

Towards a hybrid recommendation approach using a community detection and evaluation algorithm

Meriem Adraoui, Sonia Souabi, Asmaâ Retbi, Mohammed Khalidi Idrissi, Samir Bennani

Laboratory of Research in Computer Science and Education, Computer science Department, Mohammadia School of Engineers, Mohammed V University, Rabat, Morocco

Article Info

Article history:

Received Nov 1, 2022

Revised Apr 18, 2023

Accepted Apr 24, 2023

Keywords:

Community detection

Evaluation

Learning communities

Recommendation algorithms

Social network

ABSTRACT

In social learning platforms, community detection algorithms are used to identify groups of learners with similar interests, behavior, and levels. While, recommendation algorithms personalize the learning experience based on learners' profile information, including interests and past behavior. Combining these algorithms can improve the recommendation quality by identifying learners with similar needs and interests for more accurate and relevant suggestions. Community detection enhances recommendations by identifying groups of learners with similar needs and interests. Leveraging their similarities, recommendation algorithms generate more accurate suggestions. In this article, we propose a novel approach that combines community detection and recommendation algorithms into a single framework to provide learners with personalized recommendations and opportunities for collaborative learning. Our proposed approach consists of three steps: first, applying the maximal clique-based algorithm to detect learning communities with common characteristics and interests; second, evaluating learners within their communities using static and dynamic evaluation; and third, generating personalized recommendations within each detected cluster using a recommendation system based on correlation and co-occurrence. To evaluate the effectiveness of our proposed approach, we conducted experiments on a real-world dataset. Our results show that our approach outperforms existing methods in terms of modularity, precision, and accuracy.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Meriem Adraoui

Laboratory of Research in Computer Science and Education, Computer science Department, Mohammadia School of Engineers, Mohammed V University

Rabat, Morocco

Email: adraoui.meriem@yahoo.com

1. INTRODUCTION

Social learning networks have become an increasingly popular platform for online learning, providing learners with opportunities to collaborate, interact, and share information. To enhance the effectiveness of these platforms, community detection algorithms have been developed to identify groups of learners with common interests, learning styles, or goals. In this respect, several community detection algorithms have been proposed to identify optimal community structures in various domains [1]–[3]. These algorithms employ different concepts to detect communities based on various aspects and different perspectives, such as optimizing a specific objective function, label propagation, cliques' percolation, and so on. However, simply detecting communities is not enough to ensure the quality and relevance of the learning experience. It is crucial to also evaluate learning inside their communities based on their interactions, their characteristics or other relevant

metrics. On the other hand, recommender systems are gaining importance in all domains and levels [4]. Indeed, there are several types of recommender systems, including collaborative filtering [5] and hybrid recommender systems [6]. Recommendation systems have been widely used to personalize the learning experience by suggesting relevant and engaging content to learners. By combining community detection and evaluation with recommendation systems, we can generate more effective and personalized recommendations for each community [7]. The community structure identified by community detection can help to identify learners who are most likely to benefit from specific learning resources, while also identifying potential gaps in knowledge or skills within the community. This information can then be used to improve the quality of the learning experience and enhance the effectiveness of the recommendation system. Incorporating community evaluation into this process can ensure the identification of relevant and useful communities for learners, leading to a more effective and engaging social learning platform. In summary, here are some general steps that can be taken to achieve this goal:

- a. Community detection: Use network analysis techniques to identify communities of learners who share similar interests, skills, or learning goals.
- b. Community evaluation: Once the communities have been identified, it is crucial to assess their level of engagement. This assessment may encompass factors such as user interaction, user characteristics, and other relevant metrics.
- c. Content recommendation: Use the results of community detection and evaluation to recommend content that is relevant and interesting to users within each community. For example, you could recommend articles, videos, or courses that have been highly rated by users within a specific community or that are relevant to the skills and interests of that community.

For instance, imagine a social learning network for computer science students where users can connect, share resources, and collaborate on projects. By applying community detection algorithms, we can identify groups of students with common interests, skills, or learning goals. These communities might include: i) a group of students interested in web development, ii) a group of students interested in artificial intelligence, and iii) a group of students working on a specific programming project.

Once these communities have been identified, we can evaluate their quality and relevance to the learning platform by analyzing their network properties, user behavior, or other metrics. For example, we could assess each community's activity level, members' engagement with each other and the learning platform, and the amount of knowledge and expertise shared within the community. Based on this evaluation, we can then generate recommendations for each community using a recommendation system. For example, we might recommend specific programming languages, tutorials, or project ideas that have been highly rated by other members of the community. We might also recommend specific resources or tools that are relevant to the community's learning goals.

Our study aims to assess learning communities detected with the goal of implementing an effective recommendation system. Unfortunately, current community detection algorithms do not adequately address the learner evaluation phase. Despite numerous recommendation systems being developed for e-learning and social learning, they often fail to fully leverage learner information and do not consider the significance of integrating community detection with the evaluation of identified communities. Therefore, our study seeks to bridge this gap and highlight the importance of evaluating detected communities before generating recommendations for each community. Our paper aims to enhance recommendation relevance by proposing a combination of two concepts: the evaluation detection community algorithm (2) (EDCA (2)) method for detecting and evaluating communities, and a hybrid recommendation system based on the detected communities. The idea is to use the EDCA (2) algorithm to detect communities and evaluate them, and then generate recommendations for each community in parallel. This approach enables us to evaluate communities and compute recommendations simultaneously, leading to more effective personalized learning experiences.

