

*M. M. KOZULIA, V. V. SUSHKO*

## **DETERMINE RECOMMENDATION SYSTEMS TO SEARCH FOR BOOKS BY PREFERENCES OF WEB USERS**

Currently, the question of state, formation and development of the information source interaction system, the scientific interaction and users' requests in certain fields of activity remains relevant under the conditions of the development of the use of Internet services. Recommendation systems are one of the types of artificial intelligence technologies for predicting parameters and capabilities.

Due to the rapid increase in data on the Internet, it is becoming more difficult to find something really useful. And the recommendations offered by the service itself may not always correspond to the user's preferences. The relevance of the topic is to develop a personal recommendation system for searching books, which will not only reduce time and amount of unnecessary information, but also meet the user's preferences based on the analysis of their assessments and be able to provide the necessary information at the right time. All this makes resources based on referral mechanisms attractive to the user. Such a system of recommendations will be of interest to producers and sellers of books, because it is an opportunity to provide personal recommendations to customers according to their preferences.

The paper considers algorithms for providing recommender systems (collaborative and content filtering systems) and their disadvantages. Combinations of these algorithms using a hybrid algorithm are also described. It is proposed to use a method that combines several hybrids in one system and consists of two elements: switching and feature strengthening. This made it possible to avoid problems arising from the use of each of the algorithms separately.

A literature web application was developed using Python using the Django and Bootstrap frameworks, as well as SQLite databases, and a system of recommendations was implemented to provide the most accurate suggestion. During the testing of the developed software, the work of the literature service was checked, which calculates personal recommendations for users using the method of hybrid filtering. The recommendation system was tested successfully and showed high efficiency.

**Keywords:** Artificial intelligence, Recommendation system, Collaborative filtration, Content filtration, Hybrid algorithm, literary service, web application, switching, feature enhancement.

*M. M. КОЗУЛЯ, В. В. СУШКО*

## **ВИЗНАЧЕННЯ РЕКОМЕНДАЦІЙНИХ СИСТЕМ ДЛЯ ПОШУКУ КНИГ ЗА УПОДОБАННЯМ ВЕБКОРИСТУВАЧІВ**

В даний час питання про стан, формування і розвиток системи взаємодії джерел інформації, наукової взаємодії і запитів користувачів в окремих сферах діяльності залишається актуальним в умовах розвитку використання інтернет-сервісів. Рекомендаційні системи – один із видів технологій штучного інтелекту для передбачення параметрів та можливостей.

Через стрімке збільшення даних в мережі Інтернет стає складніше знайти щось справді корисне. А рекомендації, що пропонує сам сервіс, не завжди можуть відповідати вподобанням користувача. Актуальність теми полягає в тому, щоб розробити персональну рекомендаційну систему пошуку книг, що не тільки зменшить часові витрати та кількість непотрібної інформації, а й відповідатиме вподобанням користувача на основі аналізу їх оцінок та зможе надати необхідну інформацію в потрібний час. Все це робить ресурси, засновані на рекомендаційних механізмах, привабливими для користувача. Така система рекомендацій зацікавить виробників та продавців книг, бо це можливість надавати персональні рекомендації клієнтам за їх вподобаннями.

У статті розглядаються алгоритми надання рекомендаційних систем (коллаборативна та контентна системи фільтрації) і їх недоліки. Також описані комбінації цих алгоритмів з використанням гібридного алгоритму. Пропонується використовувати метод, який об'єднує кілька гібридів в одну систему і складається з двох елементів: перемикання і посилення ознак. Це дозволило уникнути проблем, що виникають при використанні кожного з алгоритмів окремо.

Було розроблено літературний вебдодаток за допомогою Python із застосуванням фреймворків Django та Bootstrap, а також баз даних SQLite, і впроваджено цю систему рекомендацій для надання найбільш точних пропозицій. У ході тестування розробленого програмного забезпечення було перевірено роботу літературного сервісу, що розраховує персональні рекомендації для користувачів, використовуючи метод гібридної фільтрації. Рекомендаційна система пройшла тестування успішно та показала високу ефективність.

**Ключові слова:** штучний інтелект, система рекомендацій, спільна фільтрація, контентна фільтрація, гібридний алгоритм, літературний сервіс, вебдодаток, перемикання, посилення ознак.

*M. M. КОЗУЛЯ, В. В. СУШКО*

## **ОПРЕДЕЛЕНИЕ РЕКОМЕНДАЦИОННЫХ СИСТЕМ ДЛЯ ПОИСКА КНИГ ПО ПРЕДПОЧТЕНИЮ ВЕБПОЛЬЗОВАТЕЛЕЙ**

В настоящее время вопрос о состоянии, формировании и развитии системы взаимодействия источников информации, научного взаимодействия и запросов пользователей в отдельных сферах деятельности остается актуальным в условиях развития использования интернет-сервисов. Рекомендательные системы – один из видов технологий искусственного интеллекта для предсказания параметров и возможностей.

Из-за стремительного увеличения данных в сети Интернет становится сложнее найти что-то действительно полезное. А рекомендации, которые предлагает сам сервис, не всегда могут соответствовать предпочтениям пользователя. Актуальность темы состоит в том, чтобы разработать персональную рекомендательную систему поиска книг, которая не только уменьшит временные затраты и количество ненужной информации, но и будет соответствовать пользовательским предпочтениям на основе анализа их оценок и сможет предоставить необходимую информацию в нужное время. Все это делает ресурсы, основанные на рекомендательных механизмах, привлекательными для пользователя. Такая система рекомендаций заинтересует производителей и продавцов книг, потому что это возможность предоставлять персональные рекомендации клиентам по их предпочтениям.

