Noname manuscript No.
(will be inserted by the editor)

Feature Representation for Social Circles Detection Using MAC

Jesús Alonso · Roberto Paredes · Paolo Rosso

Received: date / Accepted: date

Abstract Social circles detection is a special case of community detection in social network that is currently attracting a growing interest in the research community. In this paper we propose an empirical evaluation of the multi-assignment clustering method using different feature representation models. We define different vectorial representations from both structural egonet information and user profile features. We study and compare the performance on two available labelled Facebook datasets and compare our results with several different baselines. In addition, we provide some insights of the evaluation metrics most commonly used in the literature.

Keywords social circles detection · community detection · feature representation · multi-assignment clustering · evaluation metrics

1 Introduction

Nowadays, users in social networks tend to organize the contacts in their personal networks by means of social circles, a tool already implemented by the major companies, like for instance Facebook lists or Google+ circles. However, this labelling is still mostly done manually and, therefore, a growing interest has risen in the automatic detection of these circles. In addition, this problem is related to the more general task of community detection in graphs, or the identification of subnetworks in a given network. The main difference between both problems is the use of information from users' profiles (node attributes), apart from the network structure itself.

Despite the lack of a precise and well-accepted definition of community, there is a wide variety of methods and techniques designed to cope with community detection [9,26]. Moreover, some techniques specifically designed for social circles detection have been developed recently [17,20]. In this article, we continue the research started in [1], based on multi-assignment clustering (MAC) [29,10]. MAC is a clustering technique not necessarily related to networks or graphs, which permits to assign the same object into several different clusters or, in our case, social circles. In [1], we conducted preliminary experiments using this method, defining several vectorial representations of both structural network information and users' profile information, with the aim to investigate which of these representations provides the best results. In this new study, we expand the experimentation, making variable the number of users' profile features employed. Unlike in [1], we allow for the clustering technique to automatically detect the number of predicted circles, instead of using the number of groundtruth circles. We test our method on two different datasets and evaluate it by means of measures present in the literature. In this regard, we provide a critical commentary of the evaluation metrics, as we believe they have some flaws to which we need to pay attention. We compare our results to the ones provided by the state-of-the-art method in [17,20].

J. Alonso
PRHLT Research Center, Universitat Politècnica de València, Valencia, Spain
E-mail: jealnan@dsic.upv.es

R. Paredes
PRHLT Research Center, Universitat Politècnica de València, Valencia, Spain
E-mail: rparedes@dsic.upv.es

P. Rosso
PRHLT Research Center, Universitat Politècnica de València, Valencia, Spain
E-mail: prosso@dsic.upv.es

The rest of the paper is structured as follows. In Section two, we present previous works on community detection and social circles detection. In Section three, we describe thoroughly our methodology, including the different data representations proposed. In Section four, we present the datasets, the evaluation measures and the results of our experiments. Finally, we draw some conclusions.

2 Previous Work

2.1 Community Detection in Networks

From an abstract point of view, a network is equivalent to a graph, defined by a set of nodes connected by edges. However, the concept of network has additional connotations. Networks can represent real structures such as social networks, biological networks (neural synaptic networks, metabolical networks), technological networks (the Internet, the World Wide Web), logistic networks (distribution networks), etc. There is no well-accepted formal definition of community in general networks. However, there is a consensus on the fact that it consists of a group of nodes that are more densely connected to each other than to the nodes outside. The relation of membership in a community usually has an extra meaning, and the vertices in a community share common properties or play similar roles within the graph.

Community detection is the task of automated identification of the communities of a network. A considerable number of methods have been developed to solve this problem [9,26]. The classical methods are classified in four categories: graph partitioning [15,30], hierarchical clustering [12], partitional clustering [19,18] and spectral clustering [7]. Another family of algorithms is based on the optimization of a clustering quality index known as modularity. Modularity optimization is an NP-complete problem [3] but there are fairly good approximations which perform in a reasonable time [11, 23,2,22].

