

# Methods of information systems synthesis

UDC 004.6

doi: 10.20998/2522-9052.2018.4.09

Ye. Meleshko

Central Ukrainian National Technical University, Kropyvnytskyi, Ukraine

## METHOD OF COLLABORATIVE FILTRATION BASED ON ASSOCIATIVE NETWORKS OF USERS SIMILARITY

The **subject matter** of the article is the processes of generating a recommendations list for users of a website. The **goal** is to develop the new method of collaborative filtering based on building associative networks of users similarity to improve the quality of recommender systems. The **tasks** to be solved are: to develop the method of collaborative filtering based on building associative networks of user similarity, develop software to test this method, conduct experiments on the developed software to test the effectiveness of the developed method, determine the quality of its work and compare this method with the standard method of collaborative filtering. The **methods** used are: graph theory, mathematical statistics, the theory of algorithms, object-oriented programming. The following **results** were obtained: the method of collaborative filtering based on building associative networks of user similarity was developed, to implement this method the software was developed, experiments using the developed software to test the developed method were conducted. **Conclusions.** The possibility of using associative networks in recommender systems was researched. The associative rule for building associative networks of users similarity was proposed. The collaborative filtering method based on associative networks of users similarity, which can be used to improve the quality of recommender systems, was developed. Experiments conducted on the developed software have shown that the developed method significantly increases such performance indicators of the recommender system as user space coverage, item space coverage, user interaction coverage, and makes it possible to create better-quality lists of recommendations for website users.

**Keywords:** recommendation systems; collaborative filtering; associative networks; similarity coefficients.

### Introduction

Today, recommendation systems are used to improve a work of many types of websites, such as online stores, content websites, search engines, and more. They are a good addition to the classic data search algorithms and can significantly increase the overall interest in a website, and give individual users the opportunity to get more useful information and pay attention to more objects that are relevant to his or her preferences.

The working principle of a recommendation system is based on the fact that for each user in a set of users  $K = \{k_1, k_2, \dots, k_n\}$  creates a set of recommendations  $R = \{r_1, r_2, \dots, r_m\}$ , sorted in descending order the relevance of a recommendation to a interests of corresponding a user.

One of the basic algorithms for building recommendation systems is collaborative filtration [1, 2]. It is based on the calculation of similarity coefficients between users to find a most similar users and objects.

To calculate similarity coefficients, previously collected information about users and objects of a system is used, such as ratings that users put on objects, user transactions, pageview history, description of objects and their characteristics, etc.

More often, in recommendation systems the following similarity coefficients are used: Euclidean distance (1), Heming distance (2), Pearson correlation coefficient (3), cosine similarity (4), and others:

$$d(x_1, x_2) = \sqrt{\sum_{i=1}^m (x_{1i} - x_{2i})^2}; \quad (1)$$

$$d(x_1, x_2) = \sum_{i=1}^m |x_{1i} - x_{2i}|; \quad (2)$$

$$d(x_1, x_2) = \frac{X_1 \cdot X_2}{\|X_1\| \cdot \|X_2\|} = \frac{\sum_{i=1}^m x_{1i} \cdot x_{2i}}{\sqrt{\sum_{i=1}^m x_{1i}^2} \cdot \sqrt{\sum_{i=1}^m x_{2i}^2}}; \quad (3)$$

$$d(x_1, x_2) = \frac{\sum_{i=1}^m (x_{1i} - X_1) \cdot (x_{2i} - X_2)}{\sqrt{\sum_{i=1}^m (x_{1i} - X_1)^2} \cdot \sqrt{\sum_{i=1}^m (x_{2i} - X_2)^2}}, \quad (4)$$

where  $d(x_1, x_2)$  – the distance between objects  $x_1$  and  $x_2$ ;  $x_{1i}$ ,  $x_{2i}$  – the value of the  $i$ -th attribute, respectively, in the 1st and 2nd objects;  $X_1$ ,  $X_2$  – the set of values of attributes in the 1st and 2nd objects.

After defining similarity coefficients between elements of a system, they are used to select elements similar to those previously selected by a user for their further recommendation.

Not always a information gathered about elements of a system is enough to determine a required number of similar items to a given item. To solve this problem, an additional analysis of a available data is needed.

Associative Network (AN) is a set of objects and associative relationships between them.

Associative relationships are built on the basis of associative rules.

Associative rules allows find patterns between related facts. For example, if user A will buy  $x_1$ , then he will also buy  $x_2$  with probability  $p$  [3].

