Application of machine learning in recommendation systems

Agata Nawrocka
AGH University of Science and Technology
Faculty of Mechanical Engineering and Robotics
Department of Process Control
Krakow, Poland
nawrocka@agh.edu.pl

Andrzej Kot
AGH University of Science and Technology
Faculty of Mechanical Engineering and Robotics
Department of Process Control
Poland, Krakow
e-mail: ankot@agh.edu.pl

Marcin Nawrocki
AGH University of Science and Technology
Faculty of Mechanical Engineering and Robotics
Department of Mining, Dressing and Transport Machines
Krakow, Poland
marcin.nawrocki@agh.edu.pl

***Abstract*— Study includes basic information about machine learning and recommender systems with their examples. More broadly addressed was the topic of machine learning's algorithms, which are used in such systems. The paper mainly focused on filtering algorithms based on the neighborhood of users or objects, and based on content. The description of these algorithms includes: similarities, disadvantages and advantages, measures for evaluating the algorithm, and calculation of the sample value of the evaluation prediction. The design part of the work begins with the description of the used databases from the MovieLens portal. Afterwards, the technology and practical implementation of the algorithms described above are then presented. The next part contains an analysis of the results and conclusions based on the simulations carried out on the computer to assess how the algorithms work. At the end of the work, there is a summary, performance evaluation of recommendation systems, and lessons learned from the project, as well as a proposal for further work on the issue of such systems.**

***Keywords*—machine learning, recommendation, systems filtering.**

I. INTRODUCTION

The Internet has become ubiquitous in the modern world. Used for shopping, watching movies, listening to music or communicating with friends. The activities of a huge number of people - Internet users - open up the possibility of gathering information from thousands, millions, and even billions of people. It is an invaluable opportunity for people who research this data. This allows them to define collective intelligence in the group under study - its behaviour, preferences, and world view. These are valuable insights, for example for the marketing market. They show whether their sales tactics for a given social group work, and sometimes help to come up with the best way to reach a given target group.

The Internet, in turn, allows you to monitor the various activities of network users without affecting their intentions. Data from various social groups around the world are at your fingertips. The biggest problem encountered by researchers and designers of systems based on data mining is the analysis of the information base. Due to its size, people are not able to effectively carry out complicated and tedious calculations on it. The problem of machine learning comes with help. One of the basic methods of machine learning can be statistical methods: regression and correlation analysis. More advanced methods are issues related to learning neural networks or fuzzy logic. The designer creates a recommendation algorithm, and the computer on its basis, acting on a given set of data, determines the conclusions related to the properties of this set. Such systems give great opportunities.

Recommendation systems are becoming more and more popular. Research on their subject has been somehow forced by the development of the Internet and its result - the flood of information. There are thousands of movies, millions of scientific articles, countless amount of music. These numbers mean that a single person is not able to get through all of this in his or her life. The recommendations are of inestimable value for such people.[1].

II. MACHINE LEARNING

For such widely used learning algorithms one can present some major classes of problems with which I have to deal with in order to perform the presented task [2]:

A. Classification

It deals with problems related to assigning classes to each of the analyzed items. An example of such a task can be the problem of image recognition.

978-1-5386-4762-2/18/$31.00 ©2018 IEEE

B. *Regression*

This is an estimate of the actual value of the object. A good example of such a task may be an attempt to estimate the value of bonds on the basis of economic variables.

C. *Ranking*

He is responsible for sorting items according to a specific criterion. The most popular task of this type is the search engine returning websites that satisfy the query sent by the user.

D. *Clustering*

It deals with issues concerning the division of objects into certain homogeneous groups. Such algorithms can be used in social networks to identify smaller social groups from a large number of people.

E. *Dimensionality reduction*

It is the transformation of the original representation of the examined object into a representation of smaller dimensions without losing the original data. Over time, the amount of different data becomes too large for efficient interpretation. Therefore, by discovering the correlation between several variables, they can be reduced to one variable.

