



Friend Recommendation Engine for Facebook Users via Collaborative Filtering

M. Alshammari, A. Alshammari

Mohammed Alshammari

Department of Computer Science
Faculty of Computing and Information Technology
Northern Border University, Saudi Arabia
Northern Border University, Arar, Alhudud Alshamalia 91431, Saudi Arabia
Corresponding author: mohammed.alshammari@nbu.edu.sa

Aadil Alshammari

Department of Information Systems
Faculty of Computing and Information Technology
Northern Border University, Saudi Arabia
Northern Border University, Arar, Alhudud Alshamalia 91431, Saudi Arabia
aadil.al-shammari@nbu.edu.sa

Abstract

Today's internet consists of an abundant amount of information that makes it difficult for recommendation engines to produce satisfying outputs. This huge stream of unrelated data increases its sparsity, which makes the recommender system's job more challenging. Facebook's main recommendation task is to recommend a friendship connection based on the idea that a friend of a friend is also a friend; however, the majority of recommendations using this approach lead to little to no interaction. We propose a model using the matrix factorization technique that leverages interactions between Facebook users and generates a list of friendship connections that are very likely to be interactive. We tested our model using a real dataset with over 33 million interactions between users. The accuracy of the proposed algorithm is measured using the error rate of the predicted number of interactions between possible friends in comparison to the actual values.

Keywords: Artificial Intelligence, Machine Learning, Recommender Systems, and Social Media.

1 Introduction

The number of people using the internet worldwide has been rapidly increasing over the past two decades, from about 400 million users in 2000 to more than 3.4 billion users in 2016 [1]. The increasing number of online users correlates with the growing amount of online data, especially in the field of social media, where most of the data is created by users. This makes finding the right information very challenging. According to Statista, the global digital population in 2022 consisted of five billion users, with 93% of them being social media users [2].

Over the past decade, social media giants such as Facebook, Twitter, and Instagram have attracted many users around the world, making these platforms convenient places to find and start new relationships. Just like in real life, where friendships are established through real interactions in a workplace, school, and travel, on these virtual platforms, interactions are different; they may come in the form of a like or dislike, a picture comment, or a private message, among various other means. Therefore, the recommendation engine task is essential for social networks, as they process this enormous amount of data with diverse types of interactions to generate valid recommendations. Online social media users find and connect with already known friends, as well as new friends, to interact with them. Friendship recommendation algorithms based on collaborative and content-based filtering have been thoroughly explored [3, 4, 5]. However, most of these algorithms focus on recommending relationship connections with a high probability of being accepted, but with a low probability of being interactive.

Table 1: Interaction Rate of our Collected Dataset on Facebook

Interaction rate with friends using comments	2.45%
Interaction rate with friends using likes	13.23%

Table 1 shows that, on average, a user only interacts with 2.45% of his or her declared friends using comments and 13.23% using likes. In this paper, we design, develop, and implement a novel collaborative friendship recommendation algorithm that generates easily accepted friendship connections with a very high probability of leading to interactions. The remainder of this paper is organized as follows:

1.1 Research Design

First, we will give an introduction to the topic, show its importance in the related research fields, and state the research question that the paper tries to answer. Second, a thorough literature review will be provided, showing the history of the topic and how related research has proposed solutions to problems similar to ours. The third part will focus on the methodology used to solve the issue encountered and the data collection process. The fourth part is designed to discuss the evaluation process, which includes the experimental setting, data cleaning, parameter tuning, and related steps. In the last part, some overall insights will be provided to conclude the work.

2 Literature Review

2.1 Social Networks & Recommendation Algorithms

Facebook has dominated the social network industry for the past decade. In 2022, Facebook was reported as the most popular social network worldwide, with more than 2.9 billion monthly active users [6]. The giant social network uses its well-known Friends-of-Friends (FoF) algorithm. This friendship recommender algorithm is one of the key factors that plays a major role in the growth of the social network. Facebook's FoF algorithm has been largely successful in increasing the size of the network by recommending easily accepted friendship connections among its users. However, the percentage of interactive connections recommended by the FoF algorithm is reportedly very low [7, 8, 9, 10]. The strength of the FoF algorithm is that it finds already-known people, which explains its high acceptance rate. However, an accepted relationship recommendation does not necessarily lead to interaction. Therefore, further filtering is required to find interactive friendship connections among friends of friends.