Our paper is structured as follows: first, we provide a general overview of the previous work on community detection, community evaluation, and recommender systems, then we outline our proposed approach by focusing on each part in detail. Then, the following part evaluates the relevance of our approach by analyzing a database and discussing the results. Finally, a general conclusion is highlighted with potential research perspectives.

2. LITERATURE REVIEW

In this section, we present a survey of existing research methods proposed for community detection in social networks. Generally, many such methods based on different ideas have been proposed during the last decade. Fortunato [1] published a survey paper that represent different community detection approaches, such as hierarchical clustering-based methods, clique-based methods, and optimization-based methods. A recent survey presented by Coscia *et al.* [8] presents the most recent and significant definitions of community and

presents some community discovery methods. In addition, there are several surveys that give a detailed description of all existing community detection approaches [9].

Karataş and Şahin [10] discuss different application areas of community detection such as criminology, public health, politics, marketing, and others. In our case, we are interested to detect learning communities in social learning field. The objective of this study is to use learning communities' characteristics to evaluate them. Jan and Vlachopoulos [11] explored the influence of learning design on the formation and evolution of different types of the learning community. Different measures are used, like average density. The aim is to identify online health sciences course learning communities.

Recommender systems seek to predict learner preferences within collaborative learning. In the era of information technology (IT), the internet can reflect the social network, allowing community detection and clustering algorithms to be incorporated to improve their performance. Several researchers have studied the use of community detection within recommender systems. Gasparetti *et al.* [12] discuss the synthesis of the different works conducted on social recommender systems based on community detection. Ahuja [13] deals with communities of users with common interests to generate recommendations. On the other hand, several researchers addressed community detection methods in recommender systems. Souabi *et al.* [14] proposed a new recommender system based on multiple graphs. Jalali and Hosseini [15] used local dynamic overlapping community detection in a social recommendation system. Parimi and Caragea [16] integrated community detection methods in neighborhood-based recommender systems for recommending articles to users. The proposed recommendation system relies on the users' implicit preferences.

3. PROPOSED APPROACH

This contribution focuses on recommender systems that use a learning community detection algorithm. The objective is to provide better outcomes and guarantee scalability in the learning domain. As shown in Figure 1, our approach is divided into three steps: step 1 consists in detecting learning communities based on maximal clique notion. In step 2, we evaluate communities detected based on learners' social interactions (dynamic evaluation) and their socio-economics characteristics (static evaluation). Finally, in step 3, recommendations are generated based on the recommendation system integrating correlation and co-occurrence and processing each community separately.

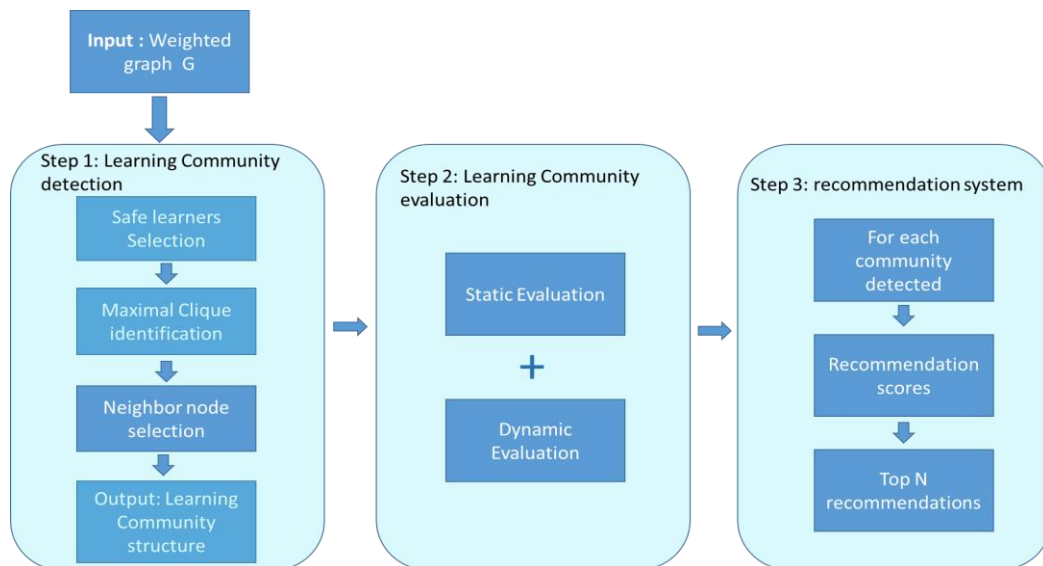


Figure 1. Step-by-step illustration process of our proposal

3.1. Preliminaries

Generally, A network is represented as a graph $G(V, E)$. V is a set of nodes, points, or vertices. E is a set of edges or links.

Definition 1: Learning community: A learning community (LC) is a group of students who share common goals, interests, learning levels, and characteristics.

Remark 1: Learning community' members work collaboratively with one or more professors.

Definition 2: Clique, A clique (Cl) is a subset of vertices of a graph G, such that each vertex is neighbor to every other vertex. As well as a maximum clique, denote the largest possible size of vertices.

Definition 3: Maximal clique, A maximal clique is a clique that cannot be extended by including one adjacent node.

Remark 2 : A maximum clique is a maximal clique but not necessarily vice versa (the opposite is incorrect).

Definition 4: Degree centrality is one of the most accessible measures to calculate. It represents the number of connections linked to a vertex.

Remark 3 : Nodes with high degrees have high centrality, representing more actives in the network.