In real networks, nodes are often shared among different communities. The most popular technique to detect overlapping communities is the clique percolation method [24]. Given a graph, a k-clique is defined as a complete subgraph of size k. Clique percolation consists in the identification of k-clique communities, defined as the union of all k-cliques that can be reached from each other through a series of adjacent k-cliques. Despite of the good performance of this technique, clique percolation remains a hard computational problem, new and improved implementations still scale worse than some other overlapping community finding algorithms. Some alternative algorithms that detect overlapping clusters have been developed recently, such as the multi-assignment clustering technique serving as the prediction method in this article [29,10].

2.2 Social Networks and Social Circles Detection

The study of social networks is a research topic with a history of decades and it has been recently revitalized by the appearance of new information and communication technologies which have opened new ways of interacting. Clustering within social networks has been studied designing several procedures. Some approaches base the clustering on the network links [26], while others consider the semantic content of social interactions [33]. In between both methodologies, there has also been work on combining the links and the content for doing the clustering [25,28]. Very recently, a new technique studied the characteristics of community structures formed around topical discussion clusters, using modularity maximization algorithms [6].

Social circles detection constitutes a particular case of social clustering. Within a social network, an ego network or egonet is defined as the subgraph of the contacts of a particular user (called the ego). Thus, it includes all the contacts of the ego (named the alters) and the contact relationship between every pair of them. Then, the social circles of an ego can be considered as clusters of the egonet. Social circles may overlap (share nodes), for example university friends who were high school friends as well; and they may also present hierarchical inclusion (the nodes of a circle totally included into another), for example university friends into a generic friends category. In addition to the graph structure itself, node attributes from the users' profiles may be used as a source of information, as well. Some current, successful work in social circles detection involves a generative model that considers circle memberships and a circle-specific profile similarity metric [17,20]. However, other approaches are being considered, such as the use of MAC [29,10] for circle prediction. In [17,20], some tests are performed with this idea, but using only the node attributes, no information from the graph structure. In [1], MAC was fed with representations of both the graph structure and the node attributes, with successful results. The main aim of this paper is to focus on users' profiles features and to conduct an exhaustive study on how many of them are necessary to obtain the best results.

3 Methodology

3.1 Multi-Assignment Clustering

Multi-Assignment Clustering [29,10] is a clustering method, originally developed for Boolean vectorial data, which allows for the possibility to assign the same object into several different clusters. It provides a decomposition of the data matrix $\mathbf{X}$ into a matrix containing the clusters prototypes $\mathbf{Z}$ and a matrix representing the degree to which a particular data vector belongs to the different clusters $\mathbf{Y}$. Finding optimal matrices $\mathbf{Z}$ and $\mathbf{Y}$ is NP-hard [31], but a probabilistic representation allows to drastically simplify the optimization problem. In [29] the authors propose a mixture model where X_{ij} is either drawn from a signal or a noise component. The probability of X_{ij} under the signal model is the following, being β_{kj} independent random variables for the deterministic centroids $\mathbf{Y}$:

$$p(X_{ij}|\mathbf{Z},\beta) = \left[1 - \prod_{k=1}^{K} \beta_{kj}^{Z_{ik}}\right]^{X_{ij}} \left[\prod_{k=1}^{K} \beta_{kj}^{Z_{ik}}\right]^{1-X_{ij}}, \quad (1)$$

where $\beta_{kj} := p(Y_{kj} = 0)$

In addition to the signal model, there is a noise model for the difference between the original data and the reconstruction made from $\mathbf{Z}$ and $\mathbf{Y}$. The model parameters are inferred by deterministic annealing [27,4]. The computational cost of MAC is proportional to the parameters, β and $\mathbf{Z}$, and we have observed empirically that the execution time of MAC is normally lower than that of other methods such as clique percolation. When MAC is applied to social circles detection, $\mathbf{Y}$ is the matrix indicating which users belong to the different clusters, social circles.