III. TYPES OF MACHINE LEARNING ALGORITHMS

Due to different scenarios of availability of training data, test data and evaluation of teaching methods, the following types of machine learning algorithms can be distinguished [2]:

a) *Supervised learning*

In the case of learning with the teacher, the algorithm receives training data in which the output value known from the input data is known. This is one of the most popular learning methods.

b) *Unsupervised learning*

In contrast to the teaching method with the teacher, the algorithm receives training data that does not take into account which output value should be obtained from the input data. In this scenario, the assessment of the extent to which the algorithm has mastered the training data can be troublesome.

c) S*emi-supervised learning*

Training data, while partially supervised, consist of samples having the expected initial value as well as samples that do not have it. This method is popular when the input data is easy to obtain, but the output data is much more expensive.

d) Reinforcement *learning*

The training and testing phases are combined in a reinforcement approach. The learned algorithm, by interacting with the environment, collects data. He receives, depending on the action taken, a reward or penalty. The purpose of this method is to maximize the reward for the learned algorithm.

IV. PRACTICAL IMPLEMENTATION OF THE RECOMMENDATION SYSTEM

The scripts for calculating predictions were written using the Python programming language. In particular, the libraries *pandas* and *numpy* were used for operations on matrices and tables. Algorithms based on the neighborhood of users and objects were selected to generate predictions. To calculate the value of prediction, the key issue is to create a user-object matrix (Tab. 1.).

TABLE I. PART OF THE USER - OBJECT MATRIX

title	'Til There Was You (1997)	1-900 (1994)	101 Dalmatians (1996)	12 Angry Men (1957)	187 (1997)	2 Days in the Valley (1996)	20,000 Leagues Under the Sea (1954)	2001: A Space Odyssey (1968)	3 Ninjas: High Noon At Mega Mountain (1998)
user_id									
1	NaN	NaN	2.0	5.0	NaN	NaN	3.0	4.0	NaN
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0
3	NaN	NaN	NaN	NaN	2.0	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5	NaN	NaN	2.0	NaN	NaN	NaN	NaN	4.0	NaN

The next step is to calculate the value of the similarity function. If you tried to calculate the similarity function based on the cosine distance, the time needed to calculate all the values was too long. Due to limitations resulting from computational capabilities of the equipment on which the scripts were executed, all similarity functions are calculated on the basis of Pearson's correlation coefficient (1) [3,4,5,6].

$$s(u,a)=\frac{\sum_{i\in I_u\cap I_a}\left(r_{u,i}-\bar{r}_u\right)\left(r_{a,i}-\bar{r}_a\right)}{\sqrt{\sum_{i\in I_u\cap I_a}\left(r_{u,i}-\bar{r}_u\right)^2}\sqrt{\sum_{i\in I_u\cap I_a}\left(r_{a,i}-\bar{r}_a\right)^2}} \quad (1)$$

where:
I_u, I_a - collections of object rated by users *a* and *u*
$r_{u,i}, r_{a,i}$ – rating issued to the object by the user *a* and *u*
$\bar{r}_u, \bar{r}_a$ - average of all ratings issued by the user *a* and *u*

TABLE II. PART OF THE SIMILARITY FUNCTION MATRIX FOR A CF ALGORITHM BASED ON USER SIMILARITY

user_id	1	2	3	4	5	6	7
user_id							
1	1.000000	0.160841	NaN	NaN	0.420809	0.295410	0.258137
2	0.160841	1.000000	NaN	NaN	NaN	0.446966	0.643675
3	NaN	NaN	1.0000	-0.2626	NaN	-0.109109	0.064803
4	NaN	NaN	-0.2626	1.0000	NaN	NaN	-0.266632
5	0.420809	NaN	NaN	NaN	1.000000	0.241817	0.175630

Depending on the type of algorithms, the coefficient was calculated for object or users. Tab. 2 shows a part of the similarity matrix for a CF (*Collaborative Filtering*) algorithm based on user similarity.

For the algorithm based on similarity of objects, the analysis and prediction values were carried out, assuming different values of the minimum number of common assessments between the objects under consideration and a different number of objects in the neighbourhood. Due to the computational complexity and the significant length of the simulation, the following values of the minimum number of evaluations were assumed: 150, 100, 50, 25, 15, 10, 5. However, the analysed quantities of objects in the neighbourhood were assumed as follows: 1, 2, 3, 4, 5, 10, 15, 20.

For the MovieLens 1M Dataset database, due to operation on matrices with much larger dimensions, the value of the minimum number of ratings was assumed: 50, 20, 5.