Definition 5: Adjacency matrix is a matrix of $n \times n$ dimension, which is used to represent a graph G with n nodes. The matrix elements are 0 or 1 according to whether two nodes are adjacent or not.

Definition 6: Safely centrality measure proposed by Adraoui *et al.* [17], this measure intended to calculate the degree of success of each learner in a social learning network. In addition, "safely centrality" is a useful concept to detect the most successful learners in the network. This measure is based on students' interactions in the network. Hence, the calculation of the distance between all pairs of vertices in a graph becomes necessary. In a more formal sense, if there exists an interaction between two nodes, the distance between them is assigned a value of 1, whereas if there is no interaction, the distance is considered null. "Safely centrality" is defined by (1).

$$\text{Safely centrality } (U_i)_{i \neq j} = \frac{\sum_{i,j \in V} d(i,j)m_{ij}}{(N-1) D_i} \tag{1}$$

The distance between two vertices (u and $v \in V$), denoted as $d(u, v)$, is defined as 1 if there is a connection between u and v , and 0 if there is no connection. The arc weights, represented by m_{ij} , indicate the total number of interactions between two learners. N represents the total number of learners in the network, and D_i represents the degree of node i .

3.2. EDCA (2) approach

This subsection presents our approach named evaluation detection community algorithm (2) (EDCA (2)). EDCA (2) is a clustering algorithm based on the maximal clique notion. EDCA (2) is divided into three steps. To explain these steps better, we illustrated in example on a weighted graph G consisting of 9 vertices and 11 links Figure 2. Indeed, EDCA (2) algorithm follows the following steps:



Figure 2. The adjacency-weighting matrix of an undirected weighted graph G represents the relationships between nodes. The weight of the links indicates the presence of a connection between two nodes, while null values indicate no connection between them

Step 1: Safe learners selection: To begin with EDCA (2), the initial step consists of identifying safe learners. We utilized the "safely centrality" measure to evaluate the level of success for each learner within the network. Based on definition 6, this measure utilizes pairwise distances and the degree D_i . To calculate these distances, we utilized the adjacency matrix defined in definition 5, which was further multiplied by the link weight. In our illustration, the weighted graph G's adjacency matrix is depicted in Figure 2. Consequently, learners with elevated "safely centrality" values are categorized as safe, whereas those with lower values are considered at-risk Figure 3.

- Step 2: Maximal cliques' identification: In the second step, we detect maximal cliques containing safe learners. If two k -cliques share $k-1$ nodes, we compute their degree centrality and select the clique with the higher value. From Figure 3, we find three maximal 3-cliques: $\{a, b, c\}$, $\{a, b, h\}$, and $\{e, f, g\}$. The first two cliques share nodes a and b . Their degree centrality values are 8 and 4, respectively. Hence, we choose the first clique. After identifying all maximal cliques, we assign each to a separate community.
- Step 3: Neighbor node selection: The last step of our algorithm is the neighbor node selection; the objective of this step is to find the neighbor of all remaining nodes in the graph in order the shape communities. More precisely, if neighbors of a node belong to the same community, then we insert it in the community of its neighbors. Otherwise, we insert the node in the community with a maximum weight value.

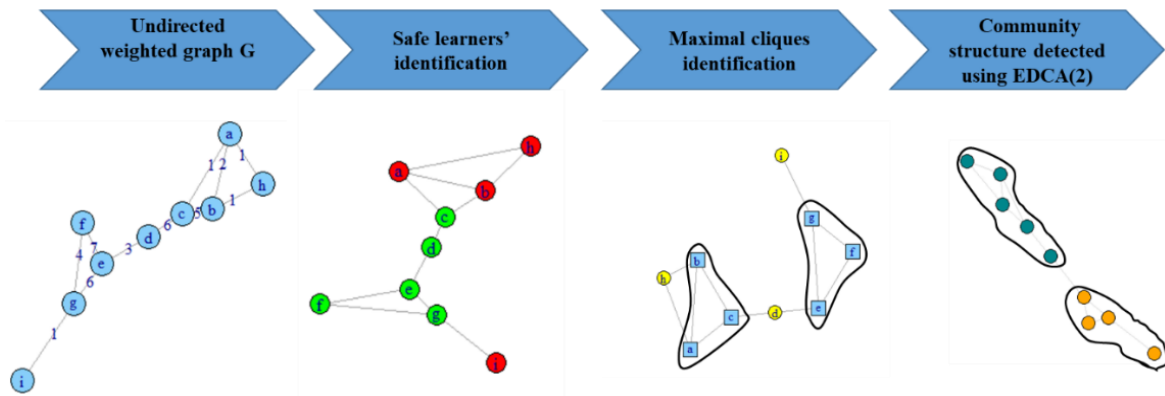


Figure 3. Step-by-step illustration process of our proposal of EDCA (2) on a graph G

3.3. Learning community evaluation

In all social learning environments, it is crucial to monitor the progress of learners in real-time [18]. Generally, there are many methods to assess them, such as quizzes, game-based assessments, online video conferencing, interactions. Our study proposes a new method for evaluating learners within their clusters based on their interactions and socio-economics characteristics. More precisely, our process includes two types of evaluation:

- Dynamic evaluation: the ability to interact with others plays a crucial role in a student's ability to learn. When students engage with their peers, they are exposed to a wide range of perspectives and insights, which can enlarge their understanding of a particular topic. Furthermore, interacting with their peers can help students to identify their own strengths and weaknesses, and give them the opportunity to collaborate and work together to achieve a common goal. The dynamic evaluation process proposed in this paper is a method used to assess learners based on their interactions with each other. This process allows instructors to track their progress and understanding, and adapt their teaching methods accordingly. By monitoring student interactions, teachers can identify areas in which students are struggling and provide additional support or guidance as needed. In order to evaluate the students' success, we use the EDCA (2) algorithm to detect groups of students who are working together. The "safely centrality" measure is then used to determine the success of each student within their respective community. This measure takes into account both the number and quality of interactions a student has with his or her peers. Using this measure, we are able to identify students who are highly engaged and actively contributing to the success of the group. After that, we used the density measure [19] to calculate the density of communities. This measure helps us understand the level of cohesion within each community and identify potential areas for improvement.
- Static evaluation: Learners' characteristics are essential for instructors because they allow them to make learning more effective and more helpful. In the significant case, learners prefer to study with learners of the same characteristics. In this way, the static evaluation process aims to evaluate learners according to their socio-economic characteristics. In this context, we use the Homophily measure [20] to calculate the community assortment rate. As a result, we identify two types of communities: Safe community and at-risk community [21].

3.4. Recommendation system

This phase consists in generating recommendations for each community detected in the previous steps. The considered recommendation system involves a combination of two aspects: correlation and

co-occurrence. The idea is to focus on the activities performed by learners, i.e., on the implicit feedbacks related to the pedagogical elements. The analysis of these actions is an opportunity to generate more appropriate recommendations for social learning environments. The process calculates the correlation score and co-occurrence from the learners' history, then calculates the recommendation scores and generates the top N recommendations. This system undergoes several significant phases:

- a) First phase: It involves preparing the data by cleaning, filtering, and transforming it. After collecting learners' interaction data, it is adapted and converted into a table of interactions between learning objects and activities. The resulting data is then filtered to include relevant actions for recommendation calculations. To evaluate the recommendation system's relevance, the data is divided into two parts: one for designing the recommendation model Figure 4 and the other for measuring evaluation metrics.
 - b) The second phase: The focus of this phase is on utilizing two sections of the database. The majority (80%) of the converted and filtered data is utilized for recommendation calculations within each detected community, while the remaining 20% is used to assess the system's relevance. This process is repeated for each identified community, as recommendation scores are calculated based on correlation and co-occurrence scores within the training database. Finally, the correspondence between these scores and actual preferences within each community is examined to evaluate the effectiveness of the recommendations.
- Thus, to calculate the total recommendation score, we combine the correlation score (2) and the co-occurrence score (3) to obtain the pursuing results (4).

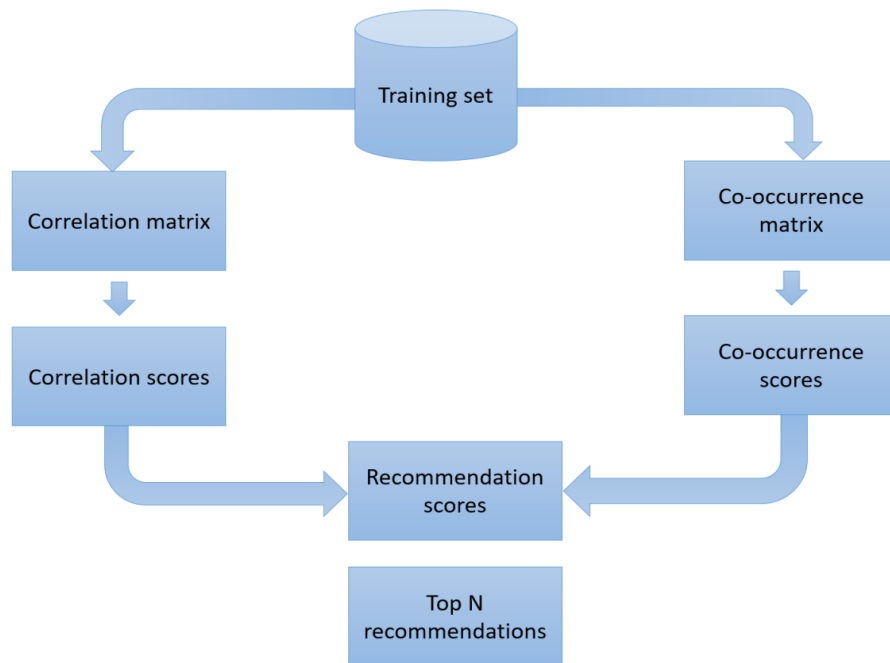


Figure 4. The process of the recommendation approach

$$\begin{aligned}
 \text{Correlation score matrix} &= \begin{bmatrix} h_{i_1 a_1} & h_{i_1 a_2} & \dots & h_{i_1 a_m} \\ h_{i_2 a_1} & h_{i_2 a_2} & \dots & h_{i_2 a_m} \\ \vdots & \vdots & & \vdots \end{bmatrix} \times [1 \quad \text{cor}(a_1, a_2) \quad \dots \quad \text{cor}(a_1, a_m)] = \\
 &\begin{pmatrix} \text{correlation score}(i_1) \\ \vdots \\ \text{correlation score}(i_n) \end{pmatrix} \tag{2}
 \end{aligned}$$

