Journal of Universal Computer Science, vol. 23, no. 9 (2017), 907-931
submitted: 31/7/15, accepted: 26/9/17, appeared: 28/9/17 © *J.UCS*

Using Content to Identify Overlapping Communities in Question Answer Forums[1]

Mohsen Shahriari, Sabrina Haefele, Ralf Klamma
(Advanced Community Information Systems (ACIS)
RWTH Aachen University, Germany
lastname@dbis.rwth-aachen.de)

Abstract: Nowadays, people use online social networks almost every day. They activate either due to their interests, or to search or catch their desirable information. Users of online social networks generate structural and contextual traces that can be analyzed by, i.e., network science researchers. Researchers can describe networks fabricated out of online traces from different perspectives that one of them is communities. Overlapping communities are overlapped structures, in which nodes have denser connections with each other than the rest of the network. Different approaches have addressed this problem; however, few analyses and methods have focused on contextual traces generated by users. As such, in this paper, we propose an algorithm that uses actual content produced by users. This algorithm uses term frequency of words generated by users and combines them by an extended clustering technique. Our evaluation results compare the proposed content-based community detection with structural-based methods. We also reveal community properties as well as its relation to contextual information. Administrators can use these algorithms in question & answer forums where the explicit links among users are missing.

Key Words: Overlapping community detection, Question & answer forums, Content-based community detection, Network science
Category: H.2, H.3.7, H.5.4

1 Introduction

Online social networks have received increasing attention recently. People join these platforms due to their interests. While participating in online social networks, they generate a tremendous amount of data. Often, researchers apply data mining and machine learning methods to analyze the data and to find useful patterns. Networks are non-detachable part of modern life, in which we can see various kinds of them in everyday life, i.e., social networks, forums, and citation networks. We can understand that social networks are connected to people's life. For instance, students might use social networks, e.g., Facebook, to communicate to each other or people may use social networks like LinkedIn to find jobs. We can describe networks from different perspectives. Shrinking diameter,

[1] This is an extended version of the paper Contextualized versus Structural Overlapping Communities in Social Media, presented at the First Workshop on Recommender Systems and Big Data Analytics co-located with I-KNOW'2016 in Graz, Austria, October 2016

small-world-ness, temporality, motifs and community structures are among essential properties of complex networks as well as social networks [Leskovec et al., 2007, Milo et al., 2002, Cazabet et al., 2010].

Community structures can be observed in networks constructed out of users' interaction, in which vertices have a higher level of connection inside community than the rest of the network. For instance, people who are interested in the same Web pages or who perform the same social activities have similar interests, innovations, and opinions, and thus one may consider these nodes as belonging to the same community [Li et al., 2012, Xie et al., 2011]. In a big picture, networks face temporal changes in the structure. Similarly, in a smaller picture, communities as well encounter some temporal changes. As such, people may leave a community or join new ones as they search for their interests and activities [Palla et al., 2009, Greene et al., 2010]. Research has proposed different criteria to detect communities, i.e., density, connected-ness; however, there is still no unique definition for communities [Bouguessa et al., 2008].

Many community detection algorithms only detect disjoint structures; however, overlapping community detection algorithms identify nodes shared among communities, in which the latter is more realistic in real life. The amount of data generated in networks is large-scale and communities may not only be formed by connections through friendships or co-authorship. While grouping of people with same interests, it might be important to consider the content produced by posts or threads. As such, users are not necessarily connected to each other, e.g., no explicit links exist among users in a forum except communications through threads. Although there are already many algorithms on the topic of overlapping community detection, one may see the problem from a different perspective. In this regard, most of the proposed methods of community detection work based on explicit links among users of the network; however, few methods consider actual content of networks. In other words, network science researchers have proposed structural methods that only consider mixing patterns of nodes; however, these algorithms may not be suitable for all existing contexts, and they may be content-blind. As such, algorithms might have particular dynamics behind its workflow, in which they are ideal for the specific environment. For instance, the map equation method focuses on system behavior, network structures, and local interactions. On the contrary, stochastic and modularity techniques consider network processes and their formations [Rosvall et al., 2009]. Moreover, other algorithms take into account identification of influential members as well as opinion formation in networks [Shahriari et al., 2015b]. One problem with structural-based methods of community detection is that they might not reflect the actual changes happening in the community context, in which we do not know if these structures might be meaningful. In a real situation, people initiate some connections or messages when they have some opinions and ideas to share and communicate. Hence, the communication tendency of people may remain some traces that we can detect important community structures [Baek et al., 2009].