As part of the analysis of the calculated prediction values, for the given data set, four values were examined: the RMSE value, the MAE value, the number of recommendations and the execution time of the script. All of them were presented depending on the number of items in the neighborhood. The number of recommendations was presented as a percentage of all grades that were included in the analyzed database.

Having matrices describing the relations: user - object and values of the similarity function, one had to start calculating the prediction values. In the case of a CF algorithm based on user similarity, it was used (2) and (3) [4,5].

$$p_{u,i} = \bar{r}_u + \frac{\sum_{a=1}^{n} s(u,a)\left(r_{a,i} - \bar{r}_a\right)}{\sum_{a=1}^{n} |s(u,a)|} \tag{2}$$

where:
$p_{u,i}$ - prediction value for the u user's assessment of the object i
$\bar{r}_u$ - the average of all ratings issued by the user u
$\bar{r}_a$ - average of all ratings issued by user a, user's u neighbour
$r_{a,i}$ - rating of the object i by the user a
$s(u,a)$ – value of the similarity function between the user u and a
n - number of the nearest neighbours of the user u

$$p_{u,i} = \bar{r}_u + \sigma_i \frac{\sum_{a=1}^{n} s(u,a)\left(r_{a,i} - \bar{r}_a\right)/\sigma_a}{\sum_{a=1}^{n} |s(u,a)|} \tag{3}$$

where:
σ_i - standard deviation of ratings issued by the user i
σ_a - standard deviation of ratings issued by the user a

$$p_{u,i} = \frac{\sum_{j \in s_i} r_{u,j} s(i,j)}{\sum_{j \in s_i} |s(i,j)|} \tag{4}$$

where:

$r_{u,j}$, - the rating issued by the user u for the object j

$s(i,j)$ – the value of the similarity function between objects i and j

s_i - it is a collection of objects similar to the object i, usually it is k the most similar items, which were also evaluated by the user u.

$$MAE = \frac{1}{n} \sum_{k=1}^{n} |p_k - r_k| \tag{5}$$

where:
n – the number of all ratings
r_k – real rating issued by the user
p_k - calculated by the system rating prediction

$$RMSE = \sqrt{\frac{1}{n} \sum_{k=1}^{n} (p_k - r_k)^2} \tag{6}$$

However, for an algorithm based on similarity of objects, it was used (4), without taking into account items for which the similarity function values $(i,)<0$.

After the calculation of the prediction, its value was compared with the real estimate issued for a given item by the user under consideration. Then MAE and RMSE errors were calculated, according to (5) and (6). Within the script, the length of its operation and the number of predictions obtained were also recorded.

