

User Weighting Affect in Neighbourhood based Collaborative Filtering using Firefly Algorithm

User Weighting Affect in Neighbourhood based Collaborative Filtering using Firefly Algorithm

8 Hartatik
Faculty of Computer Science
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia
hartatik@amikom.ac.id

Yahya Sugandi.Y
Faculty of Computer Science
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia
yahya.17@students.amikom.ac.id

Andri Sya35nto
Department of Informatics
STMIK El Rahma
Yogyakarta, Indonesia
andrisyafrianto@gmail.com

Abstract— The Recommendation System is a system that can provide content that is likely to be chosen by users because the content provided will be based on filtering information that takes preferences from user behavior and history. Recently, the researchers researched to improve the quality of one of the recommendations by using swarm intelligence in the traditional recommendation system collaborative filtering. This study aims to determine the effect of user weighting on a traditional recommendation system, one of the swarm intelligence, namely firefly algorithm, to give weight to users and get active users on the dataset. In this study, the firefly algorithm was able to find 100 users who had a significant influence on the prediction results with an MAE error value of 0.8101. However, using all the data can give a lower MAE error value of 0.8007.

Keywords— Recommendation System, Firefly Algorithm, Pearson Correlation Coefficient, Cosine Similarity, User Weighting.

I. INTRODUCTION (HEADING 1)

A recommendation system can provide content that users are likely to choose because the content provided will be based on filtering information that takes preferences from user behavior and history. The amount of data on the internet that continues to grow every day makes recommendation systems one of the indispensable requirements for personalizing users.

Several well-known e-commerce companies have used recommendation systems in their systems, including YouTube, Amazon, Alibaba, Facebook, and others. The purpose of making a recommendation system is to connect customers with the goods to be purchased so that later it is expected that sales of each product can increase or save costs [1]. Not only that, but the Recommendation System is also the basis for users to perform searches such as movies, songs, restaurants, and other products [2].

Netflix is trying to grow its business enormously to become a market leader in the film provider market. The recommendation system is the core of this business. With the development of a recommendation system, Netflix can be saving over \$ 1 billion annually [3].

Two methods are used to build a recommendation system, namely the Collaborative Filtering and Content-Based methods [1]. To date, Collaborative Filtering is the most successful method and is widely used in recommendation systems [4]. In its application, the collaborative filtering method requires similarity measurement. The similarity measurement in recommendation systems is a statistical measure of how two users or items are related. Several traditional similarity techniques such as Pearson's Correlation (COR), Cosine Similarity (COS), Mean Squared Difference (MSD), and Jaccard Coefficient can be used to measure similarity [5].

However, this technique has several shortcomings, namely Cold Start, Sparsity, Scalability [6].

Several researchers have conducted experiments to overcome these problems and improve the quality of the Recommendation System, one of which is the research conducted by Guibing Guo using the TrustSVD technique to reduce the decline in the quality of the Recommendation System caused by sparsity and cold start [7]. Researchers often use metaheuristic Swarm Intelligence (SI) techniques such as Cuckoo Search, Artificial Bee Colony (ABC), Particle Swarm Optimization, and Bat Algorithm (BA) to find optimal solutions [8]. Furthermore, many also use swarm intelligence combined with traditional recommendation systems and show a considerable improvement in providing recommendations [9].

In this study, the authors propose a study to determine the effect of user weighting on traditional recommendation systems using the Metaheuristic Swarm Intelligence (SI) technique, namely Firefly Algorithm (FA).

II. RECOMMENDATION SYSTEM

Recommendation systems are software tools and techniques that provide suggestions for items that the user will find helpful, such as what items to buy, what music to listen to, or what online news to read. Item is a general term used to indicate what the system recommends to the user. Recommendation systems typically focus on specific types of items. According to their design, graphical user interface, and recommendation techniques used to generate recommendations, all are tailored to provide valuable and practical suggestions for specific item types. One example is a book recommendation system that helps users select books to read and items to buy [10].

In the early years of the development of the internet, much research on recommendation systems was carried out to find new approaches to solve a large amount of information available on the internet. The approaches commonly used in recommendation systems are the content-based filtering and collaborative filtering approaches [4].