In this regard, we assume that content, debates, and similarities among people help to bring them together, in which community structures can be detected out of the tracked traces. In other words, structural properties might be affected by contextual properties in social networks. As such, considering content might help to detect more realistic clusters. In the following, we mention the research questions that we tackled in this work:

- To what extent structural properties like the number of overlapping nodes, modularity, and average community size are affected by contextual similarities among users in a forum?
- To what extent adding of content improves detection of community structures?
- How much different are structural properties, i.e., modularity and average community sizes, for different overlapping community detection algorithms?

To answer these research questions, we use information retrieval techniques such as Term Frequency and Inverse Document Frequency (TF-IDF) to propose OCD algorithms. We extract and convert posts related to each user. Next, we use an optimization problem with K-means clustering algorithm to detect communities. We identify overlapping nodes by using a threshold value based on node distances to centroids - we name this algorithm, Cost Function Optimization Algorithm (CFOCA). To compare this content-based technique with another method, we also devise a simple algorithm based on merging of communities. We consider each term as a cluster and merge features based on an overlapping threshold, which result in overlapping communities; we call this algorithm Term Community Merging Algorithm (TCMA). Furthermore, we add users' content similarity as new weights. In other words, we combined weights induced by the implicit number of communications among members with content weights. Afterwards, we applied structural methods of community detection on the new graphs such as SLPA [Xie et al., 2011], DMID [Shahriari et al., 2015b], SSK [Stanoev et al., 2011] and CLiZZ [Li et al., 2012]. Results reveal the positive effect of content on structural-based methods; moreover, the CFOCA and TCMA competitively detect communities compared to the structural-based methods. We compared and contrasted these algorithms concerning the number of overlapping nodes, modularity, and average community sizes, in addition to CFOCA similarity costs versus several structural properties are plotted and analyzed. To summarize, we make the following contributions:

- We analyze the problem of overlapping communities on datasets with different contexts, i.e., OSS, learning forums. We as well crawled a new data from interactions and activities in an OSS forum named Jmol - we have made this dataset publicly available [2].

- We devised two simple content-aware overlapping community detection algorithms using information retrieval and optimization techniques. Our experimental results show that these algorithms are competitive in various contexts. We presented the approach and partial results of this work in a workshop paper in conjunction with i-KNOW conference [Shahriari et al., 2016].
- We compared content-based and structure-based algorithms concerning the number of overlapping nodes, average community size and modularity and their correlation with the content similarity of forum members. Results show some reverse relation between content similarity and modularity.

The structure of the paper is as follows. In section 2, we mention the related work. Section 3 describes the proposed content-based OCD algorithms. In section 4, we describe the evaluation protocol and used datasets for our experiments. Section 5 reveals of the experiments by applying the algorithms on a couple of datasets. Finally, in section 6, we conclude the paper.

2 Related Work

In the field of overlapping community detection, there exist a lot of research, in which OCD models have been used widely in social, citation, co-authorship, communication, biological networks, etc. We can categorize the algorithms into local and global methods. Local algorithms only consider local network properties - as proposed in [McAuley and Leskovec, 2012] and [Coscia et al., 2012] - where they used node neighborhoods to detect overlapping communities. Xie et al. [Xie et al., 2011] proposed a so-called speaker-listener method, in which it initializes the nodes as listeners and labels are sent from the neighbors/speakers. Each listener has some listening rules, and according to these rules, only some received labels are stored. Then the nodes are assigned to the communities matching the sorted labels. However, global algorithms take the whole graph into account, and most of the so far published methods belong to this category - examples are [Palla et al., 2005, Akoglu et al., 2012, Ge et al., 2008, Balasubramanyan and Cohen, 2014, Xu et al., 2012]. Furthermore, some algorithms can be denoted as leader-based. They are mostly two-phase algorithms, which identify the leaders (nodes that are strong in some way, for example highly connected) in the first phase and allocate the other nodes to the leaders in the second phase. Examples of the leader-based approaches are [Stanoev et al., 2011, Shahriari et al., 2015a],

[2] https://github.com/rwth-acis/REST-OCD-Services/wiki/Jmol-Dataset

in which they identify the leaders using a random walk transition matrix. Stanoev and Kocarev [Stanoev et al., 2011] use transitive link weights to determine the influence of the nodes on their neighbors and identify the most influential nodes, i.e., the leaders. They assigned the non-leaders by another random walk process.