В статье рассматриваются алгоритмы предоставления рекомендательных систем (коллаборативная и контентная системы фильтраций) и их недостатки. Также описаны комбинации этих алгоритмов с использованием гибридного алгоритма. Предлагается использовать метод, объединяющий несколько гибридов в одну систему и состоящий из двух элементов: переключения и усиления признаков. Это позволило избежать проблем, возникающих при использовании каждого из алгоритмов в отдельности.

© M. M. Kozulia, V. V. Sushko, 2021

Было разработано литературное вебприложение с помощью Python с использованием фреймворков Django и Bootstrap, а также баз данных SQLite, и внедрена эта система рекомендаций для предоставления наиболее точных предложений. В ходе тестирования разработанного программного обеспечения была проверена работа литературного сервиса, рассчитывающего персональные рекомендации для пользователей, используя метод гибридной фильтрации. Рекомендательная система успешно прошла тестирование и показала высокую эффективность.

**Ключевые слова:** искусственный интеллект, система рекомендаций, общая фильтрация, контентная фильтрация, гибридный алгоритм, литературный сервис, вебприложение, переключение, усиление признаков.

**Introduction.** The beginning of the XXI century is a period of continuous increase in the number information and technology development. The World Wide Web provides everyone access to search engines, directories, various web services, thereby provides a comfortable life for each user. Many literary services provide user access to reading books online. Everyone can find the right book, read it and add to your library. Due to the rapid increase in data on the Internet, it is becoming more difficult to find something useful. However, the recommendations offered by the service itself are not always can match the user's preferences. Fortunately, information-filtering systems offer an excellent solution to this problem.

The relevance of the topic is to develop a personal a recommended book search system that will not only reduce time costs and the amount of unnecessary information, but will match the user's preferences based on the analysis of their assessments and will be able to provide the necessary information in the right time. All of this makes resources based on referral mechanisms attractive to the user. Such recommendation systems will interest producers and sellers of books, because it is an opportunity to provide personal recommendations to customers according to their preferences. Recommendations will be based on user's preferences using hybrid systems, which is what it is novelty of this work.

Thus, the purpose of the study is to analyze and determine the recommendation systems and the creation of a literary web application based on the combination of recommendation systems – collaborative and content filtration, which is for the first time for literary services.

According to the work aim it is necessary to solve the following tasks:

- description of the recommendation system and ensuring its effectiveness;
- selection and modification of the method for developing a recommendation system based on the analysis of literature sources;
- implementation of the selected recommendation system for program realization;
- development of literary service software.

**Literature Review.** Recommendation systems have been introduced to meet the needs of the user and save searching time for the literature he needs [1].

In the early 90s, the first works describing recommendation systems [2] appeared – collaborative [3, 4, 5] and content [6, 7] filtration systems.

The author [8] described in his work the methods of calculating recommendations for content and collaborative filtering, as well as named their advantages and disadvantages. And Amir Salihefendic in work [9] raises the issue of ranking objects, which partially avoids the problem of cold start.

Based on research, researchers [10, 11, 12] concluded that collaborative filtering is much more satisfying to the needs of users than content.

Recommendation systems (RS) are used both in the commercial [13] sphere and in the entertainment [14]. The study of RS in the music field was started in 2010 by [6]. Later, works were written [15, 16], which collected a description of various concepts and methodologies for such subject areas as data analysis, decision-making systems, news and marketing.

In [17], the authors reviewed 26 articles and found that in most cases, the recommendation systems use the Baew approach or the tree-based approach for movies, music, news, and products for web stores.

Xiang Li [18] proposed a multidimensional context-sensitive RS method to improve the random forest algorithm. Xibin Wang [19] uses a method to determine user preferences based on their interests in the field of films. And Mladen Marovic [20] compares methods for predicting movie ratings for users.

In the article [21] was investigated that to avoid problems of content and collaborative filtering is used their combination – hybrid recommendation systems. The first hybrid recommendation system combined two methods of recommendations to achieve excellent performance with a slight deficit of any individual [22].

In the work [21] are described different types of hybrid systems. The [23] considers hybrid systems based on probabilistic methods, such as neural networks, Bayesian networks, clustering, hidden features (eg, SVD), genetic algorithms.

Research on recommendation systems has been developing rapidly in recent years, which confirms their relevance and the need for further research. The use of hybrid approaches is not limited to the use of a single method of combining algorithms. If necessary, it can be used several hybrids in one system. And the area of reference systems of literature is insufficiently studied. Therefore, in this paper is used several hybrids in one system to predict the books rating for users.

The main problem in literary recommendation services is "cold start" – after registration, the user is provided with incorrect or no books for familiarization. In this article, it is recommended to come to a combination of recommendation systems by switching and enhancing features that will solve the problem of "cold start" and provide more accurate recommendations.

**Methods.** To predict the parameters and capabilities is used artificial intelligence technologies: recommendation systems, expert systems and decision support systems [24].

Expert systems are narrowly focused: they provide recommendations, analyze, classify and solve problems through the examination of a human specialist [25].

Decision Support System (DSS) is an interactive computer automated system (software package) designed to help and support various human activities in making decisions about solving structured or unstructured problems. The use of DSS provides a thorough and objective analysis of the subject area in decision-making in difficult conditions [26].

A recommendation system (RS) is a subclass of an information filtering system that builds a rating list of objects (movies, music, books, news, websites) that a user can prefer. The recommendation system generates a list individually for each user of the system based on a preliminary analysis of the user profile: gender, age, books reviewed, grades, etc. [25].

Based on the analysis of the collected sources of information, conclusions were made about the advantages and disadvantages of existing technologies and it was decided that recommendation systems are best suited for the task of researching model tools for determining the preferences of web users.

RS is a replacement for search algorithms, as RS finds objects based on the analysis of previous data, at a time when another system may not even offer the right one.