Associative rules can be used in recommendation systems to determine the similarity between objects that were not detected using similarity coefficients.

The rules in form "if A and B, then with probability  $p$  also C", describe, for example, products that was jointly purchased. In this case, the search for associative rules can be done using, for example, the following queries to a system:

- What products are sells together?
- How often do A and B products sell together?
- How the choice of a product A increases the probability of choosing a product B?
- How the choice of a product B increases the probability of selecting a product A?

Some of the most common algorithms used to construct associative rules are APriori [4], DHP [5], Partition [6], DIC [7], and others.

If the association rules apply to descriptive analysis, to determine which products are sells together, recommendation system can get some new information that can be used in the process of forming a recommendation.

The disadvantage of this kind of information is that there will be a low item space coverage with associative rules: such rules will exist only for a small number of items. At the same time, even under such conditions, the application of associative rules and the use of an associative network of items will improve the quality of a recommendation system.

The goal of this work was to create collaborative filtering method based on associative networks of similarity of users in order to increase the amount of data on the basis of which a list of recommendations is built.

### The main material

When developing recommendation systems based on collaborative filtering, there is the following pattern: the more users are identified with high similarity coefficients for a particular user, the more useful recommendations for him will be can to create. In this way, the task of finding as many as possible of similar users becomes relevant. It should be investigated whether more relationships of similarity can be found between users that correspond to reality than the quantity that can be found through collaborative filtering. Associative rules and associative networks were used to solve this problem.

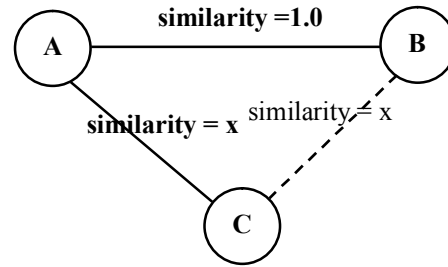
Associative rules in recommendation systems are used to determine associative relationships between products, based on how often different products fall into one transaction.

In this paper, associative relationships are proposed to be built between users based on their similarity coefficients in order to find more similar users.

The following associative rule was developed:

**If** the similarity coefficient of users  $A$  and  $B$  equals 1, this means, users are "completely" similar, **and** the similarity coefficient between users  $A$  and  $C$  equals  $x$ , **then** the similarity between users  $B$  and  $C$  equals  $x$ .

Let's illustrate this rule schematically in Fig. 1/



**Fig. 1.** Schematic representation of the associative rule for determining a similarity of users

The associative network is proposed to be built as follows:

1) apply the proposed associative rule to build additional similarity relationships (similarity of level 2) between users;

2) repeat the action (1) for a network, taking into account similarity relationships with the basic similarity and similarity of level 2 for obtaining similarities of level 3.

To create recommendations it is suggested to use all three types of similarity relationships and consider them to be equivalent.

The series of experiments were conducted to test this rule.

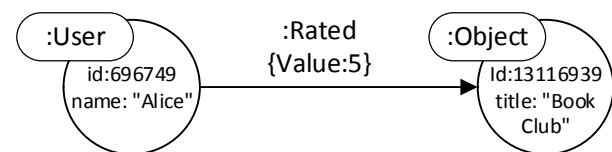
To test the work of the developed method, the software to create recommendations and testing the recommendation system was developed.

Consider the developed software. To build the recommendation system, it was decided to choose the programming language Python and database type NoSQL with the representation of data in the form of a graph - database management system Neo4j.

Neo4j is the first and one of the most popular graph database management systems. It has the application programming interface for many programming languages, including Java, Python, Ruby, PHP [8].

To perform queries, it uses own language Cypher.

The example of the data format in the Neo4j database is shown in Fig. 2.



**Fig. 2.** The example of the data format in the Neo4j database

To add nodes and relationships to the Neo4j database, the following queries can be used:

```
// creating nodes
CREATE (User1:User {id:$id_user1,
name:$name_user1})
CREATE (Object1: Object {id:$id_Object1,
title:$title_Object1})
// creating relationship
CREATE (User1)-[:Rated{Value:5}]->(Object1)
```

To work with Neo4j in Python the neo4j.v1 library was used. It allows you to connect to a database server and make requests.

Recommendations for users of a web-resource are formed on the basis of previously collected information about them and information about objects of a system, which is convenient to represent in the form of a graph (Fig. 3) and write to a graph database.

Advantages of using graph database Neo4j for recommendation systems [8]:

**1. Productivity.** Allows calculating recommendations in real-time, ensuring their actuality.

**2. Convenient data model.** Labels and properties of nodes and relationships allow to easily filter datasets and allocate a necessary subgraph for analysis.