3

A. Collaborative Filtering

Collaborative Filtering is a recommended method that bases its predictions and recommendations on ratings or the behavior of other users on the system. The basic assumption behind this method is that other users' opinions can be selected and aggregated in such a way as to provide suitable

5 predictions based on the preferences of other active users. Intuitively, they assume that if a user agrees about the quality and relevance of an item, then that user is likely to agree about another item. If a group of users likes the same thing, for example, if a group of users likes the same things as Mary, Mary tends to like the things they like [11].

There are two approaches in CF, namely model-based filtering (model-BF) and memory-based filtering (memory-BF) [12]. The model-BF works by calculating the time between counting various items and then selecting the user with the most similar neighbors from the active users. The memory-based method or better known as neighborhood-based, is the earliest method developed for collaborative filtering. Neighborhood methods calculate how similar users or items are in the rating matrix. A similarity score or proximity measure is used to identify users and similar items. Commonly used similarity measures are the Cosine similarity and Pearson correlation coefficient. The general approach to neighborhood methods consists of two steps to predict ratings, namely [1]:

1. Looking for the similarity between items (item-based) or similarity between users (user-based) who have given a rating.
2. Provide a rating of user ratings or similar items.

Collaborative Filtering can be defined in two ways, namely User Based Collaborative Filtering (UB-CF) and Item Based Collaborative Filtering (IB-CF) [12].

UB-CF is designed to find similarity from users who have a similar rating pattern to other users and who have rated the item in question. For example, if Alice and Bob have rated films the same way in the past, then other users can use the ratings that Alice observed in the Terminator films to predict the ratings for Bob, who did not rate the film. In general, the user who is most similar to Bob can make a ranking prediction for Bob. The similarity function calculated between rating rows to find similar users [12]. Several methods can be used to calculate the similarity between users, such as the Pearson Correlation Coefficient and Cosine Similarity [13].

B. Pearson Correlation Coefficient

Pearson Correlation Coefficient (PCC) is a correlation search method developed by Karl Pearson. Meanwhile, correlation is a measurement technique that determines how close the relationship between two variables is. The measurement results of the PCC can be either positive or negative. A positive relationship shows that the two variables have a parallel (linear) increase in value. Meanwhile, a negative relationship shows that the two variables have a parallel (linear) decrease in value. The parallel is the increase or decrease in value that follows between two variables [14]. The Pearson Correlation Coefficient method can be calculated using the equation 1:

$$7 \quad sim(u, v) = \frac{\sum_{i \in I(u,v)} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I(u,v)} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I(u,v)} (r_{v,i} - \bar{r}_v)^2}} \quad (1)$$

$sim(u, v)$ is the similarity value between user u and user v . $r_{u,i}$ dan $r_{v,i}$ is the value of user u and user v with respect to item i . \bar{r}_u dan \bar{r}_v is the average value of user u and user v against the item.

C. Cosine Similarity

Cosine Similarity measures the angle between two measured vectors where the smaller angle indicates more remarkable similarity and the higher angle indicates the lower similarity. The cosine angle is 0 to 1, where a higher value indicates the closest similarity between users u and v [15]. The Cosine Similarity method can be calculated using equation 2:

$$2 \quad sim(u, v)^{cos} = \frac{\sum_{i \in I(u,v)} R(u,i) \cdot R(v,i)}{\sqrt{\sum_{i \in I(u,v)} R(u,i)^2} \cdot \sqrt{\sum_{i \in I(u,v)} R(v,i)^2}} \quad (2)$$

Where $sim(u, v)$ is the similarity value between user u and user v . $R(u, i)$ is the rating of item i given by user u . $R(v, i)$ is the rating of item i given by user v . And, $I(u, v)$ is the number of jointly rated items from users u and v .

D. Firefly Algorithm

Swarm intelligence is an evolutionary model based on social behavior and inspired by nature. Xin-She Yang developed the Firefly Algorithm (FA) at Cambridge University in 2007 [16]. FA is used to solve optimization problems. Fireflies produce short, rhythmic lights whose light patterns differ from one another. In the FA algorithm, fireflies are compared to one another. The less attractive firefly move towards the more attractive fireflies. Simply put, a firefly's appeal is directly proportional to the light intensity of the fireflies that are nearby. The equation of variation of attractiveness β with distance r is defined in equation 3 [17].

$$\beta = \beta_0 \cdot e^{-\gamma \cdot r^2} \quad (3)$$

Where β_0 is the value of firefly attraction when $r = 0$ and is γ light absorption coefficient.