Several reasons helped us making the choice of MAC over alternative methods for our study. First of all, social circles detection is a soft clustering task [9] in which single data points are assigned into multiple social circles; and MAC, unless other fuzzy clustering strategies, provides hard, binary assignments into different clusters, instead of fractional assignments with different membership levels. In addition, MAC provides a noise channel along with the signal channel, which gives important information within the task of social circles detection. Other techniques, such as Gaussian mixture models [8], do not model noise. Furthemore, MAC is more adequate for large networks than other methods with a very high computational cost, like clique percolation. Finally, MAC is a state-of-the-art technique, having recent and influential publications, such as [17,20], in which it was employed and considered as a baseline method for social circles detection. In this work, we continue the research started in [1] and explore further the possibilities of MAC for this task, investigating novel representations. We defend the fact that this technique still has further potentiality and better results can be obtained. A simple example of the performance of MAC can be seen in Figure 1.

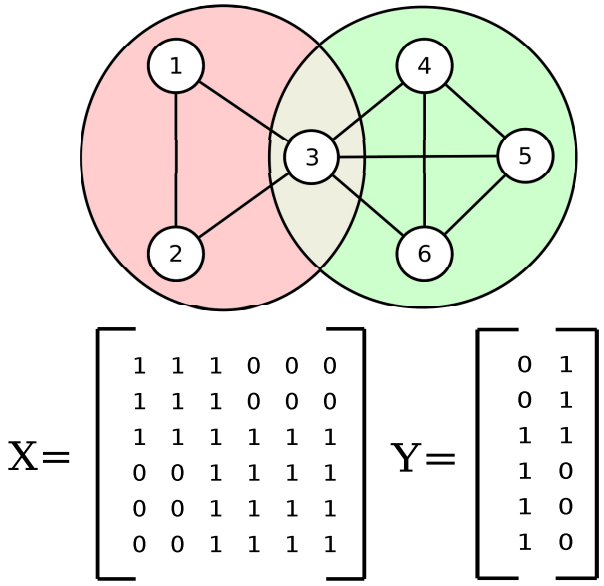

Fig. 1 Example of the performance of MAC on a simple graph. The data matrix, $\mathbf{X}$, which serves as input to the method, is the adjacency matrix of the graph. $\mathbf{Y}$ is the output matrix showing which nodes belong to any of the two clusters. Note that MAC identifies correctly the two cliques that constitute the graph, including the overlapping node, 3.

As a novelty, we model the structural information of the egonets into diverse vectorial representations ready to be supplied to the algorithm. Several vectorial representations for user profile features were developed as well. Unlike the original MAC, we allow the input to be real data in $[0,1]^n$ as a way to model a hierarchy of link levels in the case of structural information, or an aggregation of the number of feature values shared by two users profiles in the case of user profile information.

In all the experiments, the input data matrix $\mathbf{X}$ is a horizontal concatenation of a matrix $\mathbf{S}$, containing structural network information, and a matrix $\mathbf{P}$, containing profile features information: $\mathbf{X} = [\mathbf{S} \,|\, \mathbf{P}]$. Rows represent users of the egonet and, therefore, for every user u there is a row vector of structural network information, $\mathbf{S}_u$, and a row vector of profile features information, $\mathbf{P}_u$. Therefore, the number of rows of the matrix $\mathbf{X}$ is the number of users in the ego-network $|\, u \,|$, and the number of columns of the matrix $\mathbf{X}$ is the

total number of features used to represent structural and profile information of each user.

3.2 Structural Network Representation

Structural network information is complete, and every edge in the egonets is contained in the dataset. In this subsection, we present the different representations of the structural network information that have been considered. All of them transform graph links into a matrix **S**. We use the following concepts:

- *Friendship ranks:* when there is a link between two users, we say they are direct friends or rank 1 friends. When two users are not direct friends but have a common direct friend, we say they are rank 2 friends. Friendship ranks of greater levels can be further defined. In this study we consider up to rank 3 friends, as we have noticed that greater levels of friendship introduce a vast amount of noise. There is a column in **S** for every friendship rank and user in the egonet. An element of **S** is 1 if the row user and the column user are friends of such rank, and 0 otherwise. Obtaining in total $3\times \mid u \mid$ structural features for each user.
- *Weighting:* the data is weighted depending on the friendship rank it represents. Rank 1 friendship is left with 1, whereas rank 2 friendship is weighted to 0.5 and rank 3 friendship is weighted to 0.25. Like in the previous case, obtaining in total $3\times \mid u \mid$ structural features for each user.
- *Aggregation:* for every user, the different friendship ranks are aggregated into just one value. This is obtained by calculating the maximum weighted friendship rank. Reducing the number of structural features to $\mid u \mid$.

The graphs considered in our experiments are not oriented, and thus all our matrices **S** are horizontal concatenations of symmetric matrices. However, the method would be equally used for oriented graphs, which would result in non-symmetric matrices.

3.3 Users' Profile Representation

There are up to 57 profile features for every user in the data corpus we used for the experiments. Nevertheless, in contrast to the structural network information, some of them contain blanks for the majority of users, providing little useful information. Other profile features are redundant or not relevant for the task. Thus, we have selected the most informative features and we use only these. We understand them as the ones which have information for the greatest number of users and that are the most relevant for the social circles detection task. Some examples of these features are, in order of importance: hometown, schools, employers, gender and birthday. We conduct our experiments with different subsets of these features: the most important, the 2 most important, the 3 most important,... All the features can take different discrete values from a finite set.

We define as $\mid f \mid$ the number of features considered, and as $\mid v \mid$ the total number of values of the considered features that are taken by at least one user in the egonet. We encode the profile features information in the matrices **P**, for which the following representations have been defined:

- *Explicit:* There is a column of **P** for every different value of the considered features. An element of **P** is 1 if the row user takes the column value for the respective feature, and 0 otherwise. Obtaining in total $\mid v \mid$ profile features for each user.
- *Intersection:* There is one column of **P** for every user in the egonet and every considered profile feature. An element of **P** is 1 if the sets of values of the row user and the column user, for that particular feature, intersect. It is 0 otherwise. In this case, obtaining $\mid f \mid \times \mid u \mid$ profile features for each user.
- *Weighted:* There is just one column of **P** for every user in the egonet. An element of **P** represents the proportion of features for which the row user and the column user share at least one value. It is calculated as $\frac{|s|}{|f|}$, where $\mid s \mid$ is the number of features shared between both users. Reducing the number of profile features to $\mid u \mid$.

4 Experiments

4.1 Dataset

We use two data corpora for the experiments. The first one is the training part of the dataset published for the Kaggle competition on learning social circles in networks [14]. The second one is the ego-Facebook dataset of social circles from Facebook from the Stanford Large Network Dataset Collection [16], used in [17,20]. In both cases, the data consist of hand-labelled friendship egonets from Facebook and a set of up to 57 profile features for every node in those networks. Some statistics for both datasets are shown in Table 1.

The degree of a given user is defined as the number of different circles which it belongs to. MAC takes as a parameter the range of possible degrees of the users of an egonet. In all our experiments the minimum degree

Table 1 Statistics of the data corpora

	Kaggle	ego-Facebook
Egonets	60	10
Users in smallest egonet	45	59
Users in largest egonet	670	1045
Total users	14519	4167
Connections	348131	88234
Circles	592	193

is set to 0 and the maximum degree is set to 3. In this regard, unlike previous studies, we do not include any prediction technique for the number of circles within the egonets, but we predict in every case a fixed number of 35 circles. We rely on MAC to leave empty the extra circles.

4.2 Evaluation Metrics

The aim of any evaluation metric $E(\mathcal{C}, \bar{\mathcal{C}})$ defined for the task is to measure the similarity between the set of predicted circles $\mathcal{C} = \{C_1, \ldots, C_K\}$ and the set of groundtruth circles $\bar{\mathcal{C}} = \{\bar{C_1}, \ldots, \bar{C_K}\}$. In this regard, two main approaches appear in the references. One of them is based on the definition of a similarity score $s(C, \bar{C})$ between two circles, with a further calculation of the best alignment between $\mathcal{C}$ and $\bar{\mathcal{C}}$. The other is based on an edit distance between $\mathcal{C}$ and $\bar{\mathcal{C}}$. In each case described below, the final evaluation measure is the average of the evaluation measures obtained for all the egonets in the respective dataset.