There are many different approaches for OCD algorithms, such as probability-based or clustering methods. Palla et al. [Palla et al., 2005] proposed an algorithm to detect communities using k-cliques. Their idea is that communities consist of overlapping cliques - k-cliques that have $k-1$ nodes in common. Xu et al. [Xu et al., 2012] proposed a Bayesian probabilistic model, in which they transform the community detection problem to a probabilistic inference problem by defining probabilities for each possible clustering of the vertices of a given attributed graph.

Research works in this area have used clustering with OCD algorithms. Clustering models mostly focus only on the structure of the graph and not on the attributes of the nodes [Ahn et al., 2010]. It is a slightly different approach because it does not cluster the vertices but the links. [Ahn et al., 2010] proposes to determine the similarity of links by comparing the neighborhood of the shared nodes. Each link starts as one community and then a link dendrogram is constructed. To find important communities a cut at the point of highest partition density is used. However, some approaches focus on both the structural similarities and the node attributes. Zhou et al. [Zhou et al., 2009] use k-Medoid clustering with a unified distance measure that is computed by running a random walk model on an attribute augmented graph. The attribute augmented graphs are generated by adding dummy nodes for the attributes and dummy links if the original vertices can be associated with that particular attribute value. Like this, the node attributes and the structure of the network are taken into account.

Yang et al. [Yang et al., 2013] computed probabilities, which an edge connects two nodes and the relevance of the group membership for a node. Then they used these values to determine which clusters the node is belonging. Ruan et al. [Ruan et al., 2013] created links from the content and unifies these new edges with the original ones; however, they retained only the edges with the highest similarity between neighbors, such that a graph with less joint edges will be clustered. But this approach computes disjoint communities as the approach proposed by [Dang and Viennet, 2012]. Their algorithm starts by assigning each node its community and merging the nodes according to a composite modularity gain.

Many approaches take communities as cohesive subgroups [Akoglu et al., 2012, Ge et al., 2008, Moser et al., 2009], in which node attributes are used to compute some feature similarity between the nodes. For instance, [Ge et al., 2008] computes the feature vectors from given intrinsic characteristics and uses relational data as links between the nodes. First, their algorithm computes the distances

between the nodes, and it determines possible center nodes. Then, it assigns all neighbors within a given radius which are connected to the considered center node to the same cluster. This behavior continues until all nodes are assigned to at least one center node.
Our proposed algorithm uses clustering to find the communities based on the content of threads or posts. We fabricated networks out of communications and activities in forums. Most research done so far considered structural aspects; however, there are no links between the users in a forum, except for the questions of other users they have answered. Also, the structural and content-based community detection approaches do not work with existing forum structures; due to lack of node attributes except for a username and the thread content generated by users.

3 Content-Based Overlapping Community Detection Algorithms

In this section, we introduce two OCD algorithms that use and process content of forums to identify overlapping communities. At first, we mention details of each algorithm separately. We use actual content generated by users in question & forums to detect overlapping communities. People might have a higher level of communications while they have some ideas or thoughts to share with each other [Baek et al., 2009], in other words, traces with denser connected components can form. Structural methods of community detection detect communities with the high level of precision; however, exploring content may reveal extra information about mixing patterns and community structures. When we consider question & answer forums, we do not have the explicit network structure among members, and thus we have to use structural methods of community detection on networks fabricated from user activities. In such a case, users who have posted in the same thread of communication will have a connection or who have used the same tags may be considered connected. As such, using of explicit content available in forums might reveal more valuable information regarding community structures. In this regard, one may use properties and attributes of nodes and edges, e.g., tag information; however, here we deal with actual content generated, i.e., sentences and words by users.

As such, a term matrix is computed using the $TF - IDF$ method, and it is clustered based on the optimization of a cost function. In our second simple approach, we consider each term as a community, afterward, communities are merged to achieve better resolutions. To combine structural methods with content-based methods, we merge structural weights with content weights to induce weighted graphs. Then, we can give the resulting weighted network to structural methods of community detection. On the one hand, we can generate structural weights by the number of posts issued between two users. Content-based weights, on the other hand, can be achieved by considering content similarity. In such a case, we can apply structural OCD algorithms, e.g., DMID [Shahriari et al., 2015a] or SLPA [Xie et al., 2011].