The displacement of the firefly i , which is attracted to the firefly j (which is lighter), is determined by equation 4 [8]:

$$x_i = x_i + \beta_0 \cdot e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \cdot (rand - 0.5) \quad (4)$$

x_i is the position of the firefly i . $\beta = \beta_0 \cdot e^{-\gamma r^2}$ is the variation of attractiveness of fireflies i and j . $x_j - x_i$ is the difference between fireflies j and i . α is a random number between [0,1]. $rand$ is a random number generator between [0,1].

III. RESEARCH METHODOLOGY

A. Research Pipeline

The method proposed in this study consists of several stages. The first stage is the retrieval of the MovieLens 1M dataset on the grouplens.org site. The next step is to carry out the preprocessing process to eliminate empty user data. The next step is to transform the data into utility matrix to find similarities between users using the Pearson Correlation Coefficient (PCC). The results of the similarity matrix are

optimized using the Firefly Algorithm to get the weight of each user. After the weights are obtained, the data is divided into training data and test data. K-Nearest Neighbor (KNN) is used to generate rating predictions. The results are evaluated using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). The stage of this research can be seen in the form of a pipeline in Figure 1:

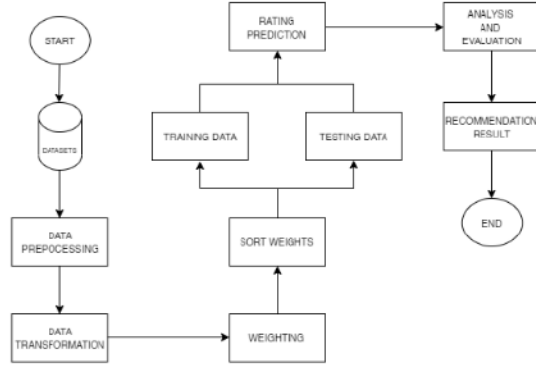


Fig. 1. Pipeline of the research

B. Prediction

There are various methods for making predictions, one of which is K-Nearest Neighbor. KNN tries to predict the rating based on the nearest neighbor who has rated the item. Here is equation number 5 for predicting items to users:

$$\hat{r}_{ui} = \frac{\sum_{v \in N_i^k(u)} \text{sim}(u,v) r_{vi}}{\sum_{v \in N_i^k(u)} \text{sim}(u,v)} \quad (5)$$

\hat{r}_{ui} is rating from user u to item i . \hat{r}_{vi} is rating from user v to item i . N is number of users. k is number of neighbors. $\text{sim}(u,v)$ is similarity between user u and user v .

C. Evaluation Metric

Several metrics are proposed to evaluate the effectiveness of a recommendation on collaborative filtering, such as Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) [14]. MAE is a method that measures the average difference between the predicted rating and the actual rating. The higher the MAE value, it indicates that the system does not predict well. The MAE formula is written in equation 6.

$$MAE = \frac{\sum_{i=1}^n |p_i - a_i|}{n} \quad (6)$$

It is different from RMSE, which measures the accuracy of the prediction results by squaring the error value and dividing it by the amount of data. This value is then rooted. We can see the formula for calculating RMSE in equation 7.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n |p_i - a_i|^2}{n}} \quad (7)$$

n is the total of all rating data calculated by RMSE. a_i is the actual rating given by the user. And, p_i is the rating predicted by the recommendation system.

IV. RESULT AND DISCUSSION

This study uses available dataset movies on the site <https://grouplens.org/datasets/movielens/1m/>. In the dataset, there are 1,000,209 rating data on movies given by 6040 users on 3883 films. The rating provided by the user is a number on a scale of 1 to 5 for each item or film. Table 1 describes the distribution of the data used:

TABEL 1: DATA DISTRIBUTION

Dataset	Distribution		
	User	Item	Rating
MovieLens 1M	6040	3883	1.000.209

Experiments were carried out using three schemes: observing the number of users, observing the amount of training data, and observing using item-based or user-based.

