Çukurova Üniversitesi Mühendislik Fakültesi Dergisi, 37(1), ss. 119-128, Mart 2022
Cukurova University Journal of the Faculty of Engineering, 37(1), pp. 119-128, March 2022

A Content-Based Product Recommendation Approach

Yıltan BİTİRİM*[1] ORCID *0000-0002-1780-2806*

[1]*Eastern Mediterranean University, Faculty of Engineering, Computer Engineering Department, Famagusta, North Cyprus*

Geliş tarihi: 21.01.2022 Kabul tarihi: 21.03.2022

Atıf şekli/ How to cite: BİTİRİM, Y., (2022). A Content-Based Product Recommendation Approach. Çukurova Üniversitesi, Mühendislik Fakültesi Dergisi, 37(1), 119-128.

Abstract

In this study, a content-based recommendation approach is proposed. It utilizes the preprocessed 245 top movie summaries of IMDB and the favorite movie genres of the user elicited with the questionnaire method and then, recommends potential products -from a product pool- that the user can “like”. For testing; a test dataset that consists of real products from Amazon.com was created, and a Web application that uses the proposed approach and leads the users to evaluate the results of this approach was designed and developed. 52 volunteered subjects attended the test. The subject examined and graded each of the 10 products displayed. Grading was based on the five-level Likert-type scale “Not at all” (0%), “Slightly” (25%), “Moderate” (50%), “Very” (75%), and “Extremely” (100%). It is possible to say that the subjects are moderately liked the products. When the product evaluations are categorized in two categories as “liked” and “disliked”, it is possible to say that the subjects liked ~78.65% of the products. This approach could be integrated into e-commerce applications like Amazon.com for recommending potential products that the user can “like”.

Keywords: Recommendation systems, Content-based, Text mining, Information filtering, Information retrieval, E-commerce

Bir İçerik-Tabanlı Ürün Öneri Yaklaşımı

Öz

Bu çalışmada, bir içerik-tabanlı öneri yaklaşımı önerilmektedir. Bu yaklaşım, IMDB’nin ön işlenmiş 245 en iyi film özetlerini ve anket yöntemiyle ortaya çıkan kullanıcının favori film türlerini kullanmakta ve daha sonra, -bir ürün havuzundan- kullanıcının “beğenebileceği” potansiyelde ürünleri önermektedir. Test için; Amazon.com’dan gerçek ürünleri içeren bir test veri seti yaratılmıştır, ve önerilen yaklaşımı kullanan ve kullanıcıları bu yaklaşımın sonuçlarını değerlendirmek için yönlendiren bir Web uygulaması tasarlanıp geliştirilmiştir. 52 gönüllü denek teste katılmıştır. Denek, görüntülenen 10 ürünün her birini ayrı ayrı incelemiş ve derecelendirmiştir. Derecelendirme, “Hiç” (%0), “Biraz” (%25), “Orta” (%50),

*Corresponding author (Sorumlu yazar): Yıltan BİTİRİM, *yiltan.bitirim@emu.edu.tr*

“Çok” (%75) ve “Son derece” (%100) olarak beş-seviyeli Likert-türü ölçüye dayalı yapılmıştır. Deneklerin ürünleri orta derecede beğendiğini söylemek mümkündür. Ürün değerlendirmeleri “beğenilen” ve “beğenilmeyen” olarak iki kategoride kategorize edildiğinde, deneklerin ürünlerin yaklaşık %78,65’ini beğendiğini söylemek mümkündür. Bu yaklaşım, kullanıcının “beğenebileceği” potansiyelde ürünler önermek için Amazon.com gibi e-ticaret uygulamalarına entegre olabilir.

Anahtar Kelimeler: Öneri sistemleri, İçerik-tabanlı, Metin madenciliği, Bilgi filterleme, Bilgi erişim, E-ticaret

1. INTRODUCTION

E-commerce applications like Amazon.com make recommendations to users from the database in order to help them to reach relevant products. However, both the academics and the companies work to improve this mechanism. Furthermore, each of these e-commerce applications contains lots of products and these products are described by people in the way that the information such as “title” and “about” could be entered with a not-well-definition. Hence, it is believed that working to “let these e-commerce applications to find out relevant products for a user from the database and then recommend them to the user” is necessary.