Recommender systems come in mutable forms, the most popular of which are content-based and collaborative filtering [11]. Other types include community-based, demographic-based, knowledge-based, and hybrid filtering [11]. Content-based research focuses on building recommendations based on the content of the system components. For example, if a user likes certain types of products (e.g., football or sports shirts), the system will try to recommend items that are similar to those he has liked, bought, or rated in the past; in this case, sports sneakers might be a good fit [12]. Collaborative

filtering techniques rely on building recommendations for the target user by looking at items that similar users have previously explored, or items that are similar to the items that the target user has interacted with in the past [13, 14].

A recommender system model aims to drive the success of a social network (e.g., Netflix, Amazon, YouTube, Facebook, or Booking) by increasing its traffic. This leads to more content (e.g., movies, products, posts, hotels, or restaurants) and more interactions (e.g., likes, dislikes, comments, or messages), which eventually leads to more profits (e.g., watching ads, buying items, or subscribing). Collaborative recommendation algorithms have been successfully used to filter online items, such as products on Amazon or movies on Netflix, but they have not found as much success in filtering and recommending friends on social networks. Alshammari & Rezgui developed friendship recommender algorithms to generate interactive relationships [7, 10], which more than doubled the recommended interactive edges compared to Facebook's FoF algorithm. In this research, we are presenting a new model to further generate better results.

A disadvantage of content-based recommender systems is that they lack the ability to deal with sparse data. The performance of those models deteriorates when the data is sparse. However, collaborative filtering techniques overcame this obstacle, excelling in recommending better results with a lower error rate.

2.2 Matrix Factorization

The Matrix Factorization (MF) algorithm is a state-of-the-art technique for predicting ratings for unseen items, especially when data are sparse [15]. The accuracy of MF is higher compared to baseline techniques [16, 17, 18]. One of the most powerful features of MF is that it allows for taking advantage of the latent spaces of both users and items. The MF technique has been exploited by many researchers in the field of recommendation engines in recent years [19, 20, 21, 22, 23]. [17] proposed a methodology that extends the initial idea of [15] by introducing context from an extra source to the model. They applied their methodology to the movie domain and used the items' extra information, such as genre and stars, to enhance the performance of the recommendation engine. It is worth mentioning that extra user-side information may not always be available, as privacy issues arise. Nevertheless, they compared their results to baselines and showed that the accuracy of their proposed methods was higher.

Another model using MF in the field of recommender systems is the one proposed by [18]. In this case, the researchers developed [17] their model by including semantic web technology in the process of building the engine. In the domain of movies and music, they exploited the power of DBpedia ¹ to retrieve more meaningful information that also helped produce more reasonable recommendations. The resulting model outperformed other baselines after conducting multiple experiments.

In a related vein, BenAbdallah [24] explored MF and extended it to include item side information (e.g., bag-of-feature) in the latent spaces and succeeded in producing annotations to images. This solution was used to tackle a famous problem in collaborative filtering, which is the cold-start problem [25].

One of the drawbacks of the current state-of-the-art MF models is that they lack transparency. The MF Algorithm is a black box system whose output cannot be easily explained, in contrast to content-based techniques, where generating an explanation is a straightforward process. Another disadvantage is the cold start problem, where the system knows nothing about new users until they rate, purchase, listen to, interact with, etc. with some items.

In this research, we designed, implemented, and tested a friendship recommender algorithm to generate interactive relationship connections that have a higher probability of leading to interactions using MF.