Observations on the number of users were carried out to see if there was an effect on the number of active users with the resulting accuracy. The accuracy measurement value used is the average value of MAE and RMSE from cross-validation results using a different number of users in each experiment, namely 100, 200, 300, and 6040 users.

The Firefly algorithm generates active users from the dataset, where each user will have their weight value in the first scheme. This weight value is then added to the user similarity calculation process by the Pearson correlation coefficient and Cosine similarity method.

Figures 2. and 3. show the results of calculating the MAE and RMSE values in the first scheme. The number of users is 200 and 300, resulting in MAE and RMSE values that are not significantly different. However, for 100 users, the addition of the firefly method on PCC and cosine resulted in better MAE values, namely 0.8252 and 0.8101. Likewise, the RMSE value that uses the addition of the firefly method on active user searches produces better values, namely 1.0482, 1.0301. The resulting weighting for the first 100 users can reduce the MAE value by 0.07 and RMSE by 0.09. However, if you look closely, you can see that the MAE and RMSE values decrease, which are consistent in the PCC and COSINE methods with the increasing number of users. This consistency is not seen in the method that uses the Firefly method for active users, where the lowest error value is 0.79 when using all users in the dataset.

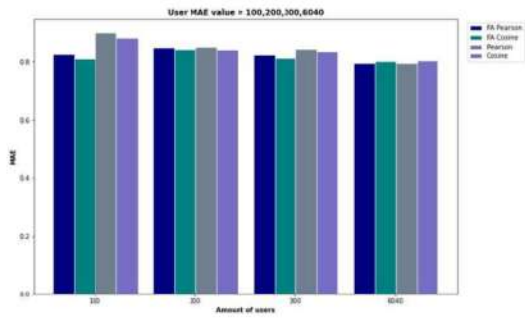


Fig. 2. Comparison of MAE values of 100, 200, 300, 6040 users.

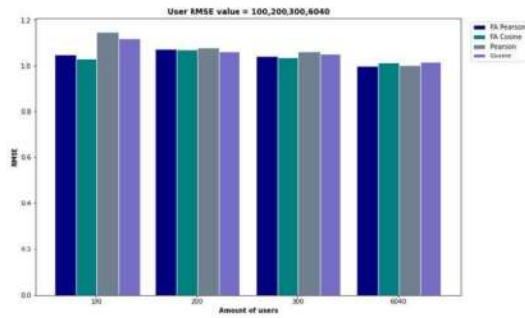


Fig. 3. Comparison of RMSE values of 100, 200, 300, 6040 users.

The KNN method requires the nearest neighbor value to calculate the similarity value search process. The experiment was carried out by comparing the error values in the number of different nearest neighbors, namely $k = 10, 20, 30, 40,$ and 50 . The results of this experiment can be seen in Figures 5 and 6. The error value obtained in the experiment shows that the number of k affects the data a lot. Tests using the number of users 100, 200, 300 do not offer many different effects.

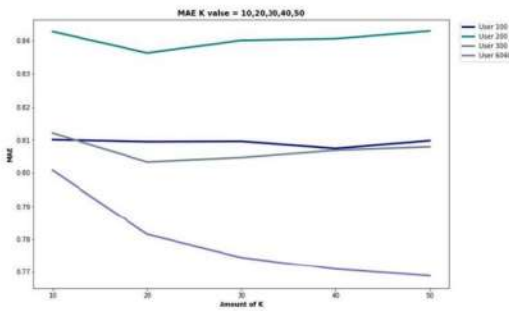


Fig. 5. Comparison of MAE values for $k = 10, 20, 30, 40, 50$

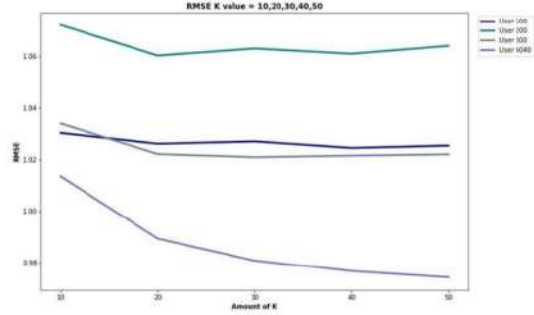


Fig. 6. Comparison of RMSE values for $k = 10, 20, 30, 40, 50$

The second testing scheme is carried out to determine the effect of the amount of training data on the resulting recommendations. Using the number of users as many as 6040, the error value is not much different at the percentage of training data 80% and 90%.