Recommendation systems guide the users to the items that might be of their interest [1]. The service providers as well as the users get benefit of recommendation systems [2]. Recommendation systems have been studied in various areas such as news (e.g., [3-5]), job (e.g., [6,7]), movie (e.g., [8-10]), publication (e.g., [11,12]), and music (e.g., [13,14]). Recommendation systems can be roughly categorized as content-based, collaborative filtering, knowledge-based, and hybrid [8]. Content-based recommendation systems are one of the most successful recommendation systems [1]. They use content to build a model for recommendation [15]. They describe the items that may be recommended, create a profile of the user that describes the types of items the user likes, and compare items to the user profile to determine what to recommend [16].

“An e-commerce recommendation system learns from a customer and recommends products that the customer will find most valuable among the available products” [17]. Various studies have been done on e-commerce recommendation systems such as [18-23].

In this study, we propose a content-based recommendation approach which utilizes the preprocessed 245 top movie summaries of IMDB and the favorite movie genres of the user elicited with the questionnaire method and then, recommends potential products -from a product pool- that the user can “like”. In the approach: “Like” is considered instead of “interest” since “interest” is for the things that the user is looking for himself or someone else but “like” is for the things that are only for the user; “user profile” is a group of the movie summaries which are decided based on the favorite movie genres elicited with the questionnaire method (the one-time explicit user feedback method); and furthermore, a score for every product is calculated based on the user profile and the product definition, and the first *n* products are given to the user. Our approach could be integrated into e-commerce applications like Amazon.com. The approach produces a list of potential products that the user can “like”, in order to use for recommendation. In the list, the products are ordered based on priority. This gives opportunity to decide display order of the products; besides, makes possible to get the potential product that the user can “like” most for using in the similar logic with “I’m Feeling Lucky” button of Google. In addition, this approach supports various product pools.

The rest of this paper is structured as follows: Section 2 describes the proposed approach; section 3 explains the methodology for testing the approach; the next section gives results of the test and discussion; and conclusion in brief and future work are given in the last section.

2. PROPOSED APPROACH

2.1. Main Dataset

First of all, “Top 1000” movies with user rating between 5.0 and 10 for the 19 years (release date: 01/01/2000-31/12/2018) were searched on “Advanced Title Search” of IMDb.com for each of 26 genres (shown on the search page) on 24/10/2018 and the movies in details were collected per genre. Note that no result returned from 6 genres. Afterwards, the author formed another collection by filtering the collected movies which have release date after 01/01/2009. This filtered collection consists of 20 genres and each genre has the corresponding movies in details. The genres and number of movies per genre are shown in Figure 1. There are total 797 movies; however, indeed there are 245 unique movies. Every unique movie is included by 1-7 genres. Almost all unique movies are included in more than one genre.

While searching movies per genre on “Advanced Title Search” of IMDb.com, all genres of each movie are given in the search results but, for the movies that have more than three genres, only three genres are given for each. For the movies with more than three genres, only three genres are also displayed on top of their individual movie pages in IMDb.com. All genres of this type of movies are always given in their individual movie pages’ details. For the movies with three or less genres, all genres are seen in the search results, on top of their individual movie pages, and in their individual movie pages’ details. This situation did not affect our study since the searches were done based on the genres and the movies in details were collected per genre.