¹dbpedia.org

3 Research Methodology

Friendship connections within a social network can be directed or undirected. This research uses a giant social network, Facebook, as our case study. The relationship edges among users within Facebook are undirected. Figure 1 shows a sub-graph example of an undirectional social network. As seen in the figure, a black edge between two nodes represents an undirected friendship connection. The cyan nodes (A, S, E, and M) are the common friends between user x and user fof . Facebook’s FoF algorithm solely uses common friends to find and recommend friends of friends. The red edges shown in the figure are designed to distinguish between interactive and non-interactive relationships. For instance, user x interacted with user A using likes and/or comments.

Our recommender model is designed to recommend friends of friends that are very likely to be interactive. The main idea behind our model is that both the target user (user x) and the friend-of-friend interacted with the same common friends. The example in Figure 1 shows that L , fof , and N are all friends-of-friends of user x . However, only user fof interacted with similar common friends, which makes user fof a possible interactive friend to user x .

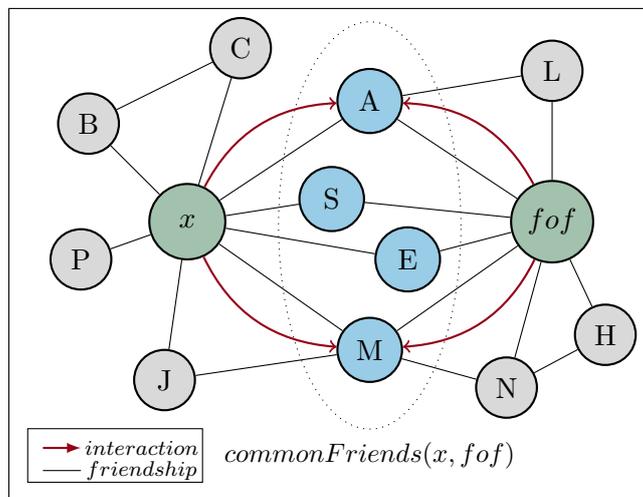


Figure 1: A Social Network’s Sub-Graph Example

In our model, the MF technique is used to build the model, and items are replaced by other users. Hence, by means of users’ interaction history, MF can learn the surfing pattern of such users (e.g., pictures he/she likes or posts he/she comments on) and then suggest similar new friends.

One of the most powerful features of MF is its ability to handle sparse data and produce more accurate recommendations compared to other models [15]. Other techniques within collaborative filtering, explained above in the literature review section, rely mostly on the explicit feedback expressed by users toward items. However, even though MF needs some information about users to lower the percentage of a cold start, it can implicitly observe the users’ patterns toward items and predict how they may like or dislike the unseen items. In our model, items are replaced by other users with whom the target user is not yet a friend with. The goal of the model is to learn implicitly about those users engaging in explicit interactions and to suggest new relationships.

In detail, every user u will be represented in a vector p_u , and every interacted-with user i is associated with a vector q_i . Conducting a dot product of those two vectors, p_u and q_i , will result in predicting \hat{r} , which is the number of interactions between all users in both vectors.

$$\hat{r} = q_x^T p_{fof}; \tag{1}$$

The objective function to be minimized is as follows:

$$J = \sum_{x, fof \in R} (r_{xfof} - p_x q_{fof}^T)^2 + \beta(\| p_x \|^2 + \| q_{fof} \|^2) \tag{2}$$

R is the set of known interactions between x and fof . r_{xfof} represents an interaction between users x and fof . p is the latent space for users, and q denotes the latent space for friends. The second