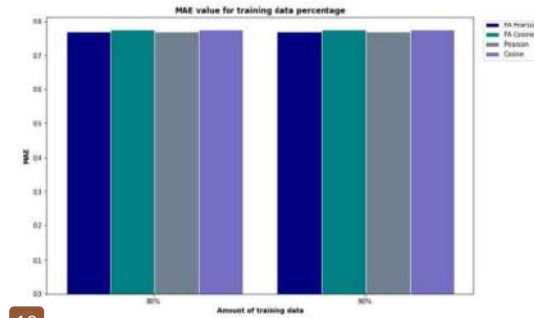


Fig. 8. Comparison of MAE values from the amount of training data

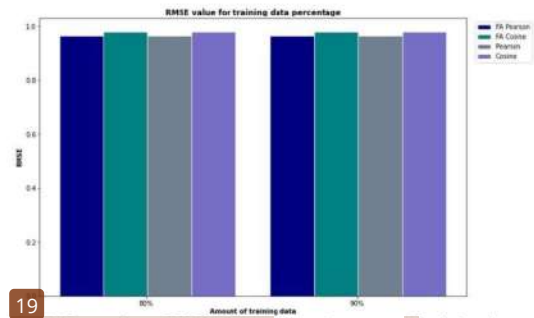


Fig. 9. Comparison of RMSE values from the amount of training data

We can look for similarities between users (User-based) or between items (Item-based) in the Collaborative Filtering method. The third test scheme is carried out to determine whether there is an influence from the use of user-based or item-based. Figures 11 and 12 show the error values obtained from the experiments carried out. The use of user-based collaborative filtering resulted in lower MAE and RMSE error values than item-based. The difference in MAE values is 0.04, while for the RMSE, there is a difference of 0.01.

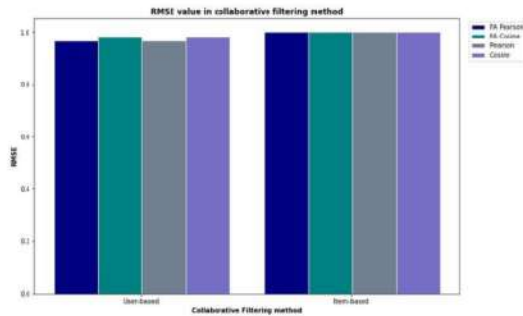


Fig. 11. comparison of RMSE values for FA and traditional methods in user-based and item-based collaborative filtering

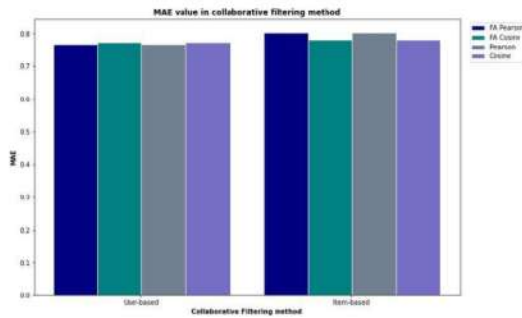


Fig. 12. comparison of MAE values for FA and traditional methods in user-based and item-based collaborative filtering

V. CONCLUSION

Firefly algorithm successfully weights active users. The difference between the MAE and RMSE error values with experiments of 100, 200, 300, and 6040 users did not produce a significant difference with the traditional method. The number of K in KNN does not affect the calculation of MAE and RMSE errors in small data. Data with 100, 200, 300 users does not show much different error values. But when using 6040 users or all data, the number of K affects the calculation of MAE and RMSE errors. By experiment using the number of K = 30, which gives a reasonable high effect. The experimental results on the distribution of training and testing data do affect the calculation of MAE and RMSE errors. The use of User-based and Item-based Collaborative Filtering

can affect the analysis of MAE and RMSE errors. From the tests carried out, User-based produces a lower error value than Item-based. The difference in MAE error reaches 0.04.