$$\begin{aligned}
 \text{Co-occurrence score matrix} &= \begin{bmatrix} H_{i_1 a_1} & H_{i_1 a_2} & \dots & H_{i_1 a_m} \\ H_{i_2 a_1} & H_{i_2 a_2} & \dots & H_{i_2 a_m} \\ \vdots & \vdots & & \vdots \end{bmatrix} \times \\
 [1 \quad \text{co-occ}(a_1, a_2) \quad \dots \quad \text{co-occ}(a_1, a_m)] &= \begin{pmatrix} \text{co-occurrence score}(i_1) \\ \vdots \\ \text{co-occurrence score}(i_n) \end{pmatrix} \tag{3}
 \end{aligned}$$

$$\begin{aligned} \text{Total score matrix} &= \text{co-occurrence score matrix} + \text{correlation score matrix} = \\ &= \begin{pmatrix} \text{correlation score}(i_1) + \text{co-occurrence score}(i_1) \\ \vdots \\ \text{correlation score}(i_n) + \text{co-occurrence score}(i_n) \end{pmatrix} \end{aligned} \quad (4)$$

where,

$\{i_1, i_2, \dots, i_n\}$: Learning objects to recommend.

$\{a_1, a_2, \dots, a_m\}$: Activities performed by learners such as a_1 is the primary activity and $\{a_2, \dots, a_m\}$ are the secondary activities.

$\{(h_{i_1 a_1}, \dots, h_{i_1 a_m}), \dots, (h_{i_n a_1}, \dots, h_{i_n a_m})\}$: Learners historical activities towards each learning object.

$\{(H_{i_1 a_1}, \dots, H_{i_1 a_m}), \dots, (H_{i_n a_1}, \dots, H_{i_n a_m})\}$: Binary history of activities performed by learners for each learning object.

4. EXPERIMENT STUDY

In this section, we test the performance of our approach experimentally. In this sense, we use a real-world dataset to detect, evaluate learning communities and generate recommendations. In addition, we compared the obtained results with three community detection algorithms: InfoMap [22], label propagation [23], and leading eigenvector [24] using the quality measure: the modularity (Q) [25].

The modularity is the first measure is one of the most well-known in the literature. It compares the actual intra-community links with the probability of uncovering those links in a random network. It has a maximum value if there are many links inside communities and only a little among them. In this context, the partition with a larger modularity score is considered the best one. The modularity of a division D for a graph G is formulated as:

$$Q(D) = \sum_{i=1}^{|D|} (e_{ii} - a_i^2)$$

where e_{ii} is the probability of intra-community link in community i while a_i is the probability of a link with at least one extremity in community i . Then, we compare two recommender systems for highlighting the importance of community detection in generating relevant recommendations. Finally, we used R for implementing our approach.

4.1. Dataset

In this experiment, we used a real-world dataset named "Students profiles and activity." This dataset was described by Martín *et al.* [26]. It was based on an educational experience that focuses particularly on video and the collaborative social concept. The platform includes the different activities carried out by students and teachers, including profiles. All interactions are included in the learning platform, including messages and friendship links. Each profile is described by the role: student and teacher as well as the affiliation. In our study, we are interested in friendships among learners to detect learning communities and suggest recommendations. In this context, the data is modeled as a graph in which nodes describe learners and relationship among them presented by links. This modeling facilitates the detection of learning communities. In this way, we have generated a friendship network that contains 415 nodes (learners) and 601 interactions (friendship). The learners' affiliation is recorded as a vertex attribute. Figure 5(a) displays the initial learning network of "Students interaction and profiles."

4.2. Results: learning community detection

This subsection presents the result obtained by the EDCA (2) approach. Figure 5 illustrates the community structure found by our algorithm. More precisely, Figure 5(a) provides the initial learning graph. Nodes represent learners, links represent friendships relations among them, and edge weights represent the relationship strength. Indeed, in Figure 5(b), communities are illustrated by different colors and they are well divided. Our approach was evaluated based on the effectiveness of identifying meaningful clusters of students, as measured by modularity Table 1. A higher modularity score indicates that the nodes within each community are more densely connected, while the connections between communities are sparser, and thus the approach is more effective in identifying clusters of students with common interests and learning goals. Afterward, we remark that the modularity values of EDCA (2), InfoMap, and label propagation are close to each other, this result indicates that these partitions are similar.