There are the following main approaches in creating RS (fig. 1) [27]:

- 1) collaborative filtering;
- 2) content-based filtering;
- 3) hybrid filtering.

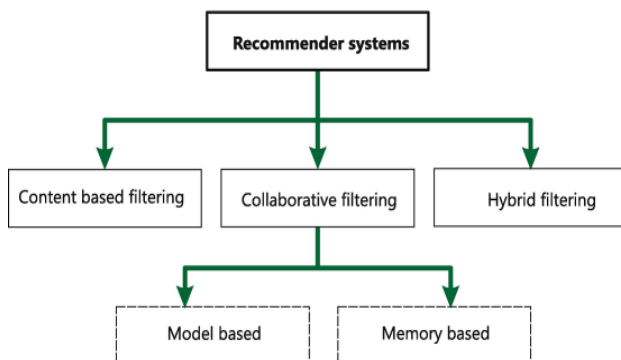


Fig. 1. Basic approaches to create recommendation systems

Collaborative filtering (CF) is a method of recommendation that analyzes only the reaction of users to objects: the ratings that users make to objects. Estimates can be either explicit (the user explicitly indicates how many "stars" he evaluates the object) or implicit (for example, the number of openings per book). The more estimates, the more accurate the recommendations will be. This helps users filter objects. Therefore, this method is also called joint filtering [28].

Content filtering or content filtering is based on information about system elements that the user has already evaluated in the past. This method analyzes data about the properties of objects, such as genre, author, publisher, and so on. The system can also use data on views and evaluations of items. The content of this method is that elements with similar content, users provide similar benefits [10].

Recommended systems, which are based on collaborative filtering, in contrast to content-based filtering, analyze user data for their work, not system elements. Each user is assigned a group of users with similar preferences. Based on this, it is hypothesized that users who have rated some objects in the past will evaluate other objects in the future in the same way [29].

One of the main problems of RS is the Cold-start Problem (CSP). It occurs when new elements appear in the system – either new users (User Cold-Start), whose preference history is empty, or new objects (Item Cold-Start), which do not yet have ratings and / or a set of features [21].

No less important problem of the recommendation system is the problem of the bubble filter. Classic recommendation systems offer users objects based only on their previous preferences. As a result, the user finds himself in an information environment in which he observes only a limited number of objects of the same type. The next problem is the considerable amount of information that complicates the situation. For example, some people are predictable and others are unpredictable, which can lead to inaccuracies and ineffective results in the recommendation system. Another problem can be considered scalability: the more data, the more difficult it is to find an accurate result [21].

Analyzing all the considered literature, it was decided that none of the methods is effective alone for creating recommendation system for program realization of literary service software.

To use the strengths of different methods, as well as to avoid the shortcomings of each of the algorithms, a hybrid RS was created (fig. 2).

Hybrid recommendation systems include:

1) Weighted: each of the algorithms is assigned a weighting factor, then a linear combination of ratings is calculated for each of the objects. For example, collaborative and content filtering types have been combined. In this case, a list of recommendations will be provided, which will list the books that were relatively highly rated by both methods.

2) Switching: choosing one option among the components of the recommendations depending on the circumstances. This approach allows, for example, to solve the problem of "cold start", which is one of the weaknesses of the method of collaborative filtration. For a "cold user" whose benefits are unknown to us, we can use to recommend the most popular documents or prepare a "universal" set [30].

3) Mixed: simultaneous use of the recommendation of different algorithms. Suppose we have several systems of recommendations that we would like to combine. In this case, you can set the mixing rules, and then show the user all the recommendations, alternating them. This method is suitable when the sources of recommendations for us act as black boxes. It does not require the objects rating calculation [31].

4) Cascade: The recommendations are given clear priority, with the lowest priorities breaking the links in winning the higher ones [31]. For example, using content filtering, we weed out books that don't interest the user at

all. And then, using collaborative filtering, we form a list of recommendations from the remaining books.

5) Feature combination: adds characteristics of the second to the first method. For example, we obtain recommendations based on contextual filtering, using such characteristics as title, author, genre, and supplement the list of characteristics through collaborative filtering.

6) Feature augmentation. This approach is similar to the previous one. In it is possible to allocate the basic and additional algorithms. But, unlike combining features, we do not add new features, but complement the values of existing ones [33]. For example, we will take collaborative filtering as a basis, and recommendations on the basis of content as an additional algorithm. This helps to avoid the problem of sparseness.

Hybridity is the combination of the results of two different approaches – content-based filtering and collaborative filtering. This can, to some extent, avoid the disadvantages of the above approaches. There are several basic options for combining filtering approaches [32]:

– *Inclusion of characteristics of elements in collaborative filtering.* The recommendation system primarily uses collaborative filtering, including some data on the elements used to find the degree of similarity of users. Benefits: increasing the relevance of recommendations; avoiding the problem of "cold start" and sparse data; taking into account the preferences of the user in the direction of the literature.

– *Inclusion of user characteristics in content filtering.* The most well-known method for implementing this approach is to reduce the dimensionality of profiles that use content filtering. One such method uses latent semantic analysis (LSA) to create collaborative user profile content. This approach is much more effective, as the recommendations are built not only due to the similarity of the elements [32].

– *Construction of a single model based on collaborative filtering and content filtering.* This approach is the most popular in the use of a hybrid system of recommendations. The essence of the algorithm is that the PC uses collaborative and content filtering simultaneously. It also helps to avoid the problem of "cold start" and sparse data and allows you to recommend elements not only for similar tastes, but also for your favorite genre or author.

The use of hybrid approaches is not limited to the use of a single method of combining algorithms. If necessary, it can be used several hybrids in one system. The proposed architecture of the hybrid RS is presented in fig. 2 [33].