REFERENCES

- [1] V. Kotu and B. Deshpande, *Recommendation Engines*. 2019.
- [2] "(SpringerBriefs in Electrical and Computer Engineering) Alexander Felfernig,Ludovico Boratto,Martin Stettinger,Marko Tkalčič (auth.) - Group Recommender Systems _ An Introduction-Springer Internation.pdf."
- [3] C. A. Gomez-Urbe and N. Hunt, "The netflix recommender system: Algorithms, business value, and innovation," *ACM Trans. Manag. Inf. Syst.*, vol. 51, no. 4, 2015, doi: 10.1145/2843948.
- [4] Suryakant and T. Mahara, "A New Similarity Measure Based on Mean Measure of Divergence for Collaborative Filtering in Sparse Environment," *Procedia Comput. Sci.*, vol. 89, pp. 450–456, 2016, doi: 10.1016/j.procs.2016.06.099.
- [5] N. Polatidis and C. K. Georgiadis, "A multi-level collaborative filtering method that improves recommendations," *Experi Syst. Appl.*, vol. 48, pp. 100–110, 2016, doi: 10.1016/j.eswa.2015.11.023.
- [6] S. Yadav, V. Kumar, S. Sinha, and S. Nagpal, "Trust aware recommender system using swarm intelligence," *J. Comput. Sci.*, vol. 28, pp. 180–192, 2018, doi: 10.1016/j.jocs.2018.09.007.
- [7] G. Guo, J. Zhang, and N. Yorke-Smith, "A Novel Recommendation Model Regularized with User Trust and Item Ratings," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 7, pp. 1607–1620, 2016, doi: 10.1109/TKDE.2016.2528249.
- [8] F. Snyman, "Solving Constrained Multi-objective Optimization Problems," *Int. Conf. Swarm Intell.*, vol. 1, pp. 57–66, 2017, doi: 10.1007/978-3-319-61833-3.
- [9] S. Yadav, Vikesh, Shreyam, and S. Nagpal, "An Improved Collaborative Filtering Based Recommender System using Bat Algorithm," *Procedia Comput. Sci.*, vol. 132, pp. 1795–1803, 2018, doi: 10.1016/j.procs.2018.05.155.
- [10] B. S. Mózo, *游無No Title No Title*, vol. 53, no. 9, 2017.
- [11] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, "Collaborative filtering recommender systems," *Found. Trends Human-Computer Interact.*, vol. 4, no. 2, pp. 81–173, 2010, doi: 10.1561/1100000009.
- [12] P. Mertens, *Recommender Systems*, vol. 39, no. 4, 1997.
- [13] Z. Tan and L. He, "An Efficient Similarity Measure for User-Based Collaborative Filtering Recommender Systems Inspired by the Physical Resonance Principle," *IEEE Access*, vol. 5, pp. 27211–27228, 2017, doi: 10.1109/ACCESS.2017.2778424.
- [14] S. Bag, S. K. Kumar, and M. K. Tiwari, "An efficient recommendation generation using relevant Jaccard similarity," *Inf. Sci. (Ny)*, vol. 483, no. January, pp. 53–64, 2019, doi: 10.1016/j.ins.2019.01.023.
- [15] F. Cacheda, V. Carneiro, D. Fernández, and V.

Formoso, "Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems," *ACM Trans. Web*, vol. 5, no. 1, 2011, doi: 10.1145/1921591.1921593.

- [16] X. Yang, *Nature-Inspired Metaheuristic Algorithms Second Edition*, vol. 4, no. C. 2010.
- [17] N. F. Johari, A. M. Zain, N. H. Mustaffa, and A. Udin, "Firefly algorithm for optimization problem," *Appl. Mech. Mater.*, vol. 421, no. April 2014, pp. 512–517, 2013, doi: 10.4028/www.scientific.net/AMM.421.512.

