analysis of the result.

IX. ACKNOWLEDGEMENT

This study was supported by International Publication Grants for Student Thesis. The Grant title is Mobile Advertising Acceptance from Users and Gratifications Theory (UGT) Perspectives and Its Impact Towards Buying Behavior of Smartphone Users (No: 1753/UN2R12/PPM.00.00/2016).

REFERENCES

[1] R. G. Curty, & P. Zhang, “Website features that gave rise to social commerce: a historical analysis” in Electronic Commerce Research and Applications, 2013, 12(4), pp. 260-279.
[2] J. Schafer, J. Konstan, & J. Riedl, “Recommender Systems in E-Commerce” in Proceedings of the 1st ACM conference on Electronic commerce, 1999, pp. 158-166.
[3] J. Bobadilla, F. Ortega, A. Hernando, & A. Gutiérrez, “Recommender systems survey” in Knowledge-Based Systems, 2013, 46, pp. 109-132.
[4] F. McCarey, M. Cinneide, & N. Kushmerick, “A Recommender Agent for Software Libraries: An Evaluation of Memory-Based and Model-Based Collaborative Filtering” in Proceedings of the IEEE/WIC/ACM international conference on Intelligent Agent Technology, 2006, pp. 154-162.
[5] M. Robillard, & R. Walker, “An Introduction to Recommendation Systems in Software Engineering” in Recommendation Systems in Software Engineering, 2013, pp.1-11.
[6] R. Kalakota and A. Whinston, Electronic commerce. Reading, Mass.: Addison-Wesley, 1997.
[7] W. Delone, & E. McLean, “Measuring e-Commerce Success: Applying the DeLone & McLean Information Systems Success Model” in International Journal of Electronic Commerce, 2004, 9:1, pp. 31-47.
[8] F. Ricci, L. Rokach, & B. Shapira, “Introduction to Recommender System Handbook” in Recommender systems handbook, 2011, pp.1-35. Boston: Springer
[9] B. Sarwar, G. Karypsis, J. Konstan, & J. Riedl, “Item-Based Collaborative Filtering Recommendation Algorithms” in WWW10, 2001, pp. 285-295.
[10] B. Sarwar, G. Karypsis, J. Konstan, & J. Riedl, “Application of Dimensionality Reduction in Recommender System A Case Study” in ACM WebKDD 2000 Web Mining for E-Commerce Workshop, 2000.
[11] J. S. Breese, D. Heckerman, & C. Kadie, “Empirical analysis of predictive algorithms for collaborative filtering” in Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, 1998, 461(8), pp. 43–52.
[12] X. Su, & T. Khoshgoftaar, “A Survey of Collaborative Filtering Techniques” in Advances in Artificial Intelligence, 2009, pp.1-19.
[13] K. Wang, & Y. Tan, “A New Collaborative Filtering Recommendation Approach Based on Naive Bayesian Method” in Advances in Swarm Intelligence, 2011, 6729, pp. 218-227.
[14] J. L. Herlocker, J. A. Konstan, J. T. Riedl, & L. G. Terveen, “Evaluating collaborative filtering recommender systems” in ACM Transactions on Information Systems, 2004, 22 (1), pp. 5–53.
[15] Ayyubi, S. A.. “E-COMMERCE: Elevenia Klaim Memiliki 2,3 Juta Pelanggan”., 2016 accessed June 20, 2016. http://industri.bisnis.com/read/20160313/105/527621/e-commerce-elevenia-klaim-memiliki-23-juta-pelanggan
[16] S. Godbole, & S. Sarawagi, “Discriminative Methods for Multi-labeled Classification” in Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2004), 2004