Community Graph Elicitation from Students' Interactions in Virtual Learning Environments

Paolo Mengoni[1](✉), Alfredo Milani[2,3], and Yuanxi Li[3]

[1] Department of Mathematics and Computer Science,
University of Florence, Florence, Italy
paolo.mengoni@unifi.it

[2] Department of Mathematics and Computer Science,
University of Perugia, Perugia, Italy
milani@unipg.it

[3] Department of Computer Science,
Hong Kong Baptist University, Kowloon Tong, Hong Kong
csyxli@hkbu.edu.hk

Abstract. In this work we introduce a novel graph-based approach to elicit students' communities. Teaching, in the blended learning environment, is delivered as a mixture of online and offline activities. While the online activities can be tracked and analysed in the Virtual Learning Environment, the offline activities fall out of the educators' control scope. In this educational setting, communications take place using side channels, such as the instant messaging applications and social network platform. Using our approach, the students' groupings and social interactions can be elicited by analysing the student-system interactions. The co-occurrence of interactions among the students give information about their social connections. This conveys information useful to elicit the students' interaction graph and the student communities contained in it. Students' leader-follower community structure can be elicited starting from the interaction network. This can empower teachers to plan and revise their Learning Designs as well as to identify situations that need teacher's intervention, e.g. students at risk of failing the exam and/or dropping the studies.

Keywords: Community detection · Student interactions · Graph analysis
Modularity maximization

1 Introduction

The use of Virtual Learning Environments (VLE) give teachers the ability to track and analyse the students' activities that take place online. Teaching, in the online scenario, can be delivered to learners with different mixtures of online and offline activities.

Finding the interactions between the students can empower educators to plan and revise their Learning Designs (LD) as well as to identify situations that need teacher's intervention, e.g. students at risk of failing the exam and/or dropping the studies. This is a difficult task in the blended learning environment. When teaching and learning take place exclusively online, like in MOOC, even the communications among the students can be tracked and analysed. In the blended learning scenario, online and offline

O. Gervasi et al. (Eds.): ICCSA 2018, LNCS 10962, pp. 414–425, 2018.
https://doi.org/10.1007/978-3-319-95168-3_28

activities take place to in different environments. The result is that not all the students' communications can be tracked by the educator. In fact, in addition to the online tools provided by the VLE, students interact directly in class as well as using social network platforms and instant messaging. The side channel communications are out of teachers' control scope and don't leave traces to analyse.

We hereby introduce an innovative approach to discover and analyse latent students' interaction network. This approach use data about the students' interactions with the VLE. The co-occurrence of interaction with the system give information about the social interactions of the students. This conveys information useful to elicit students' interaction graph and the student communities contained in it.

This paper is organized as follows. In Sect. 2 we review the related work in Community Detection in networks. In Sect. 3 we introduce the framework to discover students' communities in Virtual Learning Environments. Section 4 presents the experimental setting and methodology. In Sect. 5 we discuss the experimental results. Finally, in Sect. 6 we draw conclusion e present future works.

2 Related Work

Communities in networks are groups of elements that share similar characteristics. The community structure within a network can be elicited from the connections among the elements. Given an element of the graph, the community search class of algorithms deal with identifying in which community the element can be included. Another category of algorithms has been developed to solve the community detection problems that, given a graph, identify all the communities in the network.

Existing community eliciting strategies use one or more of the following methods to evolve starting partitions: merge two communities, split a community into two, or move elements between two distinct communities. A comprehensive review of the state of the art in community detection can be found in Fortunato work "Community detection in graphs" [1]. Different algorithms have been developed that exploit different characteristics of the graphs and their elements. Approximate solutions are often found as the problem is complex and general polynomial time algorithms cannot be found.