Within the first approach, the similarity measures s can perfectly be well-established similarity metrics between sets. In the references, the Jaccard coefficient [13], the F-measure [17] and the Balanced Error Rate [5] have been used. Their performance is similar, and we have decided to report the F-measure in this article. It is calculated as:

$$F(C, \bar{C}) = 2 \times \frac{\text{precision}(C, \bar{C}) \times \text{recall}(C, \bar{C})}{\text{precision}(C, \bar{C}) + \text{recall}(C, \bar{C})} \quad (2)$$

$$\text{being precision}(C, \bar{C}) = \frac{\mid C \cap \bar{C} \mid}{\mid C \mid},$$

$$\text{and recall}(C, \bar{C}) = \frac{\mid C \cap \bar{C} \mid}{\mid \bar{C} \mid}$$

Two different alignments between sets of circles are defined in the references. The first one is described in [32] as follows: every detected circle is matched with its most similar groundtruth community, and the performance is computed. After that, every groundtruth community is matched with its most similar predicted community, and the performance is computed again. The final evaluation function is the average of the two performance measures:

$$E_b(\mathcal{C}, \bar{\mathcal{C}}) = \frac{1}{2|\bar{\mathcal{C}}|} \sum_{\bar{C}_i \in \bar{\mathcal{C}}} \max_{C_j \in \mathcal{C}} s(\bar{C}_i, C_j) + \frac{1}{2|\mathcal{C}|} \sum_{C_j \in \mathcal{C}} \max_{\bar{C}_i \in \bar{\mathcal{C}}} s(\bar{C}_i, C_j) \quad (3)$$

This average is done because matching only from one side leads to degenerate optimal performance (for example, outputting all possible subsets of nodes as detected communities would achieve perfect matching groundtruth communities to the detected ones). However, this measure is too optimistic, several groundtruth circles can be aligned to just one predicted circle or vice versa, without any penalization to non-aligned predicted or groundtruth circles.

In [17,20] the alignment is defined as an optimal correspondence via linear assignment, found by means of the Hungarian algorithm [21]:

$$E_h(\mathcal{C}, \bar{\mathcal{C}}) = \max_{f: \mathcal{C} \to \bar{\mathcal{C}}} \frac{1}{|f|} \sum_{C \in \text{dom}(f)} (1 - s(C, f(C))$$

This approach ensures that, unlike in the previous case, there are no cases of single-to-multiple circles alignment. Nevertheless, the use of the Hungarian algorithm makes the set having the smaller number of circles to have all its circles aligned, whereas the other set will always have a number of $\max(|\mathcal{C}|, |\bar{\mathcal{C}}|) - \min(|\mathcal{C}|, |\bar{\mathcal{C}}|)$ circles without being aligned at no cost. Sadly, this leads to degenerate optimal performance, as having all possible subsets of nodes as detected communities would achieve a perfect matching. In [17,20], the authors state that this kind of undesirable behaviour only happens when the number of predicted circles is greater than the number of groundtruth circles. However, it presents other problems, as well. For instance, predicting only one perfect circle would have a perfect matching as well, forgetting that the rest of groundtruth circles remain without prediction.

The use of an edit distance as an evaluation measure for the task was introduced at the Kaggle competition on learning social circles in networks [14]. This distance, $E_d(\mathcal{C}, \bar{\mathcal{C}})$, has four basic edit operations: adding a user to an existing circle, creating a circle with one user, removing a user from a circle and deleting a circle with one user; every one of them at cost 1. We believe that it is the most complete and accurate evaluation measure of the ones described, as it is a global measure between sets of circles, it does not consider single-to-multiple circles alignments and it does not lead to degenerate optimal performance.