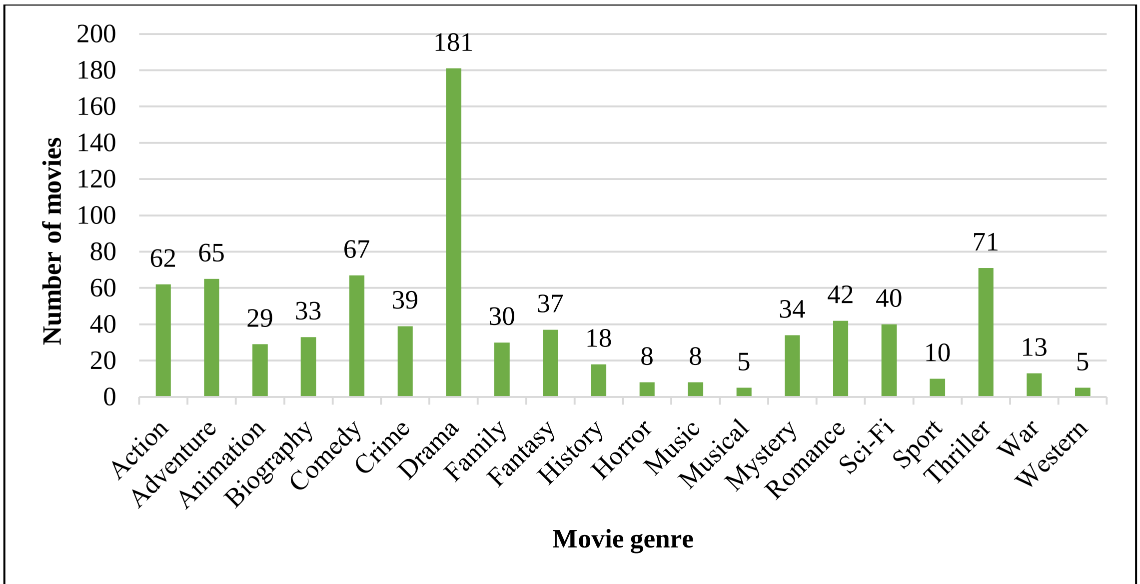

Figure 1. The genres along with the number of movies in the filtered collection

When the filtered collection was examined, it was seen that every movie was always included in its all genre groups separately. In this study, “all genres” of the movies are considered. For each of 245 unique movie, IMDb.com summary and the genres it belongs were gathered from the filtered collection into a new dataset. This dataset has 245 records.

Afterwards, the summaries in the dataset were preprocessed. In the preprocessing, various operations were done such as parenthesis removal, conversion to lowercase, stop words removal by using the English stop words list [24], private name removal, and lemmatization with the English lemmatization list [25]. (Some words have more than one lemma in the lemmatization list. When a word is this kind, the first corresponding lemma was considered. For example, when "better" word came, "good" lemma was used because "better" word has two lemmas (i.e., "good" and "well") and "good" comes before.)

When the preprocessing was finished, the preprocessed dataset named Movie Summary Dataset (MSD) was obtained. MSD consists of records for 245 unique movies with user rating 7.5 and above from 20 genres. For each movie, there is one record that consists of 1-7 genres and a preprocessed summary. Furthermore, every record has number of words between 6 and 30 in total. MSD is available at https://doi.org/10.6084/m9.figshare.19213821.v1 (file name is "MSD.txt").

2.2. Architecture

The proposed approach's architecture is given in Figure 2. The user selects her/his favorite movie genres on the questionnaire that consists of all the genres mentioned in MSD (20 genres) and submits the filled out questionnaire to Decision Module (DM). (Questionnaire method is an example of explicit user feedback method for user profiling information [26].) Our questionnaire method is one-time explicit user feedback method that elicits the favorite movie genres of the user. These genres (profiling information) are utilized by DM in order to gather every record from MSD that has at least one of these genres and then, by using these records, produce the user profile that consists of the unique movie summaries and their individual genres. So, every record in the user profile has a unique movie summary with the corresponding genres. Afterwards, DM updates the user profile; in each record, the genres are appended into the summary and the genres are removed. This is done for enriching the summaries in terms of information. Later, DM updates the user profile one more time by making all words unique in each summary. Finally, the user profile is ready-to-use and named as Double Updated User Profile (DUUP). DM also saves DUUP for using again when the corresponding user comes back.

DM gets the products stored one by one. For each of them: DM preprocesses the product's definition with various operations (such as removing special symbols, removing stop words by using the English stop words list [24], lemmatization with the English lemmatization list [25] (Some words have more than one lemma in the lemmatization list. When a word is this kind, the first corresponding lemma was considered.), and making all words unique) and then, calculates the product's score with the following equation by using the PreProcessed Definition (PPD) of the product and DUUP.