Traditional methods for community detection include graph partitioning and clustering. The former requires to specify the number of partitions to find and exploit the network edges features to find the communities [2]. Most variants are NP-Hard even if good approximations have been found [3]. Clustering algorithms use distance measures to pack the elements of a graph in different groups. Widely used in other research fields, such as emotion recognition [4–9], hierarchical [10], partitional [11] and spectral [12] methodologies can be applied to discover the different communities within the graphs. Divisive algorithms, like Girvan-Newman [13], detect the edges that connect elements of different communities and remove them, so that the groups get disconnected from each other.

The modularity value, introduced as stopping criterion for Girvan-Newman divisive algorithm, is a quality assessment of the communities. This parameter has been used in many other works, as distinctive element to identify communities in networks. Greedy [14], simulated annealing [15], evolutionary computation [16, 17] and other heuristic methods [18] have been used to solve the modularity maximization optimization problem. Statistical inference generative algorithms try to find a model that can fit the graph structure and its communities [19]. Block modelling is based on grouping vertices with common features [20].

Students' interactions with the VLE and their social interactions have been studied in various works. Monitoring tools have been developed to understand the evolution of participant relationships within discussions forums. For instance, SNAPP tool [21, 22] uses MOOC's discussion forum threads to extract the students' social network and give educators' insights about students' interactions and groupings. An early study of students' interactions with the VLEs has been conducted to understand the opportunities and impact of the new teaching environment for educators [23]. The timing of engagement with the learning objects [24, 25] in online courses has been examined in recent papers to understand the links with students' final learning outcome [26] and to discover the students' study patterns [27]. Clustering methods have been used to understand, in a flipped classroom environment, the connections between students' engagement with the VLE and their outcome as well as to discover the grouping of students with similar outcome and study behaviour [28].

3 Methodology

In this work a novel approach to discover the communities of students is introduced. The approach is based on interactions of students with the Virtual Learning Environment. The proposed methodology is based on the co-occurrence of students' interactions in the VLE to elicit the underlying social relations among the students.

When a student accesses the course material, his actions are logged by the eLearning system. The basic information that is recorded comprises the course that the student accessed, the starting time of interaction, the action (read announcements and forum posts, views and downloads of resources, submission of assignments, etc.) and the objects he interacted with.

The focus of the approach is on students' activities in the VLE as it checks the presence of interactions and their timings. Using this information, we can model the students' behaviour by analysing their interactions with the VLE and use it as the foundation for eliciting the students' connections, social activities and study behaviour.

The purpose of the proposed approach is to provide teachers and academic managers insights from static and dynamic points of view. The static point of view gives insight on community information related to each student at various levels of time and activity granularity, e.g. one or more courses in a time interval that can span from a few weeks to the complete academic career. The dynamic point of view helps to track students social interactions' evolution over time. Educators can get insights on the stability or the evolution of students' communities over the time.

The framework use information from the students' interactions with the system. Various approaches to the analysis of student-system interaction histories can be used in our framework. Clustering, time series analysis, events cascades and graph analysis are approaches for analysis of histories. A different modelling and pre-processing of the information allow to use the various approaches.

The graph approach here introduced use information from the students' interactions with the system. The co-occurrence of student-system interactions is used to build the student-student interaction network. The information conveyed by the interaction graph can help educators to determine students' connections as well as to identify the communities of students.

Moreover, this approach has several additional advantages that come from the ability to inspect the network structure. The members of the communities can be differentiated basing on their positioning in the network and their neighbourhood connections. As an example, using the information from centrality measures together with the timing of interactions, community leaders and followers can be identified.

3.1 Graph Analysis

The information about student-system interactions is used to build the students' interaction graph. This approach has its foundation on multigraph analysis. In fact, each node is connected to the others by multiple edges. The proposed method proceeds by first creating the interactions' graph between the students using the co-occurrence of interactions in the VLE and proceeds to detect communities in the students' interactions graph.

Students Interactions Graph. The student-system interaction information is used to model the students' interactions graph. The data needed by our approach comprises only information about the student that interact with the system and the timestamp of the interaction.