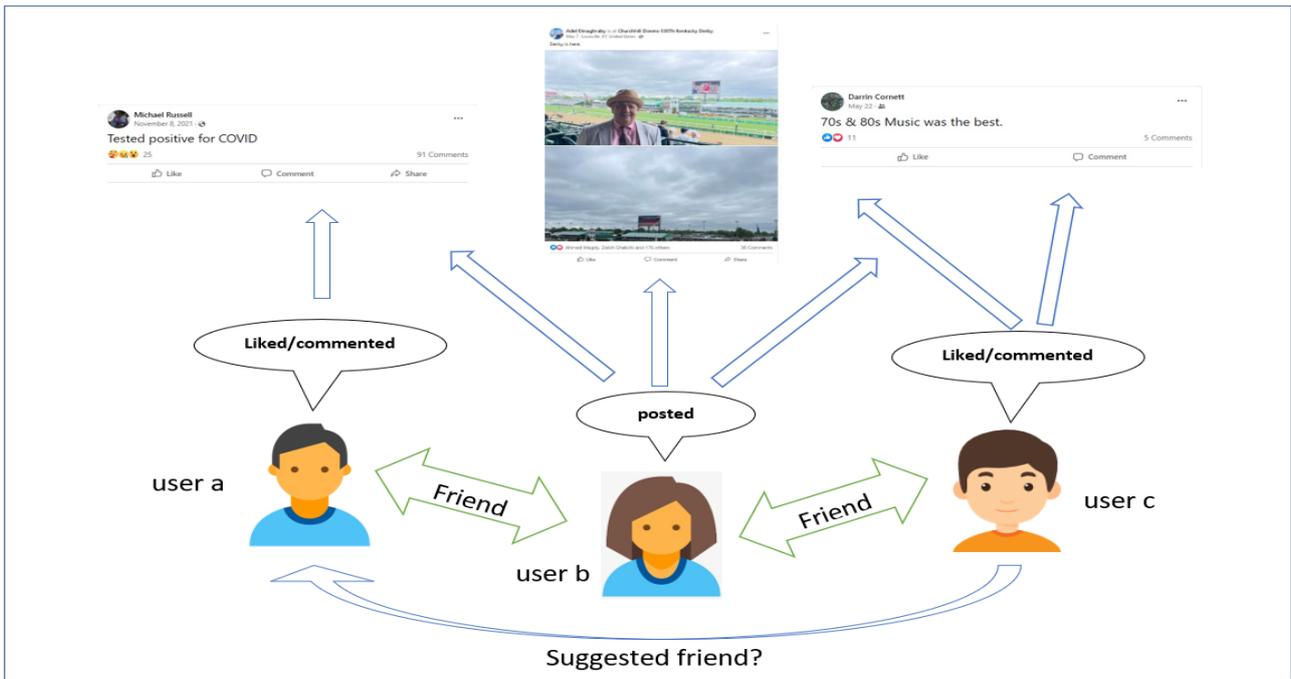


Figure 2: Example of a possible recommendation between interactive Friends-of-Friends

expression is a regularization term that helps prevent overfitting. β is a smoothing coefficient used to control the addition of new items.

The update rules for user p and friend q are as follows:

$$p_x^{(t+1)} \leftarrow p_x^{(t)} + \alpha(2(R_{x,fof} - p_x^{(t)}(q_{fof}^{(t)})^T)q_{fof}^{(t)} - \beta p_x^{(t)}) \quad (3)$$

$$q_{fof}^{(t+1)} \leftarrow q_{fof}^{(t)} + \alpha(2(R_{x,fof} - p_x^{(t)}(q_{fof}^{(t)})^T)p_x^{(t)} - \beta q_{fof}^{(t)}) \quad (4)$$

Figure 2 shows a visualization of the proposed method. As can be seen, if we have a user a that interacts with posts or pictures on Facebook while his or her friend user b also does so, an FoF user c that expresses likability to his or her friend user b might be a good suggestion to be a friend with user a . As shown in the figure, users a and b share interactions (e.g., likes or comments) with the same posts or pictures; moreover, users b and c also share the same interaction with other posts and pictures. Since users a and c are not yet introduced to each other as friends, we can infer that they may like each other to a certain degree based on the fact that they are both friends with user b . This inference can be drawn from the low dimensional latent spaces technique that matrix factorization provides to us.

To accurately test and compare our model, we designed and implemented a web crawler using Python that fetches publicly available users' profiles.

4 Results Analysis

In this section, we will explore the results of our experiments, starting with an analysis of the dataset, presenting the experimental setting, and finishing with an illustration of two kinds of measurement: accuracy and recommendation.