Table 2 Baselines and results of the experiments on the Kaggle and ego-Facebook datasets. We report results evaluated by the E_b, E_h and E_d evaluation measures. The prediction is closer to the groundtruth when the values of E_b and E_h are higher or the value of E_d is lower.

		Dataset					
		Kaggle			ego-Facebook		
Baseline		E_b	E_h	E_d	E_b	E_h	E_d
Method in [17,20]		0.4714	0.5739	267.23	0.3899	0.5335	502.40
Empty circles		*	*	285.02	*	*	423.30
All in one circle		0.3771	0.5318	352.67	0.3330	0.5242	570.80
Max. friendship rank	N. profile features	E_b	E_h	E_d	E_b	E_h	E_d
1	1	0.4133	**0.4994**	425.95	0.3778	0.4225	686.20
	2	0.4165	0.4879	342.18	**0.3935**	**0.4439**	432.70
	3	0.4078	0.4852	381.60	0.3696	0.3752	598.30
	4	0.4265	0.4326	283.22	0.3353	0.3639	486.30
	5	**0.4324**	0.4469	285.68	0.3601	0.3858	477.20
2	1	0.3135	0.3600	271.15	0.2916	0.4234	405.50
	2	0.2934	0.3664	263.28	0.1535	0.2321	414.60
	3	0.2979	0.3710	**261.47**	0.1848	0.2670	**401.30**
	4	0.2114	0.3001	271.57	0.1192	0.1846	415.00
	5	0.2481	0.3501	269.35	0.1301	0.2062	415.80
3	1	0.1960	0.2311	280.83	0.1086	0.1496	415.00
	2	0.1622	0.2041	277.27	0.1078	0.1783	420.00
	3	0.1727	0.2250	275.75	0.0825	0.1427	420.30
	4	0.1317	0.2007	280.00	0.0376	0.0580	420.90
	5	0.1388	0.2093	279.92	0.0536	0.0939	421.00

4.3 Results

One of the main objectives of this article is to compare our results to the state-of-the-art technique for social circles detection described in [17,20]. However, we do not have access to the optimal parameter values of the method and, therefore, we cannot replicate the exact results reported in those works. We have tried several parameter values, and report the best performing results among these tests.

In addition, we compare our results to two simple baselines which perform surprisingly well under certain evaluation metrics. The first of these baselines consists in defining an empty set of circles, $\mathcal{C} = \emptyset$. It can be evaluated only by the E_d evaluation measure, as there is no possible alignment between the set of groundtruth circles $\bar{\mathcal{C}}$ and an empty set. However, it is interesting to report this baseline, as the E_d evaluation measure heavily penalizes the misclassification of users into circles. Thus, defining no circle at all performs better than other simple baselines.

The second baseline consists in an only circle composed by all the alters in the egonet. This baseline performs especially well when evaluated by the E_h measure. The reason is that the greatest groundtruth circle gets aligned with the circle provided and the F-measure between them is the reported result. The larger this groundtruth circle, the better the performance of this baseline. The rest of groundtruth circles, without an aligned predicted circle, bring no penalty to that result.

The results obtained by the baselines and our experiments are shown in Table 2. We evaluate the performance with the 3 evaluation measures defined in the previous subsection: E_b (best matching), E_h (Hungarian matching) and E_d (edit distance). We employ in every case aggregated structural network representations, having maximum friendship ranks 1, 2 or 3. We use only weighted users' profile representations as well, containing information from 1 to 5 profile features. We have conducted this selection due to the results of preliminary experiments, and do not report results using more than 5 profile features, as they perform worse. The best results are very different depending on whatever evaluation measure is used. When employing the E_d evaluation measure, based on an edit distance, one particular feature representation, the one containing up to friendship rank 2 and 3 profile features, obtains the best results in both datasets. When employing the evaluation measures based on alignments, E_b or E_h, there is no single best-performing feature representation. However, the combination of a rank 1 structural network representation (equivalent to the adjacency matrix of the egonet) and a low number of profile features gives better results. The result obtained with a rank 1 structural network representation and 5 profile features, evaluated by the E_b evaluation measure, constitutes an anomaly

to this, probably explained by the extremely optimistic behaviour of E_b.