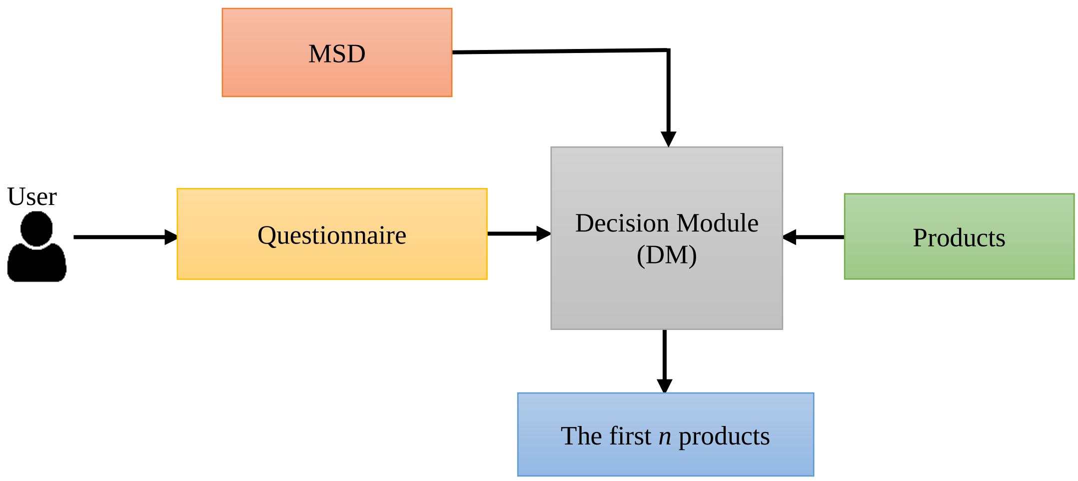

Figure 2. The architecture of the proposed approach

$$\text{score}_{\text{Pr}_i} = \sum_{k=1}^{m} \text{cosine\_similarity}(\text{PPD}_{\text{Pr}_i}, \text{Su}_k\text{\_in\_DUUP}) \quad (1)$$

where m is the number of summaries in DUUP, Pr is the abbreviation of "Product", and Su is the abbreviation of "Summary". As a result, DM obtains a score for each of the products. Then, it ranks the products in descending order based on their scores and produces the output with the first *n* products which are potential products that the user can "like". The value of *n* can be decided according to aim of using the approach. For example, if the most potential product is wanted, then 1 can be assigned to n. Furthermore, instead of giving all the products stored to DM, the products can be filtered (e.g., based on category and date range) -according to a strategy- and then, those products can be given to DM.

3. TEST METHODOLOGY

In order to be able to test the proposed approach, three steps were followed: (1) Create a test dataset that consists of real products; (2) design and develop a Web application for test that uses the proposed approach and leads the users to evaluate the results of the proposed approach; and (3) preparation of the application and use of it by some number of subjects.

Step 1: Test dataset

A test dataset consists of real products from Amazon.com was created.

19 categories (departments) were determined from Amazon.com. These categories are "Arts & Crafts", "Automotive", "Baby", "Beauty & Personal Care", "Computers", "Electronics", "Women's Fashion", "Men's Fashion", "Health & Household", "Home & Kitchen", "Industrial & Scientific", "Luggage", "Pet Supplies", "Software", "Sports & Outdoors", "Tools & Home Improvement", "Toys & Games", "Video Games", and "Deals". Except "Women's Fashion", "Men's Fashion", and "Luggage" categories, the first 24 products displayed in every category were collected. For "Luggage" category, Amazon.com displayed 12 subcategories only; therefore, each subcategory was visited and the first 2 products were collected from each. For "Men's Fashion" category, 8 subcategories were displayed and the same was done with the first 3 products. Amazon.com displayed 12 subcategories for "Women's Fashion". The subcategory "Maternity" was ignored and the first 2 products were collected from the rest of the subcategories by visiting each. As a result, 22 products were collected from "Women's Fashion" and 24 products from each of the other categories. This process was done between 18/02/2019 and 26/02/2019 and total 454 products were collected. Note 1: 2 products (one from "Sports & Outdoors" category and one from "Deals" category) was skipped since they had not "about" part, and 1 product (from "Women's Fashion") was skipped because of its picture. Note 2: In "Deals" category, 2 separate "deal of the day" (that each was for a particular group of products) came instead of 2 products. When a "deal of the day" was clicked, the group of products came. So, for each "deal of the day", the product that is the same with picture of the "deal of the day" was chosen. Note 3: For 2 products related to magazines, "Print Magazine" phrase shown along with their titles. So, this phrase was considered as a part of their titles.

Afterwards, product titles were considered and all duplicate products were removed. The remaining 388 unique products have formed the test dataset. Every product in the test dataset contains "title", "about", and "URL". The dataset is available at https://doi.org/10.6084/m9.figshare.19213821.v1 (file name is "TheTestDataset.txt").

Step 2: Web application for test

A Web application for test that uses the proposed approach and leads the users to evaluate the results of this approach was designed and developed. The application was developed with PHP. It uses NlpTools [27] for tokenization and cosine similarity calculation.