4.1 Dataset

In order to test and validate the results of our approach, we collected publicly available data from Facebook using our web crawler. The web crawler was designed to fetch publicly available users' profiles. We randomly started with seeds from different states in the US. After fetching a user's profile, the web crawler added the fetched user's friends to a queue so that they would be fetched later. For

privacy, we replaced every user ID with a randomly generated ID. For each user profile, we collected the following:

- User ID (replaced with a randomly generated numeric ID)
- Friends list
- Posts
 - Post
 - User IDs who liked the post
 - User IDs who commented on the post
- Interests

We ran our web crawler in 2018 and collected about 16,500 unique user profiles. In this dataset (DS2018), we had more than 15 million interactions (likes and comments). The average percentage of interactively declared friends in our collected Facebook dataset was 13.93%. This finding correlates with what was reported about the low interaction rate among Facebook users. Part of this problem is caused by Facebook’s FoF algorithm, which is solely designed to increase the list of friends but not interactive friends.

4.2 Experimental Setting

In this experiment, the total number of Facebook users was 6,426 and the total number of friends was 8,638. Interactions between the users and their friends reached over 55,000 single interactions. We cleaned the data by eliminating friendships between any two users who had fewer than three interactions in the data set. The reason was to minimize the number of users, which would help accelerate the model building and lessen the noise, as friends with only one or two interactions do not help the model learn about them in the low-dimensional spaces where hidden features are introduced. The train set size was 90%, and the remaining 10% was kept for testing. The total number of interactions was normalized to one. Train and test sets were cross-validated fivefold, and the mean was reported. The MF model hyperparameters were tuned, and the number of patches was set to 20 steps. Low-dimensional matrices p and q were initialized randomly, and the experiments were run 10 times; the mean was then reported to avoid the acquisition of good results by luck. Code implementation can be found in Github ².

The proposed method was evaluated using two different evaluation methodologies. The first consisted of calculating the error rate using Root Mean Square Error (RMSE), and the second involved calculating the accuracy of recommendations using Mean Absolute Precision (MAP) from the confusion matrix.

4.3 Accuracy Measure

The RMSE formula was as follows:

$$RMSE = \sqrt{\frac{1}{|T|} \sum_{(x,fof) \in T} (r'_{xfof} - r_{xfof})^2}. \quad (5)$$

where r'_{xfof} denotes the expected interactions between user x and friend FoF, while r_{xfof} is the true and known value. T is the count of all predictions. The root of the squared result was determined to eliminate any negative numbers.

In our model, the algorithm will predict a number for each friend of the target user in the dataset, representing the expected number of interactions between them. Moreover, since there are 10% of the dataset is hidden, the RMSE will calculate the difference between the true number of interactions

Table 2: RMSE results using a different number for feature K for the train and test

Facebook-MF		
# of features K	Train RMSE	Test RMSE
1	0.0907	0.1873
2	0.1082	0.3112
3	0.1193	0.4318
4	0.126	0.5456
5	0.1293	0.6638

between the target user and all of his or her true friends in the hidden test set and the predicted numbers that the model produced. Table 2 shows the resulting values, known to be lower the better.

Table 2 shows the error rate of the model for both the training and testing datasets. The first column represents the proposed number of hidden features, starting with only one and finishing with five. The results indicate that the model performed better with fewer hidden features, which may be due to the sparsity of the data. Introducing more features to the already poor graph of users and friends led to a worsening performance of the model and caused the learning process to slow down, which, consequently, led to an increase in the error rate. In the opposite situation, where the data graph was rich with connections and information about the users' relations was sufficient, more hidden features helped decrease the error rate. The train RMSE was noticeably less than the test RMSE for all different K because of the number of known interactions between users. As mentioned earlier, the training set was 90%, and the testing set was 10%, which explains the disparity in performance. One of the reasons that such a graph has insufficient data is that many of the users have set their profiles as private. As we mentioned above, we only fetched publicly available users' profiles. If our algorithm was run by Facebook with access to all the data in the graph, the results would have been even more accurate.