The best results that we have obtained with the E_d evaluation measure are more accurate than both the method in [17,20] and the two baselines. In the case of the ego-Facebook dataset, our results improve by a wide margin the ones obtained by the method in [17,20]. However, when using the evaluation measures based on alignments, our predictions are not generally more accurate than either the method in [17,20] or the baselines. Nevertheless, in these cases, the performance decrease suffered by our method is moderate. Thus, we conclude that MAC can obtain acceptable results when being evaluated by the measures based on alignments, while improving significantly the results when using the edit distance. However, the behaviour of the evaluation measures seems to be quite odd. While the three of them are designed to evaluate the similarity between a pair of sets of circles, the best-performing feature representations for the measures based on alignments provide bad results for the edit distance, and vice versa. This behaviour and the adequacy of the different evaluation measures for the task need to be further studied.

5 Conclusions

MAC, as a prediction technique for community detection in social network, was already used in [17,20]. In that work, only the users' profile information was used for prediction and its performance was lower than the one obtained with the method proposed by the authors. In this article we have proved that, provided that structural network information is incorporated and modelled in the right way, MAC can also constitute a valid technique for social circles detection. We also remark the wide disparity of the results when the different evaluation measures defined for the task are employed. Some of these metrics require alignments that present some flaws, leading to degenerate optimal performance.

There are several possible extensions of the work presented here. One of them is to improve the use of the profiles. We have conducted our experiments on the 5 profile features that we considered the most informative. However, extra tests could be done incorporating some of the less informative features, or using them alone, especially for the Kaggle dataset. In addition, new representations might be defined. Network structure representation could be enhanced, for example, using representations able to capture cycles. Moreover, some new features could be extracted from the network structure. They include node centrality measures such as the eigenvector and betweenness. Finally, we also consider the fusion of the method in [17,20] and MAC for prediction, making profit of the beneficial aspects of both techniques.

Acknowledgements This work was developed in the framework of the W911NF-14-1-0254 research project Social Copying Community Detection (SOCOCODE), funded by the US Army Research Office (ARO). The work of the first author is financed by grant FPU14/03483, from the Spanish Ministry of Education, Culture and Sport.

References

1. Alonso, J., Paredes, R., Rosso, P.: Empirical evaluation of different feature representations for social circles detection. In: Pattern Recognition and Image Analysis, *Lecture Notes in Computer Science*, vol. 9117, pp. 31–38. Springer International Publishing (2015). `http://dx.doi.org/10.1007/978-3-319-19390-8_4`
2. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mech. Theor. Exp. **2008**, P10,008 (2008)
3. Brandes, U., Delling, D., Gaertler, M., Grke, R., Hoefer, M., Nikoloski, Z., Wagner, D.: On modularity-NP-completeness and beyond. Tech. Rep. 2006-19, ITI Wagner, Faculty of Informatics, Universität Karlsruhe (TH), Germany (2006)
4. Buhmann, J., Kuhnel, H.: Vector quantization with complexity costs. IEEE Transactions on Information Theory **39**(4), 1133–1145 (1993)
5. Chen, Y., Lin, C.: Combining SVMs with various feature selection strategies. In: Feature Extraction, pp. 315–324 (2006)
6. Dey, K., Bandyopadhyay, S.: An empirical investigation of like-mindedness of topically related social communities on microblogging platforms. In: International Conference on Natural Languages (2013)
7. Donath, W.E., Hoffman, A.J.: Lower bounds for the partitioning of graphs. IBM Journal of Research and Development **17**(5), 420–425 (1973)
8. Everitt, B.S., Hand, D.J.: Finite Mixture Distributions. Chapman and Hall (1981)
9. Fortunato, S.: Community detection in graphs. Physics Reports **486**(3), 75–174 (2010)
10. Frank, M., Streich, A.P., Basin, D., Buhmann, J.M.: Multi-assignment clustering for boolean data. The Journal of Machine Learning Research **13**(1), 459–489 (2012)
11. Girvan, M., Newman, M.E.: Community structure in social and biological networks. Proc. Natl. Acad. Sci. **99**(12), 7821–7826 (2002)
12. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer-Verlag (2009)
13. Jaccard, P.: Nouvelles recherches sur la distribution florale. Bulletin de la Socit Vaudoise des Sciences Naturelles **44**(163), 223–270 (1908)
14. Kaggle: Learning social circles in networks. `http://www.kaggle.com/c/learning-social-circles`
15. Kernighan, B.W., Lin, S.: An efficient heuristic procedure for partitioning graphs. Bell Syst. Tech. J. **49**(2), 291–307 (1970)
16. Leskovec, J., Krevl, A.: SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data` (2014)