3.1 Constructing Term Matrix

To consider the problem in vector and matrix spaces, a vector named *vocabulary* is kept for each user. The vocabulary vector is constructed based on the threads a user has posted and commented. TF-IDF can be used to construct the term matrix and thus rows and columns correspond to users and term frequencies. The TF-IDF is computed as follows:

$$\begin{aligned} tf_t &= freq(t, v_u) \\ idf &= log(\frac{N}{|d \in D : t \in d|}) \\ tf - idf(t, d_u, D) &= tf(t, v_u) \times idf(t, D), \end{aligned} \tag{1}$$

where v_u denotes the vocabulary related to user u, tf is the term frequency, idf shows the inverse document frequency and N indicates the number of users. Moreover, we compute one document (a vocabulary vector) for each user u. In addition, $|d \in D : t \in d|$ and D are respectively the number of all documents containing the term t and the whole set of documents/vocabularies of all the users.

3.2 Cost Function Optimization Algorithm

In this algorithm, we use positions of nodes to detect overlapping communities. Each row of the Term matrix T shows the corresponding node position. The basic idea behind this algorithm is considering nodes with close positions in the same community and nodes with farther positions in different communities. To consider the positioning of nodes, we used K-means clustering algorithm, in which we selected some nodes as community representatives. We update the position of these particular nodes until they reach stable community structures. In this problem, one usually finds the best centroids and the optimal distances by using a cost function J. To optimize the cost in CFOCA, we applied the gradient descent method, in which we updated the centroids as follows:

$$c_j^{t+1} = c_j^t - \alpha \cdot \frac{\partial}{\partial c_j} J(c_1, ..., c_k). \tag{2}$$

With the help of an optimization objective, we find the best number of communities and the optimal distances among the centroids. To find the minimum of the cost function, a search over parameter space identifies the k, which produces the lowest costs. We consider each of the data points as a vector, and thus their distances are calculated based on cosine similarity as follows:

$$\begin{aligned} J &= \frac{1}{n}\sum_{i=1}^{n} 1 - cosSim(u_i, c^{(u_i)}) \\ &= \frac{1}{n}\sum_{i=1}^{n} 1 - \frac{u_i \cdot c^{(u_i)}}{\parallel u_i \parallel \parallel c^{(u_i)} \parallel} \\ &= \frac{1}{n}\sum_{i=1}^{n} 1 - \frac{\sum_{j=1}^{l} u_{i,j} \times c_j^{(u_i)}}{\sqrt{\sum_{j=1}^{l}(u_{i,j})^2} \times \sqrt{\sum_{j=1}^{l}(c_j^{(u_i)})^2}}. \end{aligned} \tag{3}$$

Here node u_i is assigned to $c^{(u_i)}$ and l is the number of words in T.
To compute the $\frac{\partial}{\partial c_j}J$, we proceed as follows:

$$\begin{aligned} &\frac{\partial}{\partial c}\frac{1}{n}\sum_{i=1}^{n} 1 - cosSim(u_i, c^{(u_i)}) \\ =&\frac{\partial}{\partial c}\frac{1}{n}\sum_{i=1}^{n} 1 - \frac{u_i \cdot c^{(u_i)}}{\parallel u_i \parallel \parallel c^{(u_i)} \parallel} \\ =&\frac{1}{n}\sum_{i=1}^{n} \frac{\partial}{\partial c} 1 - \frac{u_i \cdot c^{(u_i)}}{\parallel u_i \parallel \parallel c^{(u_i)} \parallel}. \end{aligned} \tag{4}$$

To compute the above derivation, we require the gradient of $g = u_i \cdot c^{(u_i)}$ and $h = \parallel c^{(u_i)} \parallel$ that can be computed as follows:

$$\begin{aligned} \nabla g &= u_i \\ \nabla h &= \frac{1}{2}\frac{2 \cdot c^{(u_i)}}{\sqrt{\sum_{j=1}^{l}(c_j^{(u_i)})^2}} \\ &= \frac{c^{(u_i)}}{\parallel c^{(u_i)} \parallel}. \end{aligned} \tag{5}$$