4.4 Recommendation Measure

The MAP formula was as follows:

$$MAP@N = \frac{1}{|U|} \sum_{u=1}^U \frac{1}{m} \sum_{n=1}^N P_u(n) \cdot rel_u(n). \quad (6)$$

N represents the desired number to be in the top list, U indicates the users' total number, and m denotes the relevant friends. A ratio of the recommended and relevant friend lists to the whole recommended friend list n was derived and then denoted by P . The last term, $rel_u(n)$ is a binary indicator showing the relevance of n^{th} recommended friend.

Table 3: MAP results using different number of feature K

Facebook-MF	
# of features K	Test MAP@10
1	1
2	1
3	1
4	1
5	1

The first column indicates the number of hidden features. The second column shows the model performance measured using the MAP of the top 10 recommended friends. Using the test set, the

²<https://github.com/drmohammedsanad/FacebookMF>

result was 1, indicating that the model succeeded in recommending at least 1 real interactive friend to the target user in the top 10 recommendations list of possible interactive friends.

In the experiment, the model was not only able to accurately recommend interactive friends to the target user but was also able to predict the possible number of interactions between the two users. We were able to measure the model's accuracy by hiding some of the target user's actual friends, for whom we have records of their interactions. For instance, in the test set, the user with ID 210485 is already a friend of the target user with ID 210619, and the model predicted a possible interactive relationship between them after predicting that they will have 22 interactions, which is only off by around 18% of the actual interactions.

5 Conclusion

Social media platforms have become abundant with data that is mostly sparse and useless unless preprocessed and cleaned to help decision-making algorithms, such as recommendation engines, perform appropriately. In this paper, we explore the power of the MF algorithm for helping Facebook users find new friends by using their sparsely expressed interactions. In this experiment, we limited the number of accepted interactions to three in order to reduce data size, which consequently helped the MF algorithm perform faster and more efficiently.

Experiments show that increasing the number of features results in deteriorating the performance of MF due to the sparsity level expansion of the current data. Assuming that we only knew one hidden feature helped the model perform better than knowing two or more features. The RMSE measure was used to prove this notion. The MAP measure, which shows whether true friends who were hidden in the building process were in fact included in the top 10 recommendations, indicates that the model using one to five hidden features succeeded in this matter. In conclusion, the rapid increase in data size on today's internet requires algorithms that have the ability to cope with these changes. Experiments illustrate that MF is state-of-the-art in this matter. It is worth adding that MF also performs better when extra item-user-side information is introduced. For future work, our model, which recommends new friends using interactions, could be supplied with an explanation feature that will result in increasing trust, efficiency, and user loyalty to the system.

Funding

The authors gratefully acknowledge the approval and the support of the research study by the grant no. CSCR-2022-11-1639 from the Deanship of Scientific Research at Northern Border University, Arar, K.S.A.

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

- [1] Max Roser, Hannah Ritchie, and Esteban Ortiz-Ospina. Internet. *Our World in Data*, 2015. <https://ourworldindata.org/internet>.
- [2] Statista. Global digital population as of April 2022, 08 2022.
- [3] Michael D Ekstrand, John T Riedl, Joseph A Konstan, et al. Collaborative filtering recommender systems. *Foundations and Trends® in Human-Computer Interaction*, 4(2):81–173, 2011.