17. Leskovec, J., Mcauley, J.: Learning to discover social circles in ego networks. In: F. Pereira, C. Burges, L. Bottou, K. Weinberger (eds.) Advances in Neural Information Processing Systems 25, pp. 539–547. Curran Associates, Inc. (2012)
18. Lloyd, S.: Least squares quantization in PCM. IEEE Transactions on Information Theory **28**(2), 129–137 (1982)
19. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proc. Fifth Berkeley Symp. on Math. Statist. and Prob.: Volume 1, pp. 281–297 (1967)
20. McAuley, J., Leskovec, J.: Discovering social circles in ego networks. ACM Transactions on Knowledge Discovery from Data (TKDD) **8**(1), 4 (2014)
21. Munkres, J.: Algorithms for the assignment and transportation problems. Journal of the Society for Industrial & Applied Mathematics **5**(1), 32–38 (1957)
22. Newman, M.E.: Modularity and community structure in networks. Proc. Natl. Acad. Sci. **103**(23), 8577–8582 (2006)
23. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E Stat. Nonlin. Soft. Matter. Phys. **69**(2), 026,113 (2014)
24. Palla, G., Derényi, I., Farkas, I., Vicsek, T.: Uncovering the overlapping community structure of complex networks in nature and society. Nature **435**(7043), 814–818 (2005)
25. Pathak, N., DeLong, C., Banerjee, A., Erickson, K.: Social topic models for community extraction. In: The 2nd SNA-KDD Workshop (2008)
26. Porter, M.A., Onnela, J.P., Mucha, P.J.: Communities in networks. Notices Amer. Math. Soc. **56**(9), 1082–1097 (2009)
27. Rose, K., Gurewitz, E., Fox, G.C.: Vector quantization by deterministic annealing. IEEE Transactions on Information Theory **38**(4), 1249–1257 (1992)
28. Sachan, M., Contractor, D., Faruqie, T.A., Subramaniam, L.V.: Using content and interactions for discovering communities in social networks. In: Proceedings of the 21st international conference on World Wide Web, pp. 331–340 (2012)
29. Streich, A.P., Frank, M., Basin, D., Buhmann, J.M.: Multi-assignment clustering for boolean data. In: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 969–976 (2009)
30. Suaris, P.R., Kedem, G.: An algorithm for quadrisection and its applications to standard cell placement. IEEE Transactions on Circuits and Systems **35**(3), 294–303 (1988)
31. Vaidya, J., Atluri, V., Guo, Q.: The role mining problem: Finding a minimal descriptive set of roles. In: Proceedings of the 12th ACM Symposium on Access Control Models and Technologies, pp. 175–184 (2007)
32. Yang, J., McAuley, J., Leskovec, J.: Community detection in networks with node attributes. In: IEEE 13th International Conference on Data Mining (ICDM), pp. 1151–1156. IEEE (2013)
33. Zhou, D., Councill, I., Zha, H., Giles, C.L.: Discovering temporal communities from social network documents. In: Seventh IEEE International Conference on Data Mining, pp. 745–750 (2007)