There are two elements of a hybrid recommendation system:

1) Switching: for a new user, the system recommends recommendations for popularity, because it does not know its benefits and tastes.

2) Feature strengthening: if the user has already used the given literary service and has exposed estimations, the system forms for it recommendations, using the basic algorithm – collaborative filtering. But in order for the user to receive not only recommendations on similar grounds with other users, they use an additional algorithm

– recommendations on the content that will provide the user with recommendations and on the basis of the elements evaluated by him in the past.

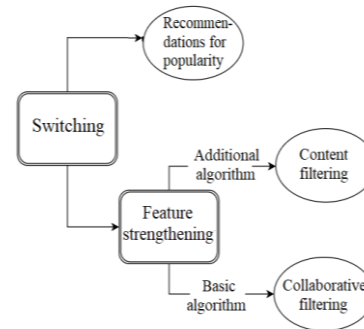


Fig. 2. Architecture of the recommendation system

Thus, the problems of "cold start" and sparse data for a new user, is solved by switching. As an algorithm for the new user, it was no coincidence that the recommendations on popularity were chosen: no additional information is required to calculate them, it is enough to evaluate the data on orders that are already available.

Each method has a separate function that builds the rating of books based on the deduction of the recommended weight and degree of proximity. Thus, we have the following algorithms for applying the hybrid system of recommendations:

Content filtering algorithm:

1) For each element, similar in characteristics (genre, year, author, etc.), the calculation of weights is performed using a measure of similarity:

$$\begin{aligned} \text{sim}(u, U') &= \cos(\vec{u}, \vec{U}') = \frac{\vec{u} \cdot \vec{U}'}{\|\vec{u}\| \cdot \|\vec{U}'\|} = \\ &= \frac{\sum_{s \in S_{u, U'}} r_{us} r_{U's}}{\sqrt{\sum_{s \in S_{u, U'}} r_{us}^2} \sqrt{\sum_{s \in S_{u, U'}} r_{U's}^2}} \end{aligned} \quad (1)$$

where  $r_{us}$  – the estimated rating of the  $s$  object for the  $u$  user;

$U'$  – a set of users who have rated the  $s$  object and belong to the same group with the  $u$  user;

$k$  – parameter for normalization, which is calculated

$$\text{for the formula } k = \frac{1}{\sum_{u \in U'} \text{sim}[u, u']}$$

2) Next, from the list of objects, a group of elements is formed that are most similar to the object calculated on the basis of weights.

3) Finally, the projected rating of objects is formed using the method of similar elements evaluation:

$$\begin{aligned} h(u, s) &= \cos(\vec{w}_u, \vec{w}_s) = \frac{\vec{w}_u \cdot \vec{w}_s}{\|\vec{w}_u\| \cdot \|\vec{w}_s\|} = \\ &= \frac{\sum_{i=1}^K w_{iu} w_{is}}{\sqrt{\sum_{i=1}^K w_{iu}^2} \sqrt{\sum_{i=1}^K w_{is}^2}} \end{aligned} \quad (2)$$

Collaborative filtering algorithm:

1) For each user of the system weights are determined by analyzing their preferences in relation to this user by calculating the degree of similarity

$$\text{sim}(u, U') = \frac{\sum_{s \in S_{uU'}} (r_{us} - \bar{r}_u)(r_{U's} - \bar{r}_{U'})}{\sqrt{\sum_{s \in S_{uU'}} (r_{us} - \bar{r}_u)^2 \sum_{s \in S_{uU'}} (r_{U's} - \bar{r}_{U'})^2}} \quad (3)$$

2) Form a group of the most similar users in relation to this user.

3) To form the forecasted rating of object, analyzing estimations of each of users from the previously formed group of users:

$$r_{Us} = k \sum_{u \in U'} \text{sim}(u, U') \cdot r_{Us} \quad (4)$$

Hybrid filtration method:

1) For each of the filtration methods described above, determine the weights. To do this, set the following system of equations:

$$\begin{cases} \gamma = \frac{1}{1 + e^{-\frac{x}{2}}}, \\ \alpha = \beta = 1 - \frac{1}{1 + e^{-\frac{x}{2}}}, \end{cases} \quad (5)$$

where  $x$  is the number of user ratings in the system. And the values change dynamically depending on the available estimates.

2) The resulting projected rating of objects is formed:

$$S(x) = \frac{1}{1 + e^{-x}} + \frac{e^x}{e^x + 1} \quad (6)$$

Thus, in the work for the first time it is proposed to use a combination of collaborative and content filtration for literary services, which provides new users based on the assessment of one book to obtain a more accurate recommendation for receiving references to the following books.

**Results.** The operation of the system is given in the form of a sequence diagram in fig. 3. In this case, the sequence diagram showing the main precedents of the literary service is the basis for software design.

In order to obtain literature, the following inputs are required: user information that helps avoid the problem of "cold start", book information and user assessment information. The system uses the recommendation algorithm, takes into account the service requirements and provides personal guidelines for the user.

To implement the recommendation system, a hybrid algorithm, which combines two methods: a collaborative on the basis of neighbors and content filtration (fig. 4) was chosen.

The algorithm consists of six major stages; on which we must receive books for the end user. The algorithm input provides information about books, user and

estimates that have left users with these books. Next, the algorithms of content and collaborative filtration are performed, with which we receive lists of similar elements and similar users. And in the end, a list of recommendations is formed.

To check the work of this literary service and recommendation system, a matrix fragment of users and their ratings is given (table 1), where 5 is the highest rating.

For the user Igor, calculating the degree of proximity according to the formulas (1), (3), a set of neighbors is obtained and a similarity matrix is formed (table 2).