By replacing the above values, gradient can be computed as follows:

$$\begin{aligned} &\frac{1}{n}\sum_{i=1}^{n} \frac{\partial}{\partial c} 1 - \frac{u_i \cdot c^{(u_i)}}{\parallel u_i \parallel \parallel c^{(u_i)} \parallel} \\ =&\frac{1}{n}\sum_{i=1}^{n} \frac{\nabla g \cdot \parallel u_i \parallel \parallel c^{(u_i)} \parallel - u_i \cdot c^{(u_i)} \cdot \nabla h \cdot \parallel u_i \parallel}{(\parallel u_i \parallel \parallel c^{(u_i)} \parallel)^2} \\ =&\frac{1}{n}\sum_{i=1}^{n} \frac{u_i \cdot \parallel u_i \parallel \parallel c^{(u_i)} \parallel - u_i \cdot \frac{(c^{(u_i)})^2}{\parallel c^{(u_i)} \parallel} \cdot \parallel u_i \parallel}{(\parallel u_i \parallel \parallel c^{(u_i)} \parallel)^2}. \end{aligned} \tag{6}$$

The centroid c_j will be updated for step $t+1$ as follows.

$$c_j^{t+1} = c_j^t - \alpha \cdot \frac{\partial}{\partial c_j} J(c_1, ..., c_k). \tag{7}$$

As the number of communities (k) is unknown, we run the algorithm with different k values. We show the pseudo code for CFOCA in Algorithm 1.

Algorithm 1 CFOCA based on k-means clustering

```
centroids ← centroid initialization
tempCent ← centroids
clustering ← membership matrix
i ← 0
j ← 0
while centroids ≠ tempCent do
    for all nodes do
        for all centroids do
            distCent1 ← dist(node.pos(i), centroid.pos(j))
            distCent2 ← dist(node.pos(i), centroid.pos(j+1))
            if distCent1 >= distCent2 then
                clustering.pos(i) ← centroid.pos(j+1)
            else clustering.pos(i) ← centroid.pos(j)
            end if
            j ← j + 1
        end for
        i ← i + 1
    end for
    tempCent ← centroids
    centroids ← updateCentroids(centroids)
end whilereturn clustering
```

Up to this step, CFOCA can only detect disjoint communities. To obtain realistic communities, we assign nodes to overlapping communities by using a suitable threshold value ε. Hence, we considered a node as overlapping if its distance to centroids is less than ε.

3.3 Term Community Merging Algorithm

To consider the term vectors and content of forums with a much more straightforward approach, we proposed an algorithm called Term Community Merging Algorithm (TCMA) that works based on the simple merging of overlapping communities. First, we fabricate a term matrix out of the existing terms of the vocabulary - denoted as T. Each column of T shows a term, and the users will be incident to these word columns if they have used the term in her posts. In the continuation, we merge the clusters by using the following overlapping coefficient:

$$overlapping\ coefficient = \frac{|C_i \cap C_j|}{min\{|C_i|, |C_j|\}}, \tag{8}$$

in which C_i and C_j are intermediate clusters identified in each step. Similarly, it requires selecting a suitable threshold β for the comparison of overlapping coefficient. We explained TCMA with a pseudo-code shown in Algorithm 2.

Algorithm 2 Term Community Merging Algorithm

```
1: beta ← Overlapping Coefficient
2: features ← Term Matrix computed using tf-idf
3: clustering ← resulting clustering matrix
4: for each column in features do
5:     if features.position(currColumn,i) ≠ 0 then
6:         addToCluster(clustering.position(currColumn), i)
7:     end if
8: end for
9: refine(clustering)
       return clustering
```

3.4 Structural-Based Techniques Used for Comparison

- **Disassortative Degree Mixing and Information Diffusion (DMID)** This algorithm is a two-phase algorithm, in which it works based on two dynamics named disassortative degree mixing and information diffusion [Shahriari et al., 2015a]. Disassortative degree mixing shows dissimilarity patterns among neighbors of a node. In the first phase of the algorithm, we combined the degree of a node with its degree to find influential nodes in the network. First, we defined an assortative matrix as follows:

$$AS_{ij} = \begin{cases} ||deg(i)| - |deg(j)|| & \text{, if } j \in deg(i) \\ 0 & \text{, otherwise} \end{cases}, \tag{9}$$

where $|deg(i)|$ shows the degree of node i and $||deg(i)| - |deg(j)||$ is the absolute value. Now, this matrix contains the degree difference corresponding to nodes. If we apply a random walker on the corresponding row-normalized transition matrix, we can identify disassortative paths. In other words, the walker tends to flow in directions with higher degree differences. Afterwards, a Disassortative Vector (DV) is considered to hold the disassortative value of each node. We initialize DV with $\frac{1}{|N|}$, and we update with the help of disassortative transition matrix TAS; therefore the update is based on:

$$DV^t = DV^{t-1} \times TAS, \tag{10}$$

after enough iterations, the process converges, and we obtain the disassortative value of each node. To be considered as a leader, we need to combine the simple degree of nodes need in addition to homophilie. We can calculate the leadership value of node i ($LV(i)$) as follows:

$$LV(i) = DV(i) \times |deg(i)|. \tag{11}$$

So far, each node i can be represented by its relative leadership value $LV(i)$. To further proceed toward final leaders, vertices need to decide regarding local leadership locally. In other words, for each node i, a local leader needs to be selected as follows:

$$LV(i) > LV(j) \quad \forall j \in deg(i). \tag{12}$$

As the formula represents, node j is the follower of node i, therefore a forest forms. Finally, DMID calculated the global leaders by considering a threshold named average follower degree. In the second phase of DMID, it uses a cascading process titled Network Coordination Game to identify the membership value of nodes to communities, i.e., leaders. Network coordination game considers two strategies of A and B, in which at some point in time one of the strategies appear in a network of unique strategies, and thus the new behavior is cascaded based on the density of communities.

– **Speaker-listener Label Propagation Algorithm (SLPA)** SLPA algorithm simulates human behavior by playing roles of speaker and listener, in which each agent has a memory to store information on received labels [Xie et al., 2011]. At first, we initialize each node with an id, afterward, it performs the actual label propagation. SLPA iterates over the nodes until convergence. This algorithm selects one node as a listener, and one of its neighbors sends messages based on a specific speaking rule. Finally, the listener accepts one of the labels from all the received labels based on a particular listening rule.

– **Algorithm by Stanoev, Smikov and Kocarev (SSK)**

  Stanoev et al. proposed a two-phase algorithm based on the influence dynamics and the membership computation [Stanoev et al., 2011]. In the first phase, they employed a random walk to calculate the local and global influence matrices. We nominate this algorithm as SSK, in which it assumes that relationships among nodes and their influences are counted more important than considering the direct connection. In other words, proxies among nodes are better established while there exist triangles among nodes. It constructs the influence matrix via adjacency matrix and triangle occurrences (3-cliques) among nodes. This matrix is further applied to achieve the local and global influential nodes, i.e., the hierarchy of the network. In other words, SSK calculates the transitive link matrix as follows:

$$tl_{ji} = tl_{ji} + \sum_{k} tlv_{ji}^{k}, \tag{13}$$

where $tlv_{ji}^{k} = minA_{ki}, A_{jk}$ is the transitive link weight for the edge (i, j) which goes through k. The corresponding transition matrix for doing the random walk can be obtained by row normalizing the tl matrix. After doing the random walk, the most influencing neighbours of node i is identified based on $(N_{influential} = j|T_{ji} = max_k T_{ki})$ that T shows the computed link weight transition matrix. By comparing the influencing neighbors with their neighbors' influences, we can detect leaders. Afterwards, we can identify the membership of nodes to set of leaders by considering weighted average membership of neighbors. The updating rule for membership computation is as follows:

$$M_i(t+1) = \sum_{j=1}^{n} A_{ij} M_j(t), \tag{14}$$

where A_{ij}, is the row-normalized adjacency matrix. A row normalized adjacency matrix is computed based on normalizing each row of A by the sum of the row. The hierarchical and decentralized working behaviors are among properties that SSK possesses.

– **Algorithm By Li, Zhang, Liu, Chen and Zhang (CLiZZ)**

  This algorithm comprises two main steps. One includes identifying leader nodes, and the other contains computing the membership of nodes to communities [Li et al., 2012]. We nominate this algorithm as CLiZZ. To identify leader nodes, it computes the influence range of members based on shortest distance. It determines the mutual effects of nodes towards each other based on the following formula:

$$LV_i = \sum_{j=1\ ;\ d_{ij} \leq = \lfloor \frac{3\delta}{\sqrt{2}} \rfloor}^{n} e^{-\frac{d_{ij}}{\delta}}, \tag{15}$$

$$M_i(t+1) = \frac{1}{1 + \sum_{j=1}^{n} A_{ij}} \left[M_i(t) + \sum_{j=1}^{n} A_{ij} M_j(t) \right]. \tag{16}$$

  The algorithm needs to determine the δ based on the topological entropy of nodes. CLiZZ is suitable for directed and weighted networks.

4 Datasets and Evaluation

To evaluate the proposed OCD algorithms as well as structural methods of community detection, we used several evaluation metrics. Average community size, number of overlapping nodes, modularity and similarity costs are among the metrics.