In this graph the nodes represent the students while the edges represent the interactions between each student pair. Edges are created when the interactions' timestamp delta of the students' actions is below a pre-set granularity. On each edge is recorded the information about the start and end time of interaction.

The graph obtained is a multigraph. In fact, recurring interactions can be recorded between each student pair.

Community Detection. Given the students interactions graph, the goal is to detect all the groups of students in it. Our approach is general as it can use various techniques to elicit the students' communities. The community detection problem has been solved using various algorithms with different characteristics. The following are some of the most frequently used:

Minimum Cut. This method finds a predetermined number of communities in a network. It works by grouping the nodes in such a way the number of connections between the communities is minimized. The partitions found by the algorithm are balanced in number of nodes. This algorithm work well for load balancing in networks and parallel computing environments.

Girvan-Newman. The top-down hierarchical algorithm, introduced in "Community structure in social and biological networks" [13], works by removing the communities-connecting edges. The edges are selected by calculating their betweenness centrality at each step and the highest ranked is removed from the network. The algorithm has high complexity that makes it slow for big networks.

Clique Detection. Cliques are groups of two or more individuals who share similar characteristics and are connected one to each other forming a network. Cliques can overlap each other which is a desirable characteristic for many social network tasks. Detection methods can find cliques of fixed size or with the maximal number of elements. The former can use percolation methods to determine the node cliques, while the classical algorithm for the latter is the Bron–Kerbosch algorithm [29].

Statistical Inference. Methods using statistical inference try to generate the network structure using a model that exploits the features of the input data. The general approach is to use a stochastic blockmodel [30] and its variants to produce graphs that contain communities of nodes. Edges within the communities are more dense than the edges connecting one community to the other. Other approaches use belief propagation [31] and Monte Carlo [32] heuristic methods.

Modularity Maximization. Modularity is value that measures the strength of division of a network into modules. Having high modularity groupings in a network is represented by having dense connections among the nodes within modules and sparse connections between nodes in different modules. Modularity maximization methods try to find the grouping that has the highest modularity within a network. The naïve method is to start with one node and proceed by adding a connected node to the group until the modularity score increases. This problem has been proved to be NP-Hard [33] but heuristic algorithms have been developed.

Each of the listed solutions work best on a defined set of problems that differ on the network structure. For social networks, statistical inference and modularity maximization, with various approaches and approximations, are widely used.

4 Experiments

4.1 Dataset

For this study a new dataset has been created using the students' interaction logs with the University of Perugia Moodle eLearning system. Data has been extracted from the course *"Ingegneria del Software"* (Software Engineering) held in the Academic Year 2016/2017. The course has been taught in blended mode using online and offline activities. The online activities include the weekly release of course material, project submissions and assessment quiz, while the offline part includes face-to-face lessons, group project development and final examination. The group project was finally submitted to the teacher through the online platform for assessment. The dataset is composed of all the interactions of the students with the teaching material, forum, quiz and assessment submission boxes.

Data Anonymization. In order to fulfil privacy requirements, the dataset has been anonymized before processing. Student name, ID, email and all the personal information has been removed to guarantee that the students cannot be identified.

Data Cleaning. Interactions of teachers and teaching assistants have been removed from the pool of data to be analysed as not needed for the purpose of this work.

Ground Truth. To evaluate and analyse the results of the group detection algorithms, the ground truth of the students' grouping has been collected.

Group project is one of the continuous assessment in this course. The students enrolled in this course form groups by themselves, and submit the group member list to the online learning platform. The scope of the group project covers multiple topics taught during the course. Hence, the students are required to have an overall view on all the course content. This submitted group member list is the ground truth, to be compared with the group information we predicted by our method.

Without losing generality, and for the scope of the assigned group projects, we can assume that students work together in groups to complete the assignment for all the tasks required during the course.

4.2 Preliminary Data Assessment

A preliminary analysis has been conducted to determine if the landscape of the gathered data can convey useful information for our task.