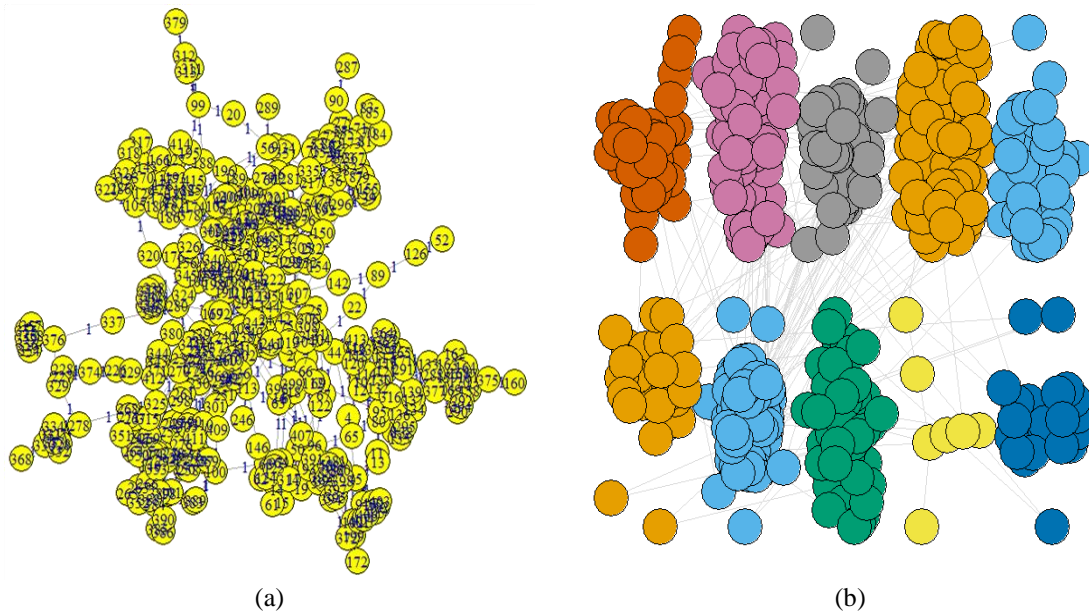


Figure 5. The community structure obtained by EDCA (2), (a) the initial graph and (b) result for student's profiles and activity using EDCA (2)

Table 1. The modularity results

Dataset	EDCA (2)	InfoMap	Label propagation	Leading Eigen vector
'Students profiles and activity'	0.69	0.62	0.62	0.59

4.3. Results: community evaluation

The paper presents a comprehensive evaluation process that encompasses two types: i) static evaluation and ii) dynamic evaluation. The approach employed in this paper and the specific details pertaining to these two types are elaborated in the subsequent paragraphs.

- Static evaluation: in this database, learners are presented by their affiliation. This characteristic allowing us to calculate the homophily by affiliation. The homophily score measures the intensity with which individuals with similar characteristics or attributes, are inclined to interact with each other. More specifically, we analyzed the extent to which learners with the same affiliation were more likely to interact with each other. Our results, presented in Table 2, we noticed that a majority of communities in our dataset had low homophily values, and the most of them are negative. These results indicate that learners' affiliation has not significantly impacted their interactions. In other words, we did not observe a tendency for learners with the same affiliation to interact more frequently or exclusively with each other, which indicates that other factors may have been more influential in shaping the social interactions between learners.
- Dynamic evaluation: in this type of evaluation, we used the "safely centrality" measure to calculate each cluster's success degree and density. As mentioned in Table 2, these two indicators are more significant, and they are helpful to identify the status of clusters.

Table 2. The community evaluation results (H: homophily, D: density, S: safely centrality)

	Students' profiles and activity			Status
	H	D	S	
Cluster 1	0.024	0.07	0.40	At-risk
Cluster 2	0.068	1.56	0.29	Safe
Cluster 3	-0.63	0.062	0.19	At-risk
Cluster 4	-0.25	0.21	0.47	Safe
Cluster 5	0.14	0.067	0.39	Safe
Cluster 6	0.31	0.065	0.50	Safe
Cluster 7	0.23	0.041	0.44	Safe
Cluster 8	-0.087	0.039	0.20	At-risk
Cluster 9	-0.203	0.0387	0.42	At-risk
Cluster 10	-0.39	0.07	0.78	Safe

In the final step, we evaluate communities using indicators such as homophily, density, and safely centrality measure. Homophily values are mostly negative or low, making it insignificant for cluster evaluation. Density and safely centrality are prioritized indicators, measuring connectivity and learners' interactions within communities. Cluster 2 demonstrates high density and success, indicating active learner engagement. On the other hand, clusters 1, 3, 8, and 9 have low density and success, suggesting challenges in learner interaction and a need for additional support.

4.4. Results: generating recommendations

This step generates the recommendations for each detected community based on the recommendation system combining correlation and co-occurrence. First, the initial database is converted to a database of interaction between learning objects and activities performed by learners. We extracted two types of activities: the evaluation of learning items considered a primary activity and the creation of comments as a secondary activity. Thus, we calculate the precision and accuracy while varying the number of recommendations from four to six in each community. Given the importance of item evaluation, this activity is considered primary. Then, the creation of comments is a further activity that might be considered a secondary activity. We aim to choose several recommendations that are both considerable and reasonable. More precisely, we cannot recommend more than six resources to the learners to overwhelm them with too many recommendations.

After partitioning the initial database into two parts, we calculate 80% of the data recommendations. Then we measure the evaluation metrics in the remaining 20% of the data: precision and accuracy. We implement this process for each community detected and then record each community's precision and accuracy values separately. Two recommendation systems are compared: the first recommendation system based on the detected communities and the second recommendation system treating all learners as a single community. Table 3 represents an extract of the total scores obtained for some learning objects.

Our objective is to exploit the interactions performed by learners towards the learning environment to generate recommendations. In this perspective, we will be restricted to operating only the communities where the maximum number of interactions is recorded and relevant for recommendations. After calculating the precision and accuracy for the selected communities, we reach the following results in Table 4.

The results show that the recommendation system based on community detection performs better than the recommendation system considering all learners as a single community. Regarding the accuracy, the first recommender system records a maximum value of 1 with a single value of 0.75. It was against values ranging between 0.26 and 0.522 for the second recommender system. The recommendation system based on community detection thus exceeds the recommendations that do not include community detection. Regarding the accuracy, the first recommendation system adopts values between 0.95 and 1 against values between 0.77 and 0.91, presenting high values compared to the recorded accuracy.