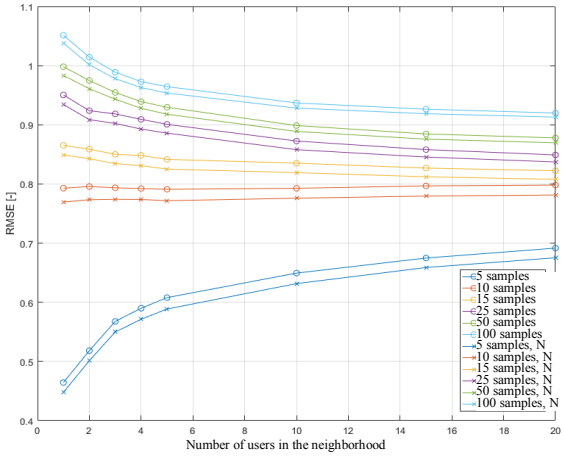

Fig. 1. The dependence of RMSE on the number of users in the neighborhood

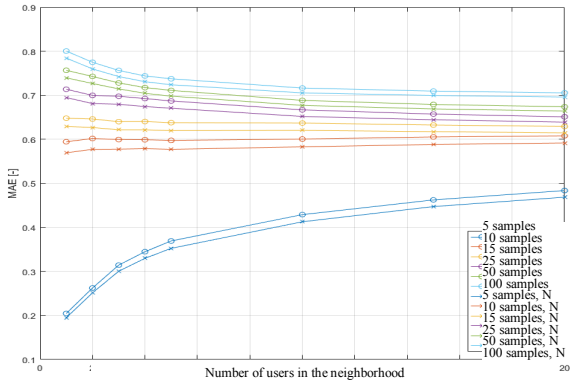

Fig. 2. The dependence of MAE on the number of users in the neighborhood

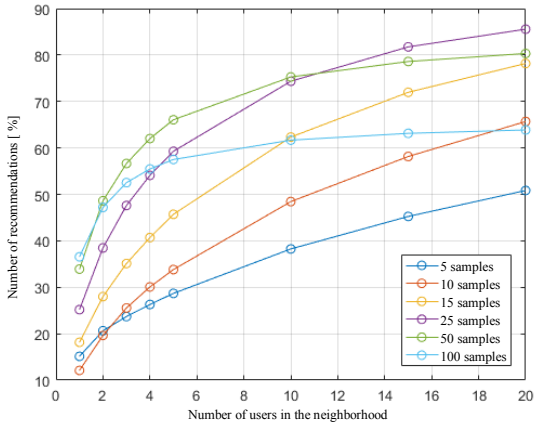

Fig. 3. Dependence of the number of recommendations on the number of users in the neighborhood

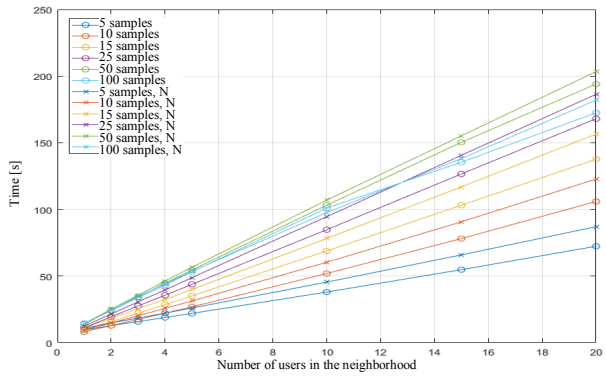

Fig. 4. The dependence of the script's duration on the number of users in the neighborhood

V. CONCLUSION

The article presents issues related to the application of machine learning in recommendation systems. Then, algorithms calculating the prediction values of ratings for users of the movie portal were implemented.

Recommendation systems are currently a very common tool used for marketing purposes. The work describes only some of the techniques used to build recommendations. The focus is on neighbourhood based algorithms and content, because they are the most widely used methods of creating commands. By using public user databases, an attempt was made to create algorithms that calculate the prediction rating.

The algorithm based on similarity of users and the similarity of objects was used for this task. For the analysed data sets, better results were obtained using the method based on user similarity. The errors considered - RMSE and IEA, for this kind of algorithm were smaller than in the case of an approach based on similar objects. What's more, this algorithm allowed more recommendations in a shorter time. On the basis of these studies, however, it cannot be said that the approach based on similar users is superior in all systems. In the analysed case, the data sets were guaranteed a certain number of minimum ratings issued by users. In fact, this situation is quite rare, and sometimes you have to create a recommendation for users who have issued only a few ratings. Further work on the issue of the recommendation system may include an attempt to implement hybrid methods to reduce errors occurring when calculating prediction values.

ACKNOWLEDGMENT

Scientific research was financed from AGH, WIMiR Statute Work: 11.11.130.766.

REFERENCES

[1] T. Segaran, „ *Nowe usługi 2.0. Przewodnik po analizie zbiorów danych*”, Gliwice, Wydawnictwo Helion, 2014 r.

[2] M. Mohri, A. Rostamizadeh, A. Talwalkar *„Foundations of Machine Learning”,* Cambridge, London, The MIT Press, 2012 r.

[3] G. Adomavicius, A. Tuzhilin „*Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions*”.http://trouvus.com/wp-content/uploads/2016/03/recommender-systems-survey-2005.pdf

[4] M. Montaner, B. Lopez, J. L De la Rosa. „*A Taxonomy of Recommender Agents on the Internet”.* https://pdfs.semanticscholar.org/f381/f58e6921a372ecf5740fd9394ec6bfa145c8.pdf

[5] How Reddit algorithms work. Dostępny: https://medium.com/hacking-and-gonzo/how-reddit-ranking-algorithms-work-ef111e33d0d9

[6] B. Sarvar, G. Karypis, J. A. Konstan, J. Riedl „*Item-Based Collaborative Filtering Recommendation Algorithms*