Figure 1 shows the students' interactions distribution during the course, grouped by hour. As can be seen, the interactions of the students are mainly gathered around the lesson days, to retrieve the lesson materials, and at the end of the course, to revise the course content before the exam. The most interesting information found in this figure is that the students use the eLearning system almost continuously during the course enactment. Their access the system, alone or in small groups, in offload moments can convey interesting information about their teamwork.

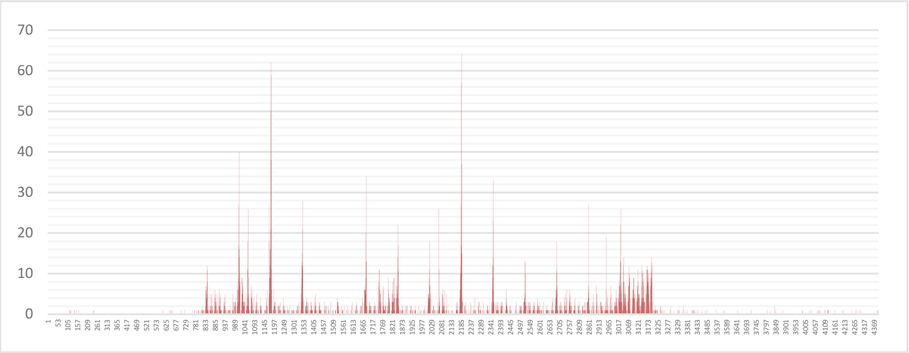

Fig. 1. Interactions distribution during course (1 h timeslots)

Figure 2 represents, for each student, the total number of interactions with the VLE. As shown all the students interacted with the eLearning environment, except one. The average of about 53 access for each student, with some peaks over 100. This means that each student, considering that the examined course lasted for 90 days, had one access every two days.

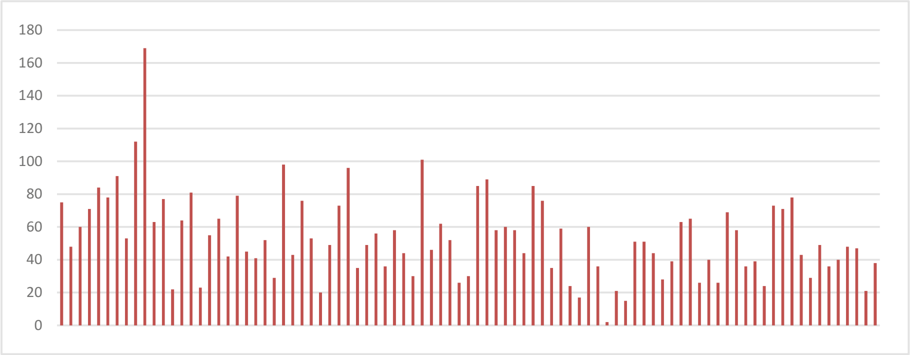

Fig. 2. Number of accesses by user during course

4.3 Graph Analysis

Experiments have been held using the dataset described in Sect. 4.1, extracting from it information about students' time of interaction and anonymized student IDs. The static point of view of the framework has been tested by analysing a single course. Students' co-occurrence of interactions is extracted using the granularity of 3600 s (1 h). The information about student-student interactions is analysed to elicit the communities of students using the modularity maximization algorithm described in next section. The weight on the edges is considered constant.

To compare with the ground truth, the parameters of the modularity maximization algorithm have been adjusted to retrieve the same number of communities as the number of students' groups submitted to the teacher.

4.4 Modularity Maximization

To discover the students groups in the interaction graph, we used the "Louvain method" described by Blondel et al. in "Fast unfolding of communities in large networks" [14]. This algorithm is particularly efficient even for very large size networks, with nodes in the order of hundreds of millions.