The recommended system based on collaborative filtration was built, and similarity coefficients between users were determined based on the Pearson correlation coefficient (2).

In the developed recommendation system, all data is represented as a graph, the example is shown in Fig. 3. All data is divided into vertices and edges. For example, users, objects, properties of objects are vertices. The edges are represented by relationships of type: "rated", "friends", "has\_characteristic".

Ratings, for example, are the weight of the edges type "rated".

Making requests to the database based on labels of nodes and labels of relationships gives the opportunity to get different subgraphs.

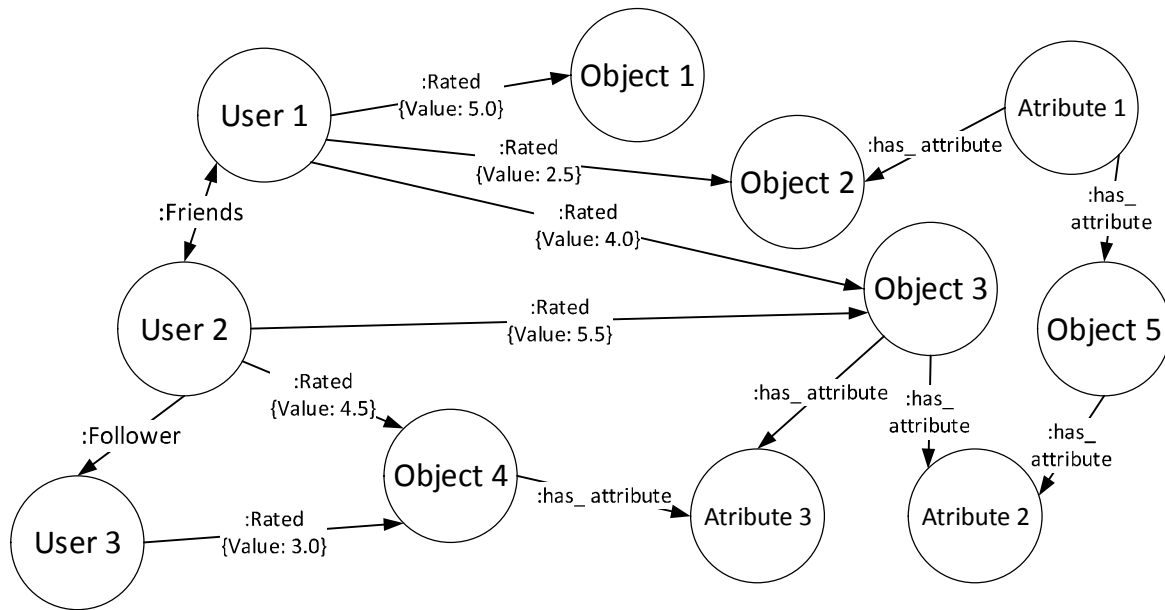


Fig. 3. The example of an input data set for the recommendation system

The developed recommendation system was tested on the open dataset – MovieLens Datasets, that was created in research laboratory at the Department of Computer Science and Engineering at the University of Minnesota [9].

During each experiment on the MovieLens Datasets,  $N_u$  users were selected. The ratings that they have set on movies were divided into two parts by timestamp for to calculate recommendations ("current data") and to test the system ("future data"). For each data set the system was launched in two modes:

- without using the associative network (without AN);
- using the associative network (with AN).

In the dataset, ratings may take values: 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0. It was decided to divide them into positive (from 3.5 to 5.0) and negative (from 0.5 to 3.0) ratings.

Predictions of users preferences were divided into positive ones when predicting positive rating, and negative ones when predicting negative rating.

To check the quality of the recommendation system, the following metrics were used:

1. Prediction accuracy – shows how accurately is predicted preferences of users.
2. User space coverage – the percentage of all users for whom the system can provide recommendations.
3. Item space coverage – the percentage of all objects that can be recommended to users.
4. User interaction coverage – the percentage of all objects among chosen by users that were recommended.

Let's consider how these metrics were determined in the system being developed.

First, consider all possible results of giving a recommendation to a user (Table 1).