Table 3. Extract of the total score results obtained

Learning object	E1	E2	E3	E4	E5	E6	E7
Recommendation score	8.3464102	0.7698004	2.9549872	2.7883049	2.5700870	0.3849002	3.1732051

Table 4. Precisions and accuracies obtained for the two recommender systems in communities selected

	Recommender system without community detection		Recommender system with community detection	
	Precision	Accuracy	Precision	Accuracy
Cluster 5	1	0.26	1	0.91
Cluster 6	0.75	0.25	0.95	0.9
Cluster 7	1	0.522	1	0.77
Cluster 8	1	0.52	1	0.79
Cluster 9	1	0.47	1	0.86
Mean value	0.95	0.4	0.99	0.846

4.5. Discussion

This paper proposed a recommender system based on the EDCA (2) approach for detecting and evaluating learning communities. Next, we analyzed the "Students profiles and activity" database, which describes the interaction among learners in an e-learning platform. Finally, we detected nine communities based on the friendship links wherein six are "safe," and the other is "at-risk".

From the detected communities, recommendations are generated for each community separately. Furthermore, when applying our recommendation system, we selected only those communities that support the generation of recommendations by exploiting the proposed system. Sixty percent of these communities are "safe", emphasizing the importance of the proposed evaluation process in applying our recommender system.

More precisely, safe communities group the most active learners in the network with similar characteristics. In general, the higher the interaction rate of these learners, the more efficient our recommendation system becomes, with higher precision and accuracy.

5. CONCLUSION

In social learning, learners tend to interact with others who share similar interests, characteristics, and learning levels. These interactions form communities within the network, where recommendations can be generated. This paper proposes an approach that combines a community detection and evaluation algorithm with a recommender generation algorithm. Our goal is to detect communities based on friendship links and evaluate them for the recommendation system. We compared our EDCA (2) algorithm with other community detection algorithms in the literature, and the obtained modularity demonstrates its efficiency. The community structure simplifies the evaluation process, focusing on friendship links and learners' affiliations. We identified two types of communities: "safe" and "at-risk". Using these communities, we applied our recommendation system and compared it to another system based on precision and accuracy. The results emphasize the importance of combining both approaches to provide more suitable recommendations. In the future, we plan to explore larger datasets with diverse interaction types and additional characteristics like age and gender.

REFERENCES




- [1] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3–5, pp. 75–174, Feb. 2010, doi: 10.1016/j.physrep.2009.11.002.
- [2] E. Akachar, B. Ouhbi, and B. Frikh, "ACSIMCD: A 2-phase framework for detecting meaningful communities in dynamic social networks," *Future Generation Computer Systems*, vol. 125, pp. 399–420, Dec. 2021, doi: 10.1016/j.future.2021.06.056.
- [3] E. Akachar, B. Ouhbi, and B. Frikh, "A new algorithm for detecting communities in social networks based on content and structure information," *International Journal of Web Information Systems*, vol. 16, no. 1, pp. 79–93, 2019, doi: 10.1108/IJWIS-06-2019-0030.
- [4] Q. Zhang, J. Lu, and Y. Jin, "Artificial intelligence in recommender systems," *Complex and Intelligent Systems*, vol. 7, no. 1, pp. 439–457, Feb. 2021, doi: 10.1007/s40747-020-00212-w.
- [5] F. Rezaimehr and C. Dadkhah, "A survey of attack detection approaches in collaborative filtering recommender systems," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 2011–2066, Mar. 2021, doi: 10.1007/s10462-020-09898-3.
- [6] N. Haubner and T. Setzer, "Hybrid recommender systems for next purchase prediction based on optimal combination weights," in *WI 2021: Innovation Through Information Systems*, 2021, pp. 56–71.
- [7] L. Paleti, P. Radha Krishna, and J. V. R. Murthy, "Approaching the cold-start problem using community detection based alternating least square factorization in recommendation systems," *Evolutionary Intelligence*, vol. 14, no. 2, pp. 835–849, Jun. 2021, doi: 10.1007/s12065-020-00464-y.
- [8] M. Coscia, F. Giannotti, and D. Pedreschi, "A classification for community discovery methods in complex networks," *Prepr. arXiv.1206.3552*, Jun. 2012, doi: 10.1002/sam.10133.
- [9] S. Souravlas, S. Anastasiadou, and S. Katsavounis, "A survey on the recent advances of deep community detection," *Applied Sciences*, vol. 11, no. 16, Aug. 2021, doi: 10.3390/app11167179.
- [10] A. Karatas and S. Sahin, "Application areas of community detection: A review," in *2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT)*, Dec. 2018, pp. 65–70, doi: 10.1109/IBIGDELFT.2018.8625349.
- [11] S. K. Jan and P. Vlachopoulos, "Influence of learning design of the formation of online communities of learning," *The International Review of Research in Open and Distributed Learning*, vol. 19, no. 4, Sep. 2018, doi: 10.19173/irrodl.v19i4.3620.
- [12] F. Gasparetti, G. Sansonetti, and A. Micarelli, "Community detection in social recommender systems: a survey," *Applied Intelligence*, vol. 51, no. 6, pp. 3975–3995, Jun. 2021, doi: 10.1007/s10489-020-01962-3.
- [13] M. S. Ahuja, "Using community detection technique in recommender system," *Advances in Mathematics: Scientific Journal*, vol. 9, no. 6, pp. 3741–3750, Jul. 2020, doi: 10.37418/amsj.9.6.52.
- [14] S. Souabi, A. Retbi, M. K. Idrissi, and S. Bennani, "Toward a recommendation-oriented approach based on community detection within social learning network," in *AI2SD 2019: Advanced Intelligent Systems for Sustainable Development (AI2SD'2019)*, 2020, pp. 217–229.
- [15] S. Jalali and M. Hosseini, "Social collaborative filtering using local dynamic overlapping community detection," *The Journal of Supercomputing*, vol. 77, no. 10, pp. 11786–11806, Oct. 2021, doi: 10.1007/s11227-021-03734-3.
- [16] R. Parimi and D. Caragea, "Community detection on large graph datasets for recommender systems," in *2014 IEEE International Conference on Data Mining Workshop*, Dec. 2014, pp. 589–596, doi: 10.1109/ICDMW.2014.159.
- [17] M. Adraoui, A. Retbi, M. K. Idrissi, and S. Bennani, "A new algorithm to detect and evaluate Learning communities in social networks: Facebook groups," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 14, no. 23, pp. 165–179, Dec. 2019, doi: 10.3991/iJET.v14i23.10889.
- [18] R. Kaliisa, B. Rienties, A. I. Mørch, and A. Kluge, "Social learning analytics in computer-supported collaborative learning environments: A systematic review of empirical studies," *Computers and Education Open*, vol. 3, Dec. 2022, doi: 10.1016/j.caeo.2022.100073.
- [19] J. Chen and Y. Saad, "Dense subgraph extraction with application to community detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 7, pp. 1216–1230, Jul. 2012, doi: 10.1109/TKDE.2010.271.
- [20] K. Z. Khanam, G. Srivastava, and V. Mago, "The homophily principle in social network analysis," *Prepr. arXiv.2008.10383*, 2020.
- [21] M. Adraoui, A. Retbi, M. K. Idrissi, and S. Bennani, "Maximal cliques based method for detecting and evaluating learning communities in social networks," *Future Generation Computer Systems*, vol. 126, pp. 1–14, 2022, doi: 10.1016/j.future.2021.07.034.
- [22] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, Jan. 2008, doi: 10.1073/pnas.0706851105.
- [23] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review E*, vol. 76, no. 3, Sep. 2007, doi: 10.1103/PhysRevE.76.036106.
- [24] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E*, vol. 74,