First of all, the user enters username & password pair on the first page to log in. Every username & password pair is one-time use and expires when the necessary details are saved since one-time attendance for each user is needed. Afterwards, the page on Figure 3 comes. The user chooses her/his favorite movie genres that can be one or more. For every genre, its definition, which was taken from IMDb.com, can be seen. When the user finishes choosing her/his favorite genres, she/he clicks "SUBMIT" button. Then, based on the proposed approach, the page on Figure 4 comes (on the figure, products are sample products). (Note that the combination of "title" and "about" fields of a product is considered as "product definition" of the product by the proposed approach in the application.) Here, *n* products are displayed. For each product; the user clicks the product link that contains the product title, examines the product from its own page on the Web (opened as another page), and grades the product by telling how much she/he liked it based on the five-level Likert-type scale "Not at all", "Slightly", "Moderate", "Very", and "Extremely". At last, the user clicks "SUBMIT" button. So, the necessary details are saved and the final page that contains thank and close-the-browser message is displayed. Note 1: The written instructions are given on all the application pages. Note 2: The users use the test-Web-application only one-time; therefore, DM -in the application- do not save DUUP for decreasing the computation time.

Step 3: Preparation and use of the test-Web-application

The Web application for test was published on an Apache Web server and the test dataset of real products mentioned in step 1 was entered into it. *n* is configured as 10 for the proposed approach.

Total 52 volunteered subjects were found. 39 of them were from computer engineering (CMPE), software engineering (CMSE), electrical and electronic engineering, and information systems engineering undergraduate students of Eastern Mediterranean University (EMU). The rest was from CMPE and CMSE instructors of EMU and CMPE graduate students of EMU.

In order to use the application, login is required. Some amount of username & password pairs were generated randomly and given to the application before. Each subject drew a pair of username & password from the bag that contains all generated username & password pairs and used for login. It was decided to generate username & password pairs randomly and deliver them by drawing from the bag; the reason was to let the subjects to attend the test anonymously which means without stress and so, to obtain more reliable results. Note that any personal information was not kept for the test.

In order to be sure that the subjects attended the test as required, two type of instructions were given for the test: The oral instructions (e.g., "while you are evaluating a product, please consider the product itself, not the information such as its price and stock status" and "if any trouble, please tell us") were given before the login; and the written instructions were given on all the application pages.

All subjects attended the test on 21/03/2019. Every subject logged-in, continued to use the application as it is required, and successfully finalized.

4. RESULTS AND DISCUSSION

Total 52 subjects attended the test. For each subject, 10 products were displayed from the test dataset (which has 388 Amazon.com products) based on the corresponding subject's DUUP (which is obtained from MSD according to the subject's favorite movie genres).

Every subject selected 1-14 (averagely ~7.79) of the movie genres. The most selected three genres are "Action", "Adventure", and "Comedy"; 44 subjects selected "Action", 37 subjects selected "Adventure", and 36 subjects selected "Comedy". The least selected three genres are "Music", "Musical", and "Western"; each was selected by 5 subjects only. Each of the rest was selected by 9-32 subjects.

The first 10 products with the highest scores were displayed for each subject. The product scores for every top 10 were between ~1.65% and ~2.64%. For all 52 subjects, total 29 products from the test dataset were displayed. This means ~7.47% of the test dataset was displayed. In the test dataset, products are from 19 categories. From the categories "Arts & Crafts", "Electronics", and "Health & Household", no product was displayed. From each of the other categories "Automotive", "Baby", "Beauty & Personal Care", "Computers", "Women's Fashion", "Men's Fashion", "Home & Kitchen", "Industrial & Scientific", "Luggage", "Pet Supplies", "Software", "Sports & Outdoors", "Tools & Home Improvement", "Toys & Games", "Video Games", and "Deals", 1-5 (averagely ~1.81) products were displayed. The individual display frequency range of every product displayed is from 1 to 52.