Expressions (2), (4) are used to predict the rating of the five new elements: multiply each user's score by the degree of similarity. The calculated sum of ratings of each book is divided by the sum of measures of similarity of the closest users are summarized in table 3.

Book1, Book5, Book8 and Book11 have already been rated by the user of the Games and it makes no sense to re-offer them to him. Book3, Book6, Book9 – books that are not included in the ranking of five books due to low coefficient.

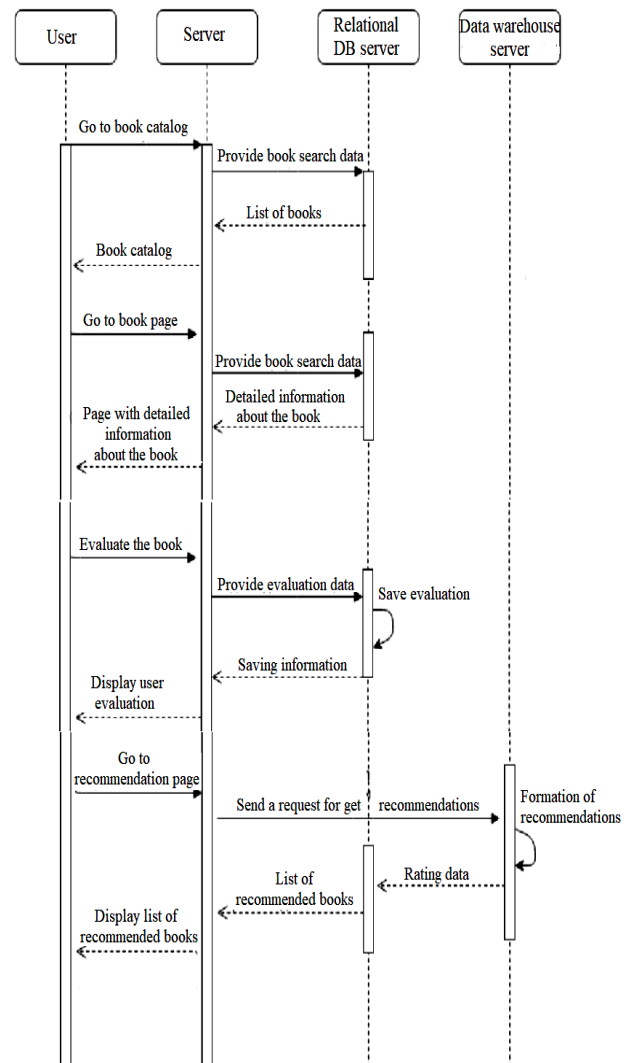


Fig. 3. Sequence diagram



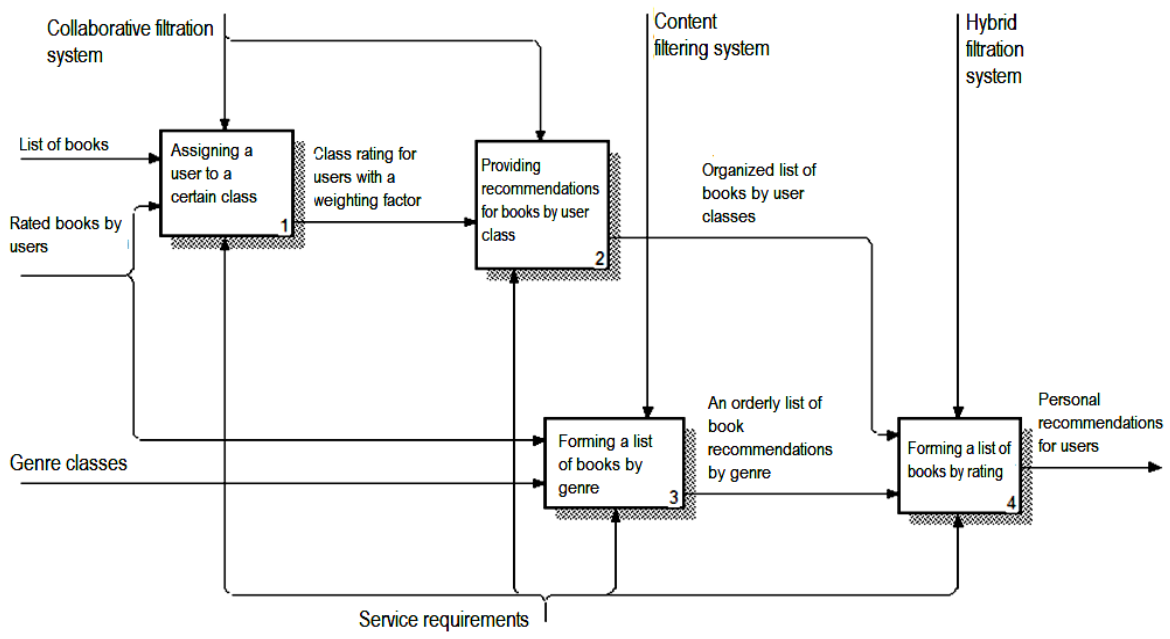


Fig. 4. Model of the system in the form of IDEF0 chart

Table 1 – Estimation matrix

	Book											
	1	2	3	4	5	6	7	8	9	10	11	12
Ivan	5	3		5	4		5	1		4		2
Igor	4				4			2	4			
Anna		1	5				4					
Nina			4	3		2	1				5	2
Nadiya	4	5		4	5			4			1	5
Artem			4	3	1		5	4	1			1