The method examines a weighted graph. At start each node is assigned to a different community. In first phase a community aggregation takes place. For each node, the considered node will join one of its neighbours' community if there is a gain in modularity. This procedure is repeated until there is no gain in modularity. The second phase consist in building a new graph composed of super-nodes that are the communities found in the first phase. The edges between super-nodes are weighted as sum of the weight of the links between nodes in the corresponding two communities. The process is repeated until there are no changes in the network structure and the maximum modularity is achieved. In Fig. 3. One step of modularity maximization algorithm. Figure 3 is shown one step of the modularity maximization algorithm. The final modularity class assignment obtained from this algorithm is considered as the community assignment of each student.

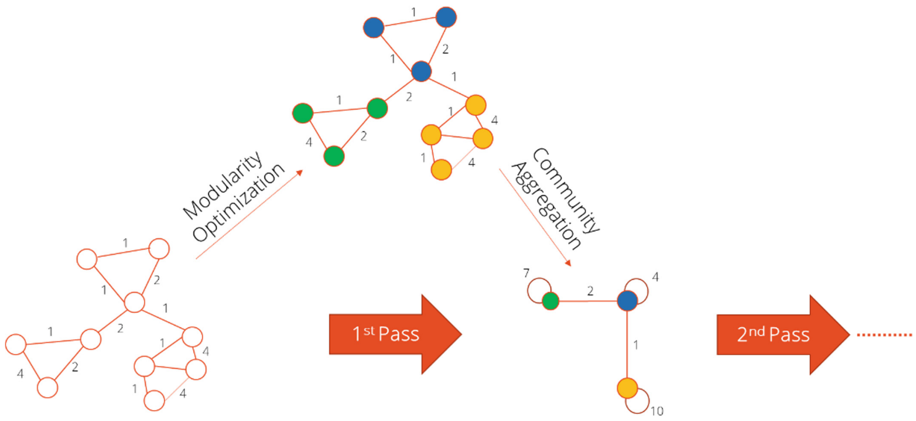

Fig. 3. One step of modularity maximization algorithm.

4.5 Evaluation Criteria

Modularity class assignments in graphs are the equivalent of cluster labelling assignments for clustering algorithms [34]. To evaluate the quality of the assignment, when using clustering methods, different measures can be used to highlight different qualities of the detected clusters. The same measures can be used for modularity class assignment by comparing the resulting classes with the ground truth assignments as defined in Sect. 4.1.

Homogeneity. Measures the quality of the modularity class label assignment by checking if the modularity class contain only elements of a single class in the ground truth assignment.

Completeness. Measures the quality of the modularity class label by checking if all the elements of a single class are assigned to the same modularity class.

V-measure. Measures how successfully the criteria of homogeneity and completeness have been satisfied [35]. It is computed as the harmonic mean of distinct homogeneity and completeness scores, like precision and recall are combined to compute the F-measure score [36].

5 Discussion

After retrieving the students' interaction graph from the VLE logs, we analysed the network to retrieve the students' communities.

The modularity class assignment has been evaluated using the criteria and the measures described in Sect. 4.5. The scores of the retrieved communities achieved 0.53518214 for homogeneity, 0.557139949 for completeness and 0.545940347 for V-measure respectively. In general, the class assignment produced assignments that put most of the students belonging to a single ground truth group together, even if they are aggregated with some members of other groups. In Table 1, the most exemplificative modularity class assignments are shown. Compared with the ground truth we can find that 3 elements of the original groups are assigned to the same community.

Table 1. Example of modularity class assignments with high completeness score.