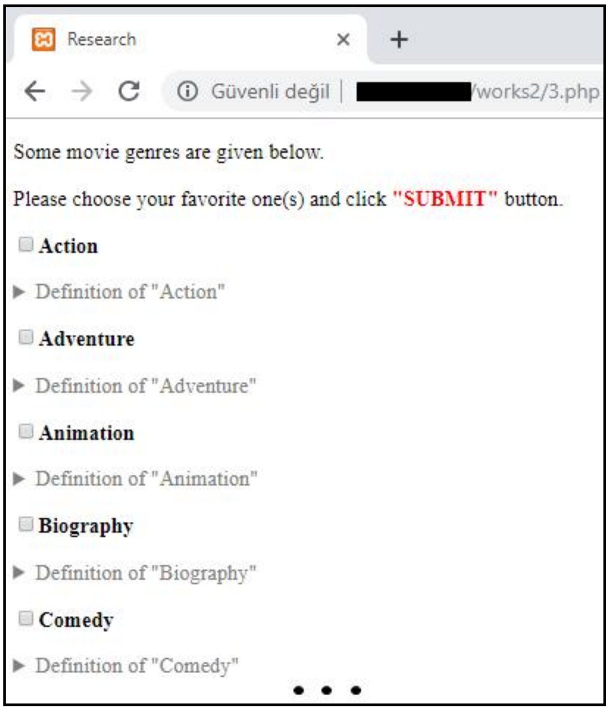

Figure 3. Genres list-page

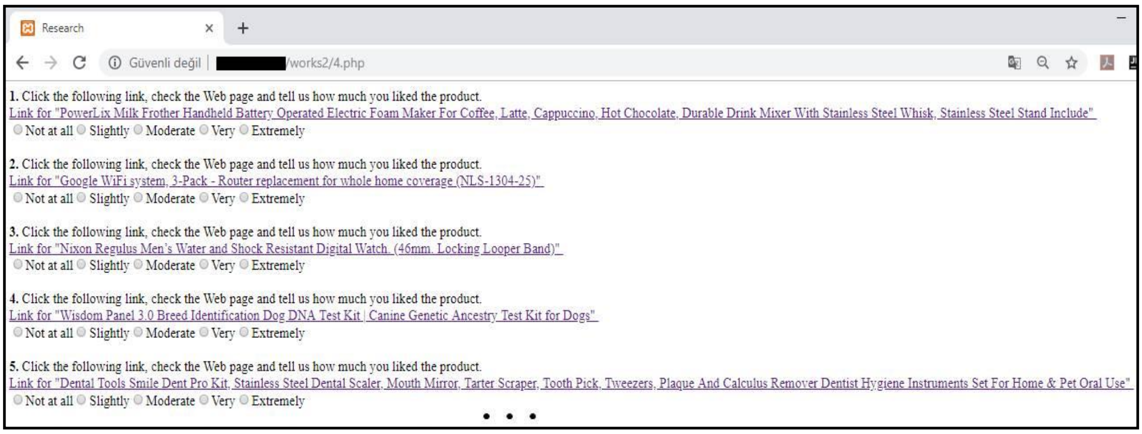

Figure 4. Products list-page with sample products

The subject examined and graded each of the 10 products displayed. Grading was based on the five-level Likert-type scale "Not at all" (0%), "Slightly" (25%), "Moderate" (50%), "Very" (75%), and "Extremely" (100%). Total 520 products were evaluated by all the subjects: 111 products were evaluated as 0% ("Not at all"), 86 products were evaluated as 25% ("Slightly"), 152 products were evaluated as 50% ("Moderate"), 113 products were evaluated as 75% ("Very"), and 58 products were evaluated as 100% ("Extremely"). For each subject, the average grade varies between 7.50% and 85%. The average product grade (or the average of the average grades of 52 subjects) is ~46.20% and so, it is possible to say that the subjects are moderately liked the products. When the product evaluations are categorized in two categories as "liked" (for "Slightly", "Moderate", "Very", and "Extremely" - 100%) and "disliked" (for "Not at all" - 0%), the followings can be said: 409 out of 520 products were liked; for each subject, the average grade varies between 10% and 100%; the average product grade (or the average of the average grades of 52 subjects) is ~78.65% and hence, it is possible to say that the subjects liked ~78.65% of the products.

For creating the test dataset, the real products were used. These products were taken from Amazon.com. Amazon.com's product descriptions are given by the sellers with the subheadings such as "title", "about", and "description". For the test dataset, only "title" and "about" fields and "URL" were collected for each product. The combination of "title" and "about" fields of a product in the test dataset was considered as "product definition" of the product by the proposed approach in the test-Web-application. However, it is observed in the test dataset that the product definition for a product might not contain fully sufficient definition. It is believed that this affected the overall success negatively. Despite of this effect, it is still reasonable since it is found as "~46.20%" and "~78.65%" (as given above).

After the test, it was encountered that two products from the test dataset had the same PPD. These are different products in the test dataset that has 388 unique products; however, they have very slight differences in their product definitions such as number of items in the package. When the product definition of each were preprocessed by DM, the things that create the difference were removed and so, each had the same PPD. These products are not from 29 products that were displayed to the subjects. If one of them would have been displayed, the first listed one in the test dataset would have been displayed in the products list-page; if two of them would have been displayed, the first listed one in the test dataset would have been displayed before the other one. Our approach can be used with various product pools and having the same PPD by more than one product is always possible. DM could easily be revised in order to act differently for this type of situations, if wanted.