Table 2 – Similarity matrix

	Ivan	Anna	Nina	Nadiya	Artem	Summ
Igor	0,5831	0	0,1334	0,215	0,0359	0,9674

Table 3 – The results of calculations of personal recommendations

	Book											
	1	2	3	4	5	6	7	8	9	10	11	12
Ivan	2,92	1,75	-	2,92	2,33	-	2,92	0,58	-	2,33	-	1,17
Anna	-	-	-	-	-	-	-	-	-	-	-	-
Nina	-	-	0,53	0,40	-	0,27	0,13	-	-	0,67	0,27	-
Nadiya	0,86	1,08	-	0,86	1,08	-	-	0,86	-	0,22	-	1,08
Artem	-	-	0,14	0,11	0,04	-	0,18	0,14	0,04	-	0,04	-
Summ	3,78	2,82	0,68	4,28	3,44	0,27	3,23	1,59	0,04	3,21	0,30	2,24
Result	3,90	2,92	0,70	4,43	3,56	0,28	3,34	1,64	0,04	3,32	0,31	2,32

To provide a set of book recommendations, the results are sorted in descending order and the number of books to display on the recommendation page is selected (in this case, five books will be displayed). Personal recommendations of the literature for the user Igor are entered in table 4.

Table 4 – Personal recommendations for the user Igor

№ Book	Rating
Book 4	Highest rating – 4,43
Book 7	rating – 3,34
Book 10	rating – 3,32
Book 2	rating – 2,92
Book 12	rating – 2,32

The results of testing the implemented system of personal recommendations in the developed web application for the user Igor are shown in fig. 5.

Recommendations for You

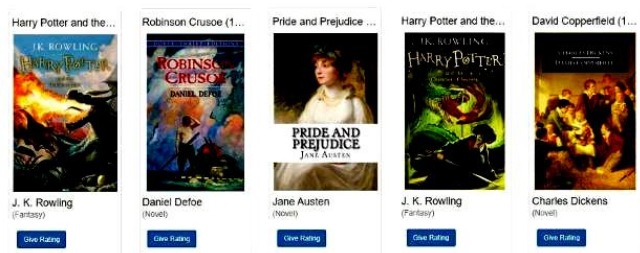


Fig. 5. Web-page of personal recommendations for the user Igor

In the event that a new user enters the system, the service analyzes his preferences on the basis of personal information and provides a list of recommendations based on it.

**Conclusions.** When the number of books provided by the electronic library is relatively large, it becomes difficult for the user to choose the right book. Thus, a solution to the problem is implemented and a hybrid algorithm is proposed, which is not just based on joint filtering and content-based algorithm, but also combines several hybrids. This approach helps to solve the problems

of each type of filtration, in particular the problem of "cold start".

During the testing of the developed software, the work of the literature service was checked, which calculates personal recommendations for users using the method of hybrid filtering. The recommendation system was tested successfully and showed high efficiency.

**Discussion.** The obtained results were checked on the work of the created literary service, namely books were uploaded to the service, accounts were created by real people and work on selection of the recommended literature on the basis of user preferences was started. Each user read several books and gave them a subjective assessment. The user also added his favorite genres to the settings, as a result of which personal recommendations were formed. As soon as the user read the book from the provided list and evaluated it, the list changed depending on the input data. Read books were not included in the list.

As a result of the experiment, recommendations were obtained for users that met their expectations and preferences.