Modularity class assignment	Ground truth group assignment
8	7
14	7
14	7
14	7
2	10
2	10
2	10
11	10
12	10
4	18
11	18
11	18
11	18
14	18

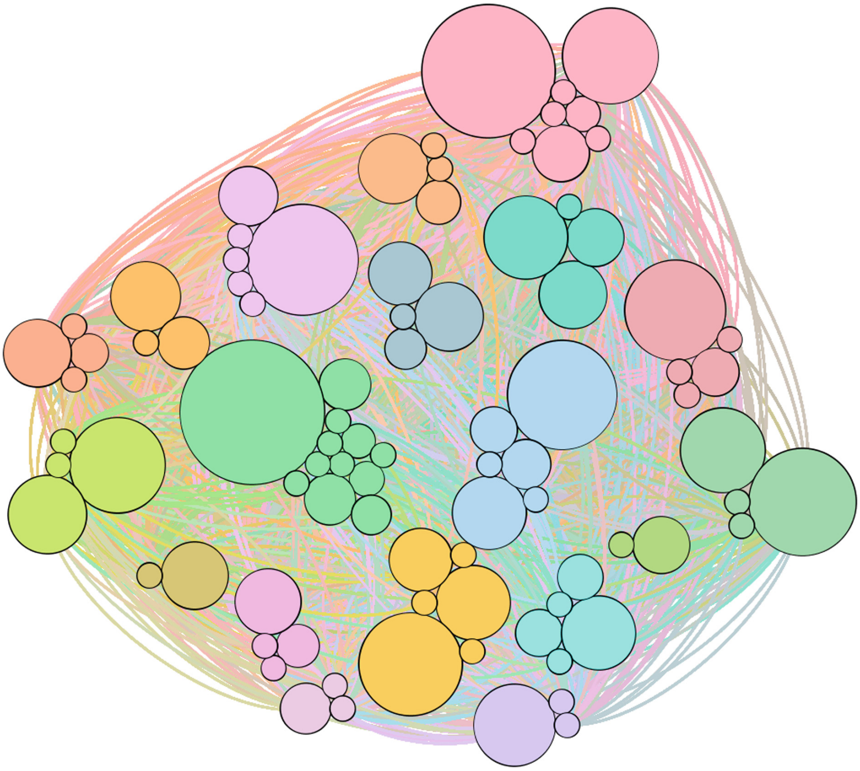

Fig. 4. Co-occurrence network.

Figure 4 shows the co-occurrence network resulting from the elaboration of our dataset. The graph is extremely dense as it is composed of over 2 million edges connecting 89 nodes. In the figure, the node colours represent the modularity class assignments and the node size represents its betweenness centrality measure. Basing on this measure, bigger nodes can be identified as leaders in the groups while smaller ones are followers. The visual representation is made by aggregating the followers of each community close to their leaders.

6 Conclusions and Future Work

To the best of our knowledge, this is the first time the graph-based student community elicitation is introduced in the VLEs.

The graph approach introduced in our work is able to identify the communities of students starting from co-occurrence of interactions in Virtual Learning Environments. The experiment results show that the students' groupings are close to the ground truth groupings in terms of homogeneity, completeness and V-measure.

Moreover, using our methodology, it's possible to distinguish between leaders and followers in students' groups.

Future works will include the implementation of a weight function for the edges of the graph that takes in account the length of interaction between students, the refinement of the leader-follower elicitation using time of interaction, and the study of grouping dynamics.

Acknowledgements. This work was partially supported by University of Perugia "Fondi Ricerca di Base 2015" project "Distributed algorithms for node ranking in complex and scale free network."

References

1. Fortunato, S.: Community detection in graphs. Phys. Rep. **486**(3–5), 75–174 (2010). https://doi.org/10.1016/j.physrep.2009.11.002
2. Kernighan, B.W., Lin, S.: An efficient heuristic procedure for partitioning graphs. Bell Syst. Tech. J. **49**(2), 291–307 (1970). https://doi.org/10.1002/j.1538-7305.1970.tb01770.x
3. Pothen, A.: Graph partitioning algorithms with applications to scientific computing. ICASE LaRC Interdiscip. Ser. Sci. Eng. **4**, 323–368 (1997). https://doi.org/10.1007/978-94-011-5412-3
4. Franzoni, V., Li, Y., Mengoni, P., Milani, A.: Clustering Facebook for biased context extraction. In: Gervasi, O., et al. (eds.) ICCSA 2017. LNCS, vol. 10404, pp. 717–729. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62392-4_52
5. Franzoni, V., Milani, A., Biondi, G.: SEMO. In: Proceedings of the International Conference on Web Intelligence - WI 2017, pp. 953–958 (2017). https://doi.org/10.1145/3106426.3109417
6. Franzoni, V., Biondi, G., Milani, A.: A web-based system for emotion vector extraction. In: Gervasi, O., et al. (eds.) ICCSA 2017. LNCS, vol. 10406, pp. 653–668. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62398-6_46