4.1 Combined Modularity

Regarding modularity, we used a combined version that both use content and structure of the constructed graphs - it is named combined modularity [Dang and Viennet, 2012]. Combined modularity applies Newman modularity with a similarity measure, i.e., cosine similarity, which we can write as follows:

$$\begin{aligned} Q_{comb} = \alpha * \frac{1}{2m} \sum_{ij} [A_{ij} - \frac{deg(i)deg(j)}{2m}] \delta(i,j) \\ + (1-\alpha) * \sum_{ij} cosSim(v_i, v_j) \delta(i,j), \end{aligned} \tag{17}$$

where m is the number of edges, A is the adjacency matrix and $\delta(i,j)$ is equal to 1 if node i and j share one community in common, otherwise 0. Moreover, v_i shows the vector corresponding to row i of the matrix T and α as a threshold determines the importance of the structure and the content. On the one hand, if $\alpha = 1$, this measure behaves like the normal Newman modularity. On the other hand, $\alpha = 0.5$ applies equal importance for both content and structure.

4.2 Datasets

We use four datasets for our experiments. Two of the datasets include forum discussion about open source software named BioJava and BioJmol. The other two datasets are Academic Exchange and Urch, in which both relate to the forum discussion about learning issues. We describe these datasets in the following:

- **BioJava**

  The BioJava project is dedicated to creating Java tools for processing biological data. The developer of this project published the first application in 2008, but the data was crawled from an earlier starting point, the year 2000. The exact period that was crawled begins in 2000 and lasts until 2013. Additionally, some information about the releases of BioJava was crawled; therefore we were able to perform an evaluation based on these release periods.

- **BioJmol**

  The Jmol[3] project resulted in an open-source Java tool for molecular modeling of chemical structures in 3D. The project itself began 1999, and the administrators of the project crawled data over a period of eleven years (2002 - 2012). It is a quite new dataset, and we analyzed this dataset for the first time.

- **Academic Exchange**

  Stack Exchange[4] is a network of Q & A communities, that covers several topics like mathematics, home improvement, statistics and English Language and usage. The most popular community might be *Stack Overflow*, a site for questions about software development. We investigated one of this Q & A communities, which is called *Academia*.

  The *Academia Stack Exchange* Q & A community mostly serves academics and those enrolled in higher education. Among others, it handles questions about publications and their formalities, research questions, and problems concerning the Ph.D., master or theses in general. It is possible to download the data dump[5], in which it includes the posts during the years 2011 and 2015; however, we picked the recent year 2015.

- **Urch**

  The URCH learning forum provides for language learners the possibility to discuss their issues. Most of the users prepare for an English test like the TOEFL, GMAT, and GRE, and thus they can address exam questions or get feedback on some written essays shared before [Petrushyna et al., 2011]. We chose the year 2004 for this dataset and did experiments based on different graphs constructed for each month.

5 Results

In this section, we demonstrate our experiments on the four datasets including academic exchange, BioJmol, BioJava, and Urch. We divide our experiments into two subsections: first, for all datasets, we compare different algorithms based on the number of overlapping nodes, average community size, and modularity. Afterwards, we explain the results regarding similarities among content and community properties on two datasets of BioJmol and URCH.
We implemented CFOCA and TCMA and compared them with several OCD approaches. We applied CFOCA, TCMA and other structural and contextual

[3] www.jmol.org
[4] stackexchange.com
[5] https://archive.org/download/stackexchange

algorithms including CDMID, DMID, CLiZZ, CCLiZZ, SSK, CSSK and SLPA on the mentioned datasets. The "C" before structural methods like DMID, CLiZZ, SSK show that we applied the structural methods on a weighted graph, in which we add content similarity to a graph constructed out of thread communications.

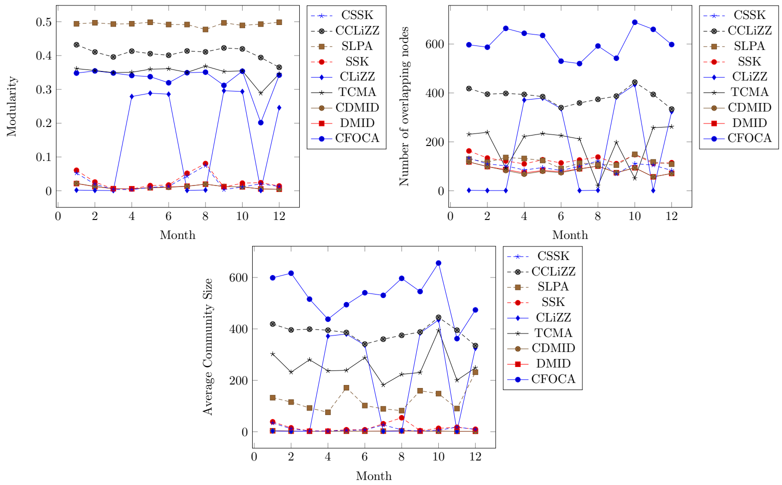

Figure 1: Modularity, number of overlapping nodes and average community size for different structural and content-based OCD algorithms on Academic Exchange dataset.