User Weighting Affect in Neighbourhood based Collaborative Filtering using Firefly Algorithm

ORIGINALITY REPORT

26%

SIMILARITY INDEX

PRIMARY SOURCES

- 1** Vijay Kotu, Bala Deshpande. "Recommendation Engines", Elsevier BV, 2019
Crossref 60 words — 2%
- 2** link.springer.com
Internet 44 words — 1%
- 3** "Computer Science and Engineering—Theory and Applications", Springer Science and Business Media LLC, 2018
Crossref 43 words — 1%
- 4** eudl.eu
Internet 39 words — 1%
- 5** ethesis.nitrkl.ac.in
Internet 36 words — 1%
- 6** Sara Ouaftouh, Ahmed Zellou, Ali Idri. "CARMS: Clustering based approach for recommendation in mediation systems", 2017 International Conference on Innovative and Creative Information Technology (ICITech), 2017
Crossref 29 words — 1%
- 7** Liang Zhang. "A scalable collaborative filtering algorithm based on localized preference", 2008 27 words — 1%

8 Krisnawati, Mardhiya Hayaty, Bayu Setiaji, Arief Setyanto. "First Time User Experience Assessment on Web based Online Examination", 2019 International Conference on Information and Communications Technology (ICOIACT), 2019 25 words — 1%

Crossref

9 Sujoy Bag, Abhijeet Ghadge, Manoj Kumar Tiwari. "An integrated recommender system for improved accuracy and aggregate diversity", Computers & Industrial Engineering, 2019 24 words — 1%

Crossref

10 Akash P Lukose, Abyson Scaria, Babu George. "A Firefly Algorithm Approach for Multirow Facility Layout Problem", IOP Conference Series: Materials Science and Engineering, 2018 23 words — 1%

Crossref

11 Phongsavanh Phorasim, Lasheng Yu. "Movies recommendation system using collaborative filtering and k-means", International Journal of Advanced Computer Research, 2017 19 words — 1%

Crossref

12 Fuzhi Zhang, Long Bai, Feng Gao. "Chapter 32 A User Trust-Based Collaborative Filtering Recommendation Algorithm", Springer Science and Business Media LLC, 2009 18 words — 1%

Crossref

13 analyticsclassroom.blogspot.com 18 words — 1%

Internet

14 sciencepubco.com

17 words — 1%

15 K. Umamaheswari, Subbiah Janakiraman, K. Chandrababha. "Multilevel Hybrid Firefly-Based Bayesian Classifier for Intrusion Detection in Huge Imbalanced Data", Journal of Testing and Evaluation, 2021
Crossref

16 Sambhav Yadav, Vikesh Kumar, Shreyam Sinha, Sushama Nagpal. "Trust aware recommender system using swarm intelligence", Journal of Computational Science, 2018
Crossref

17 "Encyclopedia of Social Network Analysis and Mining", Springer Science and Business Media LLC, 2018
Crossref

18 "Proceedings of the 7th International Conference on Emerging Databases", Springer Science and Business Media LLC, 2018
Crossref

19 Chunxue Wu, Jing Wu, Chong Luo, Qunhui Wu, Cong Liu, Yan Wu, Fan Yang. "Recommendation algorithm based on user score probability and project type", EURASIP Journal on Wireless Communications and Networking, 2019
Crossref

20 Md Shafiullah, Muhammad Ijaz, M. A. Abido, Z. Al-Hamouz. "Optimized support vector machine & wavelet transform for distribution grid fault location", 2017 11th IEEE International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG), 2017
Crossref

-
- 21 ZhiKui He, Wenjie Liang, Wenjia Xu, WenXiu Huang, XiaoMin Wang, KaiYong Huang, Li Yang. "Mapping the FACT-G to EQ-5D-3L utility index in cancer: data from a cross-sectional study in in Guangxi, China", Research Square Platform LLC, 2021
Crossref Posted Content 14 words — < 1%
-
- 22 medium.com
Internet 13 words — < 1%
-
- 23 JinHyung Cho. "Source Credibility Model for Neighbor Selection in Collaborative Web Content Recommendation", Lecture Notes in Computer Science, 2008
Crossref 12 words — < 1%
-
- 24 Ziyang Zhang, Yuhong Liu, Zhigang Jin, Rui Zhang. "Selecting influential and trustworthy neighbors for collaborative filtering recommender systems", 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), 2017
Crossref 12 words — < 1%
-
- 25 jwcn-urasipjournals.springeropen.com
Internet 11 words — < 1%
-
- 26 www.researchgate.net
Internet 11 words — < 1%
-
- 27 AKIHIRO YAMASHITA, HIDENORI KAWAMURA, KEIJI SUZUKI. "ADAPTIVE FUSION METHOD FOR USER-BASED AND ITEM-BASED COLLABORATIVE FILTERING", Advances in Complex Systems, 2011
Crossref 10 words — < 1%
-
- 28 Jianxin Li, Linlin Meng, Zekun Zhu, Xudong Li, Jinpeng Huai, Lu Liu. "chapter 10 CloudRank", IGI 10 words — < 1%