Isinkaye, Folajimi, and Ojokoh [28] stated that although explicit feedback requires more effort from the user, it provides more reliable data. They stated this by considering user feedback as continuing process. However, in our approach, explicit user feedback is taken one time and therefore, reliability comes forward. Furthermore, Kramer [29] mentioned that explicit methods of eliciting user preferences caused higher acceptance rates for recommendations than implicit and opaque methods. As a result, it could be said that the questionnaire-based explicit user feedback method used supports our proposed approach positively.

Our approach recommends products from a product pool. Various product pools are possible with this approach. The product pool can have products from one or more categories and the categories can be from a variety of fields (from "baby" to "holiday tour").

5. CONCLUSION AND FUTURE WORK

In this study, it is aimed to have a content-based recommendation approach which utilizes the preprocessed 245 top movie summaries of IMDB and the favorite movie genres of the user elicited with the questionnaire method and then, recommends potential products -from a product pool- that the user can "like". This approach supports various product pools.

52 volunteered subjects attended the test. Based on the five-level Likert-type scale, it is possible to say that the subjects are moderately liked the products with ~46.20% average product grade. Based on the two-level ("liked" and "disliked"), it is ~78.65% which means that the subjects liked about 79% of the products.

According to the test results and the features of this approach, it can be said that the approach has potential and is promising for the e-commerce applications.

In order to have more success with the proposed approach: Preprocessing for MSD and preprocessing for PPD would be improved; and subtitles of the movies would be considered for MSD. Various product sets with different sufficient-definition-levels would be considered for test in order to find out the effect of product definition quality on the success of the proposed approach.

6. REFERENCES

1. Son, J., Kim, S.B., 2017. Content-Based Filtering for Recommendation Systems Using Multiattribute Networks. Expert Systems with Applications, 89, 404-412. doi: 10.1016/j.eswa.2017.08.008.

2. Pu, P., Chen, L., Hu R., 2011. A User-Centric Evaluation Framework for Recommender Systems. The 5th ACM Conference on Recommender Systems, Chicago, Illinois, USA, 157-164. doi: 10.1145/2043932.2043962.

3. Zihayat, M., Ayanso, A., Zhao, X., Davoudi, H., An, A., 2019. A Utility-Based News Recommendation System. Decision Support Systems, 117, 14-27. doi: 10.1016/j.dss.2018.12.001.

4. Ma, M., Na, S., Wang, H., Chen, C., Xu, J., 2021. The Graph-based Behavior-aware Recommendation for Interactive News. Applied Intelligence, 1573-7497. doi: 10.1007/s10489-021-02497-x.

5. Symeonidis, P., Kirjackaja, L., Zanker, M., 2021. Session-Based News Recommendations Using Simrank on Multi-modal Graphs. Expert Systems with Applications, 180, article no. 115028. doi: 10.1016/j.eswa.2021.115028.

6. Giabelli, A., Malandri, L., Mercorio, F., Mezzanzanica, M., Seveso, A., 2021. Skills2Job: A Recommender System that Encodes Job Offer Embeddings on Graph Databases. Applied Soft Computing, 101, article no. 107049. doi: 10.1016/j.asoc.2020.107049.

7. Yang, S., Korayem, M., Aljadda, K., Grainger, T., Natarajan, S., 2017. Combining Content-Based and Collaborative Filtering for Job Recommendation System: A Cost-Sensitive Statistical Relational Learning Approach. Knowledge-based Systems, 136, 37-45. doi: 10.1016/j.knosys.2017.08.017.

8. Chen, M.H., Teng, C.H., Chang, P.C., 2015. Applying Artificial Immune Systems to Collaborative Filtering for Movie Recommendation. Advanced Engineering Informatics, 29(4), 830-839. doi: 10.1016/j.aei.2015.04.005.

9. An, H., Kim, D., Lee, K., Moon, N., 2021. Movie DIRec: Drafted-input-based Recommendation System for Movies. Applied Sciences, 11(21), article no. 10412. doi: 10.3390/app112110412.

10. Reddy, S., Nalluri, S., Kunisetti, S., Ashok, S., Venkatesh, B., 2019. Content-based Movie Recommendation System Using Genre Correlation. In: Satapathy, S.C., Bhateja, V., Das, S. (eds) Smart Intelligent Computing and Applications. Smart Innovation, Systems and Technologies, 105, 391-397, Springer, Singapore. doi: 10.1007/978-981-13-1927-3_42.

11. Wang, D., Liang, Y., Xu, D., Feng, X., Guan, R., 2018. A Content-Based Recommender System for Computer Science Publications. Knowledge-Based Systems, 157, 1-9. doi: 10.1016/j.knosys.2018.05.001.