Table 1 – Classification of possible results of a recommendation

	Recommended	Not recommended
Rated positive	True-Positive ( <i>tp</i> )	False-Negative ( <i>fn</i> )
Rated negative	False-Positive ( <i>fp</i> )	True-Negative ( <i>tn</i> )

Prediction accuracy of a recommendation system was calculated by the formula (4):

$$\text{Precision} = \frac{tp}{tp + fp}, \quad (4)$$

User space coverage was determined by the formula (5):

$$\text{User Space Coverage} = \frac{N_{ur}}{N_u}, \quad (5)$$

where  $N_{ur}$  – the number of users for which it was possible to create recommendations,  $N_u$  – the number of users that was taken in the experiment.

Item space coverage was determined by the formula (5):

$$\text{Item Space Coverage} = \frac{N_{mr}}{N_m}, \quad (6)$$

where  $N_{mr}$  – the number of movies for which were able to predict ratings,  $N_m$  – the total number of movies in the experiment.

User interaction coverage was determined by the formula (5):

$$\text{User Interaction Coverage} = \frac{tp}{N_{mc}}, \quad (7)$$

where  $N_{mr}$  – the number of correctly predicted positive ratings,

$N_{mc}$  – the total number of movies that were selected by users in the test data.

Table 2 – Test results of the developed method of construction of recommendation systems

Experiment number	Number of users	Number of correct positive recommendations		Prediction accuracy		User space coverage		Item space coverage		User interaction coverage	
		without AN	with AN	without AN	with AN	without AN	with AN	without AN	with AN	without AN	with AN
1	30	59	130	86.76%	69.89%	56.66%	83.33%	7.31%	19.56%	4.66%	10.28%
2	30	55	104	77.46%	81.25%	73.33%	83.33%	6.48%	12.16%	6.19%	11.71%
3	50	292	543	78.91%	76.80%	86.00%	92.00%	21.37%	45.50%	10.53%	19.59%
4	50	211	346	79.02%	78.28%	74.00%	90.00%	20.06%	33.51%	15.41%	25.27%
5	100	907	1153	74.71%	75.31%	94.00%	98.00%	60.48%	79.24%	28.27%	35.94%
6	100	821	1033	64.95%	61.16%	96.00%	100.00%	54.87%	77.46%	14.61%	18.38%

Based on the results of experiments it can be concluded that the developed method allows to increase the number of correct positive recommendations, user space coverage, item space coverage and user interaction coverage practically without reducing prediction accuracy of a recommendation system, and in some cases prediction accuracy even increased (experiment number 2 and number 5). In general, the number of useful recommendations for each user increases and the number of users for which there are no recommendations is reduced.

### Conclusions

The possibility of using associative networks in recommendation systems was explored.

The associative rule for building associative networks of users similarity was proposed.

The method of collaborative filtration based on associative networks of users similarity was developed.

Series of experiments has been carried out, which showed that the developed method increases the number of correct positive recommendations, user space coverage, item space coverage, user interaction coverage, practically without changing the prediction accuracy. Using the developed method significantly increases the number of useful recommendations, and reduces the number of users for whom the usual collaborative filtering could not provide recommendations.

### REFERENCES

1. Meleshko E.V., Semenov, S.G. and Khokh, V.D. (2018), “Research of methods of building advisory systems on the internet”, *Academic Journal "Control, Navigation and Communication Systems"*, Issue 1(47), Poltava National Technical Yuri Kondratyuk University, Poltava, pp. 131–136, DOI: <https://doi.org/10.26906/SUNZ.2018.1.131> (in Ukrainian).
2. Jones, M. (2013), *Recommender systems*, Part 1. Introduction to approaches and algorithms. Learn about the concepts that underlie web recommendation engines, available at:

- [https://www.ibm.com/developerworks/opensource/library/os-recommender/1/index.html?s\\_tact=105agx99&s\\_cmp=cp](https://www.ibm.com/developerworks/opensource/library/os-recommender/1/index.html?s_tact=105agx99&s_cmp=cp)
3. Shahidi, A. (2015), Introduction to analysis of the associative rules, available at: <https://basegroup.ru/community/articles/intro> (in Russian).
  4. Agrawal, R. and Srikant, R. (1994), "Fast Discovery of Association Rules", *Proc. of the 20th International Conference on VLDB*, Santiago, Chile, 1215, pp. 487-499.
  5. Savasere, A., Omiecinski, E. and Navathe S. (1995), "An Efficient Algorithm for Mining Association Rules in Large Databases", *Proc. 21st Int'l Conf. Very Large Data Bases*, Morgan Kaufmann, San Francisco, pp.422-434.
  6. Park, J.S., Chen, M.-S. and Philip, S.Y. (1995), "An Effective HashBased Algorithm for Mining Association Rules", *Proc. ACM SIGMOD Int'l Conf. Management of Data*, ACM Press, New York, pp. 175-186.
  7. Brin, S., Motwani, R., Ullman, Jeffrey D. and Tsur, S. (1997) "Dynamic Itemset Counting and Implication Rules for Market Basket Data", *Proc. ACM SIGMOD Int'l Conf. Management of Data*, ACM Press, New York, pp. 255-264
  8. (2018), *Neo4j Documentation*, available at: <https://neo4j.com/docs/>
  9. Harper, F.M. and Konstan J.A. (2016) "The MovieLens Datasets: History and Context", *ACM Transactions on Interactive Intelligent Systems (TiiS)*, No. 19, DOI: <https://doi.org/10.1145/2827872>.