7. Biondi, G., Franzoni, V., Li, Y., Milani, A.: Web-based similarity for emotion recognition in web objects. In: Proceedings of the 9th International Conference on Utility and Cloud Computing - UCC 2016, pp. 327–332 (2016). https://doi.org/10.1145/2996890.3007883
8. Biondi, G., Franzoni, V., Poggioni, V.: A deep learning semantic approach to emotion recognition using the IBM Watson Bluemix Alchemy Language. In: Gervasi, O., et al. (eds.) ICCSA 2017. LNCS, vol. 10406, pp. 718–729. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62398-6_51
9. Franzoni, V., Li, Y., Mengoni, P.: A path-based model for emotion abstraction on Facebook using sentiment analysis and taxonomy knowledge. In: Proceedings of the International Conference Web Intell. - WI 2017, pp. 947–952 (2017). https://doi.org/10.1145/3106426.3109420
10. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning, vol. 1 (2009). https://doi.org/10.1007/b94608
11. Macqueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, no. 233, pp. 281–297 (1967)
12. Donath, W.E., Hoffman, A.J.: Lower bounds for the partitioning of graphs. IBM J. Res. Dev. **17**(5), 420–425 (1973). https://doi.org/10.1147/rd.175.0420
13. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proc. Natl. Acad. Sci. **99**(12), 7821–7826 (2002). https://doi.org/10.1073/pnas.122653799
14. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mech.: Theory Exp. **10**, 2008 (2008). https://doi.org/10.1088/1742-5468/2008/10/P10008
15. Guimerà, R., Sales-Pardo, M., Amaral, L.A.N.: Modularity from fluctuations in random graphs and complex networks. Phys. Rev. E. - Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top. **70**(2), 4 (2004). https://doi.org/10.1103/physreve.70.025101
16. Santucci, V., Baioletti, M., Milani, A.: An algebraic differential evolution for the linear ordering problem. In: Proceedings of the Companion Publication of the 2015 on Genetic and Evolutionary Computation Conference - GECCO Companion 2015, pp. 1479–1480 (2015). https://doi.org/10.1145/2739482.2764693
17. Baioletti, M., Milani, A., Santucci, V.: An extension of algebraic differential evolution for the linear ordering problem with cumulative costs. In: Handl, J., Hart, E., Lewis, P.R., López-Ibáñez, M., Ochoa, G., Paechter, B. (eds.) PPSN 2016. LNCS, vol. 9921, pp. 123–133. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45823-6_12
18. Castellano, C., Cecconi, F., Loreto, V., Parisi, D., Radicchi, F.: Self-contained algorithms to detect communities in networks. Eur. Phys. J. B **38**(2), 311–319 (2004). https://doi.org/10.1140/epjb/e2004-00123-0
19. Reichardt, J., Bornholdt, S.: Statistical mechanics of community detection. Phys. Rev. E **74** (1), 16110 (2006). https://doi.org/10.1103/PhysRevE.74.016110
20. Reichardt, J., White, D.R.: Role models for complex networks. Eur. Phys. J. B **60**(2), 217–224 (2007). https://doi.org/10.1140/epjb/e2007-00340-y
21. Dawson, S., Bakharia, A., Heathcote, E.: SNAPP : realising the affordances of real-time SNA within networked learning environments. In: Proceedings of the 7th International Conference on Networked Learning, pp. 125–133 (2010)
22. Bakharia, A., Dawson, S.: SNAPP. In: Proceedings of the 1st International Conference on Learning Analytics and Knowledge - LAK 2011, p. 168 (2011). https://doi.org/10.1145/2090116.2090144
23. Sheard, J., Ceddia, J., Hurst, J., Tuovinen, J.: Inferring student learning behaviour from website interactions: a usage analysis. Educ. Inf. Technol. **8**(2002), 245–266 (2003). https://doi.org/10.1023/A:1026360026073