#### References

- Isinkaye F. O., Folajimi Y. O., Ojokoh B. A. Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*. 2015. Volume 16, issue 3. P. 261–273.
- Venkatesan M., Thangadurai K. History and overview of the recommender systems. *Collaborative Filtering Using Data Mining and Analysis*, 2016. P. 74–99.
- Shardanand U., Maes P. Social Information Filtering: Algorithms for Automating «Word of Mouth». *Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA, 1995. P. 210–217.
- Hill W., Stead L., Rosenstein M., Furnas G. Recommending and Evaluating Choices in a Virtual Community of use. *Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA, 1995. P. 194–201.
- Resnick P., Iakovou N., Sushak M., Bergstrom P., Riedl J. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 1994. P. 175–186.
- Celma O. Music Recommendation and Discovery. *Springer*, Berlin, Heidelberg, 2010. P. 43–85.
- Adomavicius G., Tuzhilin A. 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*. 2005. Volume 17, issue 6. P. 734–749.
- Burke R., Felfernig A., & Göker M. H. Recommender Systems: An Overview. *AI Magazine*. 2011. No. 32(3). P. 13–18.
- How Reddit ranking algorithms work, URL: <http://amix.dk/blog/post/19588> (accessed 27.03.2021).
- Shahbazi Z., Byun Y.C. Product Recommendation Based on Content-based Filtering Using XGBoost Classifier. *International Journal of Advanced Science and Technology*. No. 29(04). P. 6979–6988. URL: <http://sersc.org/journals/index.php/IJAST/article/view/28099> (accessed 27.03.2021).
- Madadipouya K., Chelliah S. A Literature Review on Recommender Systems Algorithms, Techniques and Evaluations. *BRAIN*. 2017. Vol.8, issue 2. P. 109–124.
- Phys J. Summary of recommendation system development, *IOP Conf. Series: Journal of Physics: Conf. Series 1187*. P. 1–5.
- Foreign online store of literature and training videos, URL: <http://shop.oreilly.com/> (accessed 25.03.2021).
- Free online cinema ivi.ru, <http://www.ivi.ru/> (accessed 25.03.2021).
- Ricci F., Rokach L., Shapira B., Kantor P.B. *Recommender Systems Handbook*. Springer US, 2011. 842 p.
- Aggarwal C. C. *Data mining*. Springer International Publishing, 2015. 734 p.
- Portugal I., Alencar P., Cowan D. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*. 2018. Vol. 97. P. 205–227.
- Li X., Wang Z., Wang L., Hu R., Zhu Q. A multidimensional context-aware recommendation approach based on improved random forest algorithm. *IEEE Access*. 2018. Vol. 6. P. 45071–45085.
- Wang X., Wen J., Luo F., Zhou W., Ren H. Personalized recommendation system based on support vector machine and particle swarm optimization. *International Conference on Knowledge Science*. 2015. P. 489–495.
- Marović M., Mihoković M., Mikša M., Pribil S., Tus A. Automatic movie ratings prediction using machine learning. *34th International Convention on Information and Communication Technology, Electronics and Microelectronics*, 2011. P. 1640–1645.
- Çano E., Morisio M. Hybrid recommender systems: a systematic literature review. *Intelligent Data Analysis*. 2017. Vol. 21, no. 6. P. 1487–1524.
- Fayyaz Z., Ebrahimian M., Nawara D. Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. *Multidisciplinary Digital Publishing Institute*. 2020. Vol. 10, issue 21. P. 7748.
- Beladev M., Rokach L., Shapira, B. Recommender systems for product bundling. *Knowl. Based Syst.* 2015. Vol. 111. P. 193–206.
- Naveen G., Naidu A., Thirumala Dr. B. Comparative Study on Artificial Intelligence and Expert Systems *International Research Journal of Engineering and Technology*. 2019. Vol. 6, issue 2. P. 1980–1986.
- Vilnius K. Intelligent Decision Support Systems. *Biometric and Intelligent Decision Making Support*. 2015. P. 31–85.
- Peter B. Keenan, Piotr J. Spatial Decision Support Systems: Three decades on. *Decision Support Systems*. 2019. Vol. 116. P. 64–76.
- Waila, P., Singh, V., Singh, M. A Scientometric Analysis of Research in Recommender Systems. *J Scientometric Res*. 2016. Vol. 5, issue 1. P. 71–84.
- Zhang Q., Jie Lu. Artificial intelligence in recommender systems. *Complex & Intelligent Systems*. 2020. Vol. 7. P. 439–457.
- Luy Z., Dou Z., Lianz J. Content-Based Collaborative Filtering for News Topic Recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2015. No. 29(1). P. 217–223.
- Li X., Xing J., Wang H., Zheng L., Jia S., Wang Q. A Hybrid Recommendation Method Based on Feature for Offline Book Personalization. *Journal of Computers*. 2019. Vol. 30, no. 5. P. 1–17.
- Lytvynenko, V., Lurie, I., Krejci, J. Two Step Density-Based Object-Inductive Clustering Algorithm. *Workshop Proceedings of the 8th International Conference on "Mathematics. Information Technologies. Education"*, 2019. P. 117–135.
- Smith J., Weeks D., Freeman J., Jacob M., Magerko B., Towards a Hybrid Recommendation System for a Sound Library. *IUI Workshops '19*, 2019. Vol. 2327. 6 p.
- Chyrun L., Burov Y., Rusyn B. Web Resource Changes Monitoring System Development. *Workshop Proceedings of the 8th International Conference on "Mathematics. Information Technologies. Education"*, 2019. P. 255–273.

#### References

- Isinkaye F. O., Folajimi Y. O., Ojokoh B. A. Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*. 2015, vol. 16, issue 3, pp. 261–273.
- Venkatesan M., Thangadurai K. History and overview of the recommender systems. *Collaborative Filtering Using Data Mining and Analysis*. 2016, pp. 74–99.
- Shardanand U., Maes P. Social Information Filtering: Algorithms for Automating «Word of Mouth». *Proceedings of the SIGCHI conference on Human factors in computing systems*. [Proceedings of the ACM CHI 95 Human Factors in Computing Systems Conference] New York, NY, USA.1995, pp. 210–217.
- Hill W., Stead L., Rosenstein M., Furnas G. Recommending and Evaluating Choices in a Virtual Community of use. *Proceedings of the SIGCHI conference on Human factors in computing systems*. [Proceedings of the ACM CHI 95 Human Factors in Computing Systems Conference] New York, NY, USA. 1995, pp. 194–201.