Received (Надійшла) 29.08.2018

Accepted for publication (Прийнята до друку) 31.10.2018

### Метод колаборативної фільтрації на основі асоціативних мереж подоби користувачів

Є. В. Мелешко

**Предметом** вивчення у статті є процес генерації списку рекомендацій для користувачів веб-сайту. **Метою** є розробка нового методу колаборативної фільтрації на основі побудови асоціативних мереж подоби користувачів для підвищення якості роботи рекомендаційних систем. **Завдання:** розробити метод колаборативної фільтрації на основі побудови асоціативних мереж подоби користувачів, розробити програмне забезпечення для тестування даного методу, провести експерименти на розробленому програмному забезпеченні для перевірки ефективності застосування розробленого методу, визначення якості його роботи та порівняння даного методу зі стандартним методом колаборативної фільтрації. Використовуваними **методами** є: теорія графів, математична статистика, теорія алгоритмів, об'єктно-орієнтоване програмування. Отримані такі **результати:** розроблено метод колаборативної фільтрації на основі побудови асоціативних мереж подоби користувачів, розроблено програмне забезпечення для реалізації даного методу, проведено експерименти на розробленому програмному забезпеченні для тестування розробленого методу. **Висновки.** Досліджено можливість використання асоціативних мереж у рекомендаційних системах. Запропоновано асоціативне правило для побудови асоціативних мереж подоби користувачів. Розроблено метод колаборативної фільтрації на основі асоціативних мереж подоби користувачів, який можна застосовувати для підвищення якості роботи рекомендаційної системи. Проведені на розробленому програмному забезпеченні експерименти показали, що розроблений метод суттєво підвищує такі показники роботи рекомендаційної системи, як покриття простору користувачів, покриття каталогу, покриття взаємодії з користувачами, та дозволяє створювати більш якісні списки рекомендацій користувачам веб-сайтів.

**Ключові слова:** рекомендаційні системи; колаборативна фільтрація; асоціативні мережі; коефіцієнти подоби.

### Метод колаборативної фільтрації на основі асоціативних мереж подоби користувачів

Є. В. Мелешко

**Предметом** изучения в статье является процесс генерации списка рекомендаций для пользователей веб-сайта. **Целью** является разработка нового метода коллаборативной фильтрации на основе построения ассоциативных сетей подоби пользователей для повышения качества работы рекомендательных систем. **Задача:** разработать метод коллаборативной фильтрации на основе построения ассоциативных сетей подоби пользователей, разработать программное обеспечение для тестирования данного метода, провести эксперименты на разработанном программном обеспечении для проверки эффективности применения разработанного метода, определения качества его работы и сравнения данного метода со стандартным методом коллаборативной фильтрации. Используемыми **методами** являются: теория графов, математическая статистика, теория алгоритмов, объектно-ориентированное программирование. Получены следующие **результаты:** разработан метод коллаборативной фильтрации на основе построения ассоциативных сетей подоби пользователей, разработано программное обеспечение для реализации данного метода, проведены эксперименты на разработанном программном обеспечении для тестирования разработанного метода. **Выводы.** Исследована возможность использования ассоциативных сетей в рекомендательных системах. Предложено ассоциативное правило для построения ассоциативных сетей подоби пользователей. Разработан метод коллаборативной фильтрации на основе ассоциативных сетей подоби пользователей, который можно применять для повышения качества работы рекомендательных систем. Проведенные на разработанном программном обеспечении эксперименты показали, что разработанный метод существенно повышает такие показатели работы рекомендательной системы, как покрытие пространства пользователей, покрытие каталога, покрытие взаимодействия с пользователями, и позволяет создавать более качественные списки рекомендаций пользователям веб-сайтов.

**Ключевые слова:** рекомендательные системы; коллаборативная фильтрация; ассоциативные сети; коэффициенты подоби.