24. Tasso, S., Pallottelli, S., Laganà, A.: Mobile device access to collaborative distributed repositories of chemistry learning objects. In: Gervasi, O., Murgante, B., Misra, S., Rocha, A.M.A.C., Torre, C., Taniar, D., Apduhan, B.O., Stankova, E., Wang, S. (eds.) ICCSA 2016. LNCS, vol. 9786, pp. 443–454. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-42085-1_34
25. Tasso, S., Pallottelli, S., Ciavi, G., Bastianini, R., Laganà, A.: An efficient taxonomy assistant for a federation of science distributed repositories: a chemistry use case. In: Murgante, B., Misra, S., Carlini, M., Torre, C.M., Nguyen, H.-Q., Taniar, D., Apduhan, B. O., Gervasi, O. (eds.) ICCSA 2013. LNCS, vol. 7971, pp. 96–109. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39637-3_8
26. Nguyen, Q., Huptych, M., Rienties, B.: Linking students' timing of engagement to learning design and academic performance : a longitudinal study. In: Proceedings of the 8th International Conference on Learning Analytics and Knowledge, pp. 141–150 (2018). https://doi.org/10.1145/3170358.3170398
27. Boroujeni, M.S., Dillenbourg, P.: Discovery and temporal analysis of latent study patterns from MOOC interaction sequences. In: Proceedings of the Eighth International Conference on Learning Analytics and Knowledge, pp. 206–215 (2018). https://doi.org/10.1145/3170358.3170388
28. Khosravi, H., Cooper, K.M.L.: Using learning analytics to investigate patterns of performance and engagement in large classes. In: Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education – SIGCSE 2017, pp. 309–314 (2017). https://doi.org/10.1145/3017680.3017711
29. Bron, C., Kerbosch, J.: Algorithm 457: finding all cliques of an undirected graph. Commun. ACM **16**(9), 575–577 (1973). https://doi.org/10.1145/362342.362367
30. Holland, P.W., Laskey, K.B., Leinhardt, S.: Stochastic blockmodels: first steps. Soc. Netw. **5** (2), 109–137 (1983). https://doi.org/10.1016/0378-8733(83)90021-7
31. Bailly-Bechet, M., et al.: Finding undetected protein associations in cell signaling by belief propagation. Proc. Natl. Acad. Sci. **108**(2), 882–887 (2011). https://doi.org/10.1073/pnas.1004751108
32. Peixoto, T.P.: Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models. Phys. Rev. E - Stat. Nonlinear Soft Matter Phys. **89**(1), 018204 (2014). https://doi.org/10.1103/physreve.89.012804
33. Brandes, U., et al.: Maximizing modularity is hard, no. 1907 (2006)
34. Mengoni, P., Milani, A., Li, Y.: Clustering students interactions in eLearning systems for group elicitation. In: Gervasi, O., et al. (eds.) ICCSA 2018, LNCS, vol. 10962, pp. xx–yy (2018)
35. Rosenberg, A., Hirschberg, J.: V-measure: a conditional entropy-based external cluster evaluation measure. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, vol. 1, no. June, pp. 410–420 (2007). https://doi.org/10.7916/d80v8n84
36. Van Rijsbergen, C.J.: Information Retrieval. Inf. Retr. Boston, 208 (1979). https://doi.org/10.1007/springerreference_16360