- [4] J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer, 2007.
- [5] Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. *Recommender systems handbook*, pages 73–105, 2011.
- [6] Statista. Global social networks ranked by number of users 2022, 07 2022.
- [7] Aadil Alshammari and Abdelmounaam Rezgui. Cidf: A clustering-based interaction-driven friending algorithm for the next-generation social networks. *IEEE Access*, 7:153555–153565, 2019.
- [8] Scott A Golder, Dennis M Wilkinson, and Bernardo A Huberman. Rhythms of social interaction: Messaging within a massive online network. In *Communities and technologies 2007*, pages 41–66. Springer, 2007.
- [9] Christo Wilson, Bryce Boe, Alessandra Sala, Krishna PN Puttaswamy, and Ben Y Zhao. User interactions in social networks and their implications. In *Proceedings of the 4th ACM European conference on Computer systems*, pages 205–218, 2009.
- [10] Aadil Alshammari and Abdelmounaam Rezgui. Better edges not bigger graphs: An interaction-driven friendship recommender algorithm for social networks. In *2020 5th International Conference on Cloud Computing and Artificial Intelligence: Technologies and Applications (CloudTech)*, pages 1–8. IEEE, 2020.
- [11] Robin Burke. Hybrid web recommender systems. In *The Adaptive Web*, pages 377–408. Springer Berlin Heidelberg, 2007.
- [12] Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In *Recommender Systems Handbook*, pages 73–105. Springer US, oct 2010.
- [13] David Goldberg, David Nichols, Brian M. Oki, and Douglas Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70, dec 1992.
- [14] J. Ben Schafer, Joseph A. Konstan, and John Riedl. E-commerce recommendation applications. In *Applications of Data Mining to Electronic Commerce*, pages 115–153. Springer US, 2001.
- [15] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, aug 2009.
- [16] Behnoush Abdollahi and Olfa Nasraoui. Explainable matrix factorization for collaborative filtering. In *Proceedings of the 25th International Conference Companion on World Wide Web - WWW '16 Companion*. ACM Press, 2016.
- [17] Yue Shi, Martha Larson, and Alan Hanjalic. Mining contextual movie similarity with matrix factorization for context-aware recommendation. *ACM Transactions on Intelligent Systems and Technology*, 4(1):1–19, jan 2013.
- [18] Nidhi Kushwaha, Shubham Mehrotra, Ronish Kalia, Dhruv Kumar, and O. P. Vyas. Inclusion of semantic and time-variant information using matrix factorization approach for implicit rating of last.fm dataset. *Arabian Journal for Science and Engineering*, 41(12):5077–5092, may 2016.
- [19] Guanzhong Liang, Chuan Sun, Jianing Zhou, Fengji Luo, Junhao Wen, and Xiuhua Li. A general matrix factorization framework for recommender systems in multi-access edge computing network. *Mobile Networks and Applications*, 27(4):1629–1641, 2022.
- [20] Mehdi Hosseinzadeh Aghdam. A novel constrained non-negative matrix factorization method based on users and items pairwise relationship for recommender systems. *Expert Systems with Applications*, 195:116593, 2022.

- [21] Mario Casillo, Brij B Gupta, Marco Lombardi, Angelo Lorusso, Domenico Santaniello, and Carmine Valentino. Context aware recommender systems: A novel approach based on matrix factorization and contextual bias. *Electronics*, 11(7):1003, 2022.
- [22] Ning Liu and Jianhua Zhao. Recommendation system based on deep sentiment analysis and matrix factorization. *IEEE Access*, 2023.
- [23] Yong Wang, Mingxing Gao, Xun Ran, Jun Ma, and Leo Yu Zhang. An improved matrix factorization with local differential privacy based on piecewise mechanism for recommendation systems. *Expert Systems with Applications*, 216:119457, 2023.
- [24] Jaafar BenAbdallah, Juan C. Caicedo, Fabio A. Gonzalez, and Olfa Nasraoui. Multimodal image annotation using non-negative matrix factorization. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*. IEEE, aug 2010.
- [25] Behnoush Abdollahi and Olfa Nasraoui. A cross-modal warm-up solution for the cold-start problem in collaborative filtering recommender systems. In *Proceedings of the 2014 ACM conference on Web science - WebSci '14*. ACM Press, 2014.



Copyright ©2023 by the authors. Licensee Agora University, Oradea, Romania.

This is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License.

Journal's webpage: <http://univagora.ro/jour/index.php/ijccc/>



This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).

<https://publicationethics.org/members/international-journal-computers-communications-and-control>

Cite this paper as:

Alshammari, M.; Alshammari, A. (2023). Friend Recommendation Engine for Facebook Users via Collaborative Filtering, *International Journal of Computers Communications & Control*, 18(2), 4998, 2023.

<https://doi.org/10.15837/ijccc.2023.2.4998>