no. 3, Sep. 2006, doi: 10.1103/PhysRevE.74.036104.




- [25] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, Jun. 2006, doi: 10.1073/pnas.0601602103.
- [26] E. Martín, M. Gértrudix, J. U.-Fuentes, and P. A. Haya, "Student activity and profile datasets from an online video-based collaborative learning experience," *British Journal of Educational Technology*, vol. 46, no. 5, pp. 993–998, 2015, doi: 10.1111/bjjet.12318.

BIOGRAPHIES OF AUTHORS






Adraoui Meriem    in 2013, she obtained a Bachelor of Science degree in Computer Science from Hassan First University in Settat, Morocco. She then pursued her studies and obtained a Master of Science degree in 2015 from the University Mohamed Ben Abdallah in Fez, Morocco. In 2021, she successfully defended her Ph.D. thesis in Computer Science at Mohammed V University of Rabat, Morocco. Her research interests are centered around online assessment techniques, social networks, as well as the detection and evaluation of learning communities. He can be contacted at email: meriemadraoui@research.emi.ac.ma.






Sonia Souabi    Doctor of Philosophy and recently graduated from Mohammadia School of Engineers in Mohammed V University of Rabat, my thesis work focuses on recommendation systems in connection with online learning, particularly social learning. I have worked on several international ranked conferences as well as Scopus-indexed Q2 journals. My work was supervised by Mr. Mohammed Khalidi Idrissi, my Thesis Director and Mrs. Asmaâ Retbi, my Co-Director. Our research team works on the application of IT tools and data analysis in the e-learning context. We have published several papers in many international conferences in Morocco and other countries like Germany, United Kingdom. We are looking forward to propose a novel recommendation approach within social learning networks combined with community detection in order to meet learners' needs. She can be contacted at email: soniasouabi@research.emi.ac.ma.






Asmaâ Retbi    at the Mohammadia School of Engineers, she holds the position of Full Professor. She obtained her Engineer degree in Computer Science in 1997, followed by a DESA degree in the same field in 2000. In 2015, she earned her PhD in Computer Science. Currently, she teaches at the Computer Science Department of EMI and her research interests include e-learning, authoring tools, model-driven engineering (MDE), and domain specific modeling. He can be contacted at email: retbi@emi.ac.ma.



Mohammed Khalidi Idrissi    is a full Professor at the Mohammadia School of Engineers and has a Doctorate degree in Computer Science (1986) and a PhD in the same field (2009). He served as the former Assistant Chief of the Computer Science Department at EMI and currently works as a Professor at the Computer Science Department of EMI. Additionally, he is a Pedagogical Tutor of Computer Science at EMI. He has authored 34 recent publications between 2014 and 2017. His research interests include software engineering, information systems, modeling, model-driven architecture (MDA), ontology, service-oriented architecture (SOA), web services, e-learning content engineering, tutoring, and assessment and tracking. He can be contacted at email: khalidi@emi.ac.ma.



Samir Bennani    is a full Professor and Deputy Director of students and academic affairs at Mohammadia School of Engineers. He holds an Engineer degree in Computer Science obtained in 1982 and a PhD in Computer Science earned in 2005. He currently serves as a Professor at the Computer Science Department-EMI. He has authored 34 recent research papers published between 2014 and 2017. His ongoing research interests revolve around software engineering, information systems, e-learning content engineering, tutoring, assessment, and tracking in the context of social interaction (SI) and modeling. He can be contacted at email: sbennani@emi.ac.ma.