29 cris.brighton.ac.uk 10 words — < 1%
Internet

30 "Soft Computing: Theories and Applications", Springer Science and Business Media LLC, 2020 9 words — < 1%
Crossref

31 HaolrwinMichael R. MaKingLyu. "Effective missing data prediction for collaborative filtering", Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR 07 SIGIR 07, 2007 9 words — < 1%
Crossref

32 Ke Niu, Xiangyu Zhao, Fangfang Li, Ning Li, Xueping Peng, Wei Chen. "UTSP: User-Based Two-Step Recommendation With Popularity Normalization Towards Diversity and Novelty", IEEE Access, 2019 9 words — < 1%
Crossref

33 Zhenhua Tan, Liangliang He. "An Efficient Similarity Measure for User-Based Collaborative Filtering Recommender Systems Inspired by the Physical Resonance Principle", IEEE Access, 2017 9 words — < 1%
Crossref

34 crabwq.github.io 9 words — < 1%
Internet

35 j.mecs-press.net 9 words — < 1%
Internet

36 mafiadoc.com 9 words — < 1%
Internet

-
- 37 www.geographie.uni-muenchen.de Internet 9 words — < 1%
-
- 38 "Hybrid Intelligent Systems", Springer Science and Business Media LLC, 2021 Crossref 8 words — < 1%
-
- 39 Qiang Yang. "EigenRank", Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR 08 SIGIR 08, 2008 Crossref 8 words — < 1%
-
- 40 Qibing Jin, Yue Zhang, Wu Cai, Yuming Zhang. "A New Similarity Computing Model of Collaborative Filtering", IEEE Access, 2020 Crossref 8 words — < 1%
-
- 41 Vijay Verma, Rajesh Kumar Aggarwal. "A New Similarity Measure Based on Simple Matching Coefficient for Improving the Accuracy of Collaborative Recommendations", International Journal of Information Technology and Computer Science, 2019 Crossref 8 words — < 1%
-
- 42 assets.researchsquare.com Internet 8 words — < 1%
-
- 43 doctorpenguin.com Internet 8 words — < 1%
-
- 44 www2.llu.lv Internet 8 words — < 1%
-
- 45 "Advances in Multimedia Information Processing – PCM 2018", Springer Science and Business Media LLC, 2018 7 words — < 1%

-
- 46 "Recommender Systems Handbook", Springer Science and Business Media LLC, 2015
Crossref 7 words — < 1%
-
- 47 "Knowledge Science, Engineering and Management", Springer Nature, 2016
Crossref 6 words — < 1%
-
- 48 M. Sandeep Kumar, Prabhu J.. "Hybrid Model for Movie Recommendation System Using Fireflies and Fuzzy C-Means", International Journal of Web Portals, 2019
Crossref 6 words — < 1%
-
- 49 Patra, Bidyut Kr., Raimo Launonen, Ville Ollikainen, and Sukumar Nandi. "A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data", Knowledge-Based Systems, 2015.
Crossref 6 words — < 1%
-
- 50 Sri Lestari, Teguh Bharata Adji, Adhistya Erna Permanasari. "Performance Comparison of Rank Aggregation Using Borda and Copeland in Recommender System", 2018 International Workshop on Big Data and Information Security (IW BIS), 2018
Crossref 6 words — < 1%
-
- 51 jist.ir
Internet 4 words — < 1%

EXCLUDE QUOTES ON

EXCLUDE BIBLIOGRAPHY ON

EXCLUDE SOURCES OFF

EXCLUDE MATCHES OFF