12. Kang, Y., Hou, A., Zhao, Z., Gan, D., 2021. A Hybrid Approach for Paper Recommendation. IEICE Transactions on Information and

Systems, E104D(8), 1222-1231. doi: 10.1587/transinf.2020BDP0008.

13. Sassi, I.B., Yahia, S.B., Liiv, I., 2021. MORec: At the Crossroads of Context-aware and Multi-Criteria Decision Making for Online Music Recommendation. Expert Systems with Applications, 183, article no. 115375. doi: 10.1016/j.eswa.2021.115375.

14. Cruz, A.F.T., Coronel, A.D., 2020. Towards Developing a Content-based Recommendation System for Classical Music. In: Kim, K.J., Kim, H.-Y. (eds) Information Science and Applications. Lecture Notes in Electrical Engineering, 621, 451-462, Springer, Singapore. doi: 10.1007/978-981-15-1465-4_45.

15. Hwangbo, H., Kim, Y.S., Cha, K.J., 2018. Recommendation System Development for Fashion Retail E-commerce. Electronic Commerce Research and Applications, 28, 94-101. doi: 10.1016/j.elerap.2018.01.012.

16. Pazzani, M.J., Billsus, D.J., 2007. Content-Based Recommendation Systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds) The Adaptive Web. Lecture Notes in Computer Science, 4321, 325-341, Springer, Berlin, Heidelberg. doi: 10.1007/978-3-540-72079-9_10.

17. Guo, Y., Wang, M., Li, X., 2017. Application of an Improved Apriori Algorithm in a Mobile E-commerce Recommendation System. Industrial Management & Data Systems, 117(2), 287-303. doi: 10.1108/imds-03-2016-0094.

18. Sabitha, R., Vaishnavi, S., Karthik, S., Bhavadharini, R.M., 2022. User Interaction Based Recommender System Using Machine Learning. Intelligent Automation and Soft Computing, 31(2), 1037-1049. doi: 10.32604/iasc.2022.018985.

19. Zhang, Y., 2021. The Application of E-commerce Recommendation System in Smart Cities Based on Big Data and Cloud Computing. Computer Science and Information Systems, 18(4), 1359-1378. doi: 10.2298/CSIS200917026Z.

20. Gwadabe, T.R., Liu, Y., 2022. Improving Graph Neural Network for Session-based Recommendation System via Non-sequential Interactions. Neurocomputing, 468, 111-122. doi: 10.1016/j.neucom.2021.10.034.

21. Kottage, G.N., Jayathilake, D.K., Chankuma, K.C., Ganegoda, G.U., Sandanayake, T., 2018. Preference Based Recommendation System for Apparel E-commerce Sites. IEEE/ACIS 17th International Conference on Computer and Information Science, Singapore, 122-127. doi: 10.1109/icis.2018.8466382.

22. Zhou, N., Tian, J., Li, M., 2021. Online Recommendation Based on Incremental-input Self-organizing Map. Electronic Commerce Research and Applications, 50, article no. 101096. doi: 10.1016/j.elerap.2021.101096.

23. Zheng, J., Li, Q., Liao, J., 2021. Heterogeneous Type-specific Entity Representation Learning for Recommendations in E-commerce Network. Information Processing & Management, 58(5), article no. 102629. doi: 10.1016/j.ipm.2021.102629.

24. Porter, M., Boulton, R., The English (Porter2) Stemming Algorithm: English Stop Words List (UTF-8 Encoding), Snowball, Available at http://snowballstem.org/algorithms/english/stemmer.html - Access Date: 7 December 2018.

25. Měchura, M., Lemmatization-Lists: lemmatization-en.txt, Available at https://github.com/michmech/lemmatization-lists, Access Date: 8 December 2018.

26. Adomavicius, G., Tuzhilin, A., 2005. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6), 734-749. doi:10.1109/TKDE.2005.99.

27. NlpTools, Available at http://php-nlp-tools.com, Access Date: 18 January 2019.

28. Isinkaye, F.O., Folajimi, Y.O., Ojokoh, B.A., 2015. Recommendation Systems: Principles, Methods and Evaluation. Egyptian Informatics Journal, 16(3), 261-273. doi: 10.1016/j.eij.2015.06.005.

29. Kramer, T., 2007. The Effect of Measurement Task Transparency on Preference Construction and Evaluations of Personalized Recommendations. Journal of Marketing Research, 44(2), 224-233. doi:10.1509/jmkr.44.2.224.