5.1 Analysing Community Properties

Academic Exchange. In Figure 1, we can observe modularity, some overlapping nodes and average community sizes for different algorithms on Academic Exchange dataset. SLPA algorithm obtained the highest modularity value. It is followed by CCLiZZ with stable modularity values over time. CCLiZZ mainly obtains modularity values of around 0.4 for the whole period of time. Next, TCMA obtains the best modularity values which have some equal values with CFOCA. CFOCA compared to TCMA faces some fluctuations with the minimum modularity value of 0.2 in November. In February, March, July, October, and December, they obtain equal modularity values. CLiZZ obtained the next position; however, the results for it is quite unstable ranging from minimal modularity values near to zero up to large modularity values of around 0.3. In other words, CLiZZ achieves the highest values in periods from April to Jun and September to October. Different algorithms reach more or less small modularity results of less than 0.1. Modularity results for SSK and CSSK overlap with each other and lie on top of DMID and CDMID. DMID and CDMID overlap with each other and reach quite similar modularity values. In fact, content does not affect modularity values of SSK and DMID algorithms in academic exchange dataset; however, it could improve the CLiZZ algorithm. Overall, CFOCA and TCMA reach satisfactory modularity values in comparison to other algorithms.

Concerning the number of overlapping nodes, CFOCA has the highest number of overlapping nodes, in which indicates that this algorithm assigns many of the nodes to more than one community. CCLiZZ and CLiZZ follow it; however, in several months CLiZZ only detects very few nodes as overlapping. One may see TCMA as an algorithm that identifies a higher number of overlapping nodes; however, for several months, the number of overlapping nodes slump such as March, August, and October. Other algorithms such as DMID, CDMID, SSK, CSSK, and SLPA achieve more stable and lower number of overlapping nodes. For instance, as for July, CFOCA, DMID, CDMID, TCMA, CLiZZ, SSK, SLPA, CCLiZZ and CSSK achieve 520, 90, 89, 212, 1, 126, 113, 359 and 100 number of overlapping nodes, respectively.

Concerning average community sizes, the pattern is similar to the number of overlapping nodes - CFOCA, CCLiZZ, CLiZZ, and TCMA obtain the highest average community sizes, respectively. CLiZZ faces unstable and erratic average community sizes of maximum 435 and minimum of 2. Although the number of nodes in different time steps is not equal, content-based algorithms tend to detect bigger communities with higher overlaps with CFOCA on top. The lowest average community sizes belong to SLPA, DMID, CDMID, CSSK, SSK. For instance, as for July, CFOCA, DMID, CDMID, TCMA, CLiZZ, SSK, SLPA, CCLiZZ and CSSK achieve 530.5, 2.5, 2.4, 182, 2, 31.2, 89, 360 and 26.5, respectively.

BioJava. We also evaluated the OCD algorithms on BioJava dataset based on a release-based period. As one may observe in Figure 2, there are nine releases considered in the evaluations. CLiZZ achieves minimal modularity values of near to zero. CCLiZZ also obtains similar results; however, its modularity magnitudes are higher as for the first five releases. Adding to content could increase obtained modularity at least for five of the releases. CSSK and SSK have similar trends to CCLiZZ with higher modularity values as for the first five releases; however, it damped to minimum values - release six onward. TCMA, SLPA, DMID, and CDMID achieve the highest modularity values over the releases; however, CFOCA falls behind with lower values. Among them one may not specify a most top winning algorithm because they have overlaps in many cases. Similar to Academic exchange evaluation, one may observe that content had a positive effect on modularity values on the CLiZZ algorithm. In general, content-based algorithms also achieve satisfactory modularity values on BioJava dataset.

Furthermore, we can also observe comparison of the number of overlapping nodes over the releases on the BioJava dataset in Figure 2. As we can see,