5. Resnick P., Iakovou N., Sushak M., Bergstrom P., Riedl J. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. *Proceedings of the 1994 ACM conference on Computer supported cooperative work*. [CSCW '94 Computer Science]. 1994, pp. 175–186.
6. Celma O. Music Recommendation and Discovery. *Springer* [Springer. Computer science], Berlin, Heidelberg, 2010, pp. 43–85.
7. Adomavicius G., Tuzhilin A. 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* [IEEE Computer Society]. 2005, Vol. 17, issue 6, pp. 734–749.
8. Burke, R., Felfernig, A., & Göker, M. H. Recommender Systems: An Overview. *AI Magazine*. 2011, no. 32(3), pp. 13–18.
9. How Reddit ranking algorithms work, URL: <http://amix.dk/blog/post/19588> (accessed 27.03.2021).
10. Shahbazi, Z. Byun, Y.C. Product Recommendation Based on Content-based Filtering Using XGBoost Classifier. *International Journal of Advanced Science and Technology*. 29(04), pp. 6979–6988. URL: <http://serc.org/journals/index.php/IJAST/article/view/28099> (accessed 27.03.2021).
11. K. Madadipouya, S. Chelliah. A Literature Review on Recommender Systems Algorithms, Techniques and Evaluations. *BRAIN* [Broad Research in Artificial Intelligence and Neuroscience], 2017, vol. 8, issue 2, pp. 109–124.
12. Phys J. Summary of recommendation system development, *IOP Conf. Series: Journal of Physics: Conf. Series* 1187, pp. 1–5.
13. Foreign online store of literature and training videos, URL: <http://shop.oreilly.com/> (accessed 25.03.2021).
14. Free online cinema ivi.ru. <http://www.ivi.ru/> (accessed 25.03.2021).
15. Ricci F., Rokach L., Shapira B., Kantor P.B. *Recommender Systems Handbook*. Springer US, 2011. 842 P.
16. Aggarwal C. C. *Data mining*. Springer International Publishing, 2015. 734 p.
17. Portugal I., Alencar P., Cowan D. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*. 2018, vol. 97, pp. 205–227.
18. Li X., Wang Z., Wang L., Hu R., Zhu Q. A multidimensional context-aware recommendation approach based on improved random forest algorithm. *IEEE Access*. 2018, vol. 6, pp. 45071–45085.
19. Wang X., Wen J., Luo F., Zhou W., Ren H. Personalized recommendation system based on support vector machine and particle swarm optimization. *International Conference on Knowledge Science* [International Conference on Knowledge Science, Engineering and Management]. 2015, pp. 489–495.
20. Marović M., Mihoković M., Mikša M., Pribil S., Tus A. Automatic movie ratings prediction using machine learning. *34th International Convention on Information and Communication Technology, Electronics and Microelectronics*. 2011, pp. 1640–1645.
21. Çano E., Morisio M. Hybrid recommender systems: a systematic literature review. *Intelligent Data Analysis*. 2017, vol. 21, no. 6, pp. 1487–1524.
22. Fayyaz Z., Ebrahimian M., Nawara D. Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. *Multidisciplinary Digital Publishing Institute*. 2020, vol.10, issue 21, p. 7748.
23. Beladev M., Rokach L., Shapira, B. Recommender systems for product bundling. *Knowl. Based Syst.* 2015, vol. 111, pp. 193–206.
24. Naveen G., Naidu A., Thirumala Dr. B. Comparative Study on Artificial Intelligence and Expert Systems. *International Research Journal of Engineering and Technology* 2019, vol. 6, issue 2, pp. 1980–1986.
25. Vilnius K. Intelligent Decision Support Systems. *Biometric and Intelligent Decision Making Supporto* 2015, pp 31–85.
26. Peter B. Keenan, Piotr J. Spatial Decision Support Systems: Three decades on. *Decision Support Systems*. 2019, vol. 116, pp. 64–76.
27. Wailla, P., Singh, V., Singh, M. A Scientometric Analysis of Research in Recommender Systems. *J Scientometric Res*. 2016, vol. 5, issue 1, pp. 71–84.
28. Zhang Q., Jie Lu. Artificial intelligence in recommender systems. *Complex & Intelligent Systems*. 2020, vol. 7, pp. 439–457.
29. Luy Z., Dou Z., Lianz J. Content-Based Collaborative Filtering for News Topic Recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2015, no. 29(1), pp. 217–223.
30. Li X., Xing J., Wang H., Zheng L., Jia S., Wang Q. A Hybrid Recommendation Method Based on Feature for Offline Book Personalization. *Journal of Computers*. 2019, vol. 30, no. 5, pp. 1–17.
31. Lytvynenko V., Lurie I., Krejci J. Two Step Density-Based Object-Inductive Clustering Algorithm. *Workshop Proceedings of the 8th International Conference on "Mathematics. Information Technologies. Education"*. 2019, pp. 117–135.
32. Smith J., Weeks D., Freeman J., Jacob M., Magerko B., Towards a Hybrid Recommendation System for a Sound Library. *IUI Workshops'19* [Joint Proceedings of the ACM IUI 2019 Workshops]. 2019, vol. 2327, 6 p.
33. Chyrun L., Burov Y., Rusyn B. Web Resource Changes Monitoring System Development. *Workshop Proceedings of the 8th International Conference on "Mathematics. Information Technologies. Education"*. 2019, pp. 255–273.

Received 07.11.2021

## Відомості про авторів / Сведения об авторах / About the Authors

**Козуля Марія Михайлівна** – кандидат технічних наук, Національний технічний університет «Харківський політехнічний інститут», доцент кафедри програмної інженерії та інформаційних технологій управління; м. Харків, Україна; ORCID: <https://orcid.org/0000-0002-4090-8481>; e-mail: [mariia.kozulia@khp.edu.ua](mailto:mariia.kozulia@khp.edu.ua)

**Сушко Владислава Володимирівна** – Національний технічний університет «Харківський політехнічний інститут», студент, магістр; м. Харків, Україна; ORCID: <https://orcid.org/0000-0001-8759-6203>; e-mail: [sushko.kn36a@gmail.com](mailto:sushko.kn36a@gmail.com)

**Козуля Марія Михайлівна** – кандидат технических наук, Национальный технический университет «Харьковский политехнический институт» доцент кафедры программной инженерии и информационных технологий управления; г. Харьков, Украина; ORCID: <https://orcid.org/0000-0002-4090-8481>; e-mail: [mariia.kozulia@khp.edu.ua](mailto:mariia.kozulia@khp.edu.ua)

**Сушко Владислава Владимировна** – Национальный технический университет «Харьковский политехнический институт», студент, магистр; г. Харьков, Украина; ORCID: <https://orcid.org/0000-0001-8759-6203>; e-mail: [sushko.kn36a@gmail.com](mailto:sushko.kn36a@gmail.com)

**Kozulia Mariia Mikhailovna** – candidate of engineering science, National Technical University "Kharkiv Polytechnic Institute", associated professor of the Department of Software Engineering and Management Information Technologies, Kharkiv, Ukraine; ORCID: <https://orcid.org/0000-0002-4090-8481>; e-mail: [mariia.kozulia@khp.edu.ua](mailto:mariia.kozulia@khp.edu.ua)

**Sushko Vladislava Vladimirovna** – National Technical University "Kharkiv Polytechnic Institute", student, master; Kharkiv, Ukraine; ORCID: <https://orcid.org/0000-0001-8759-6203>; e-mail: [sushko.kn36a@gmail.com](mailto:sushko.kn36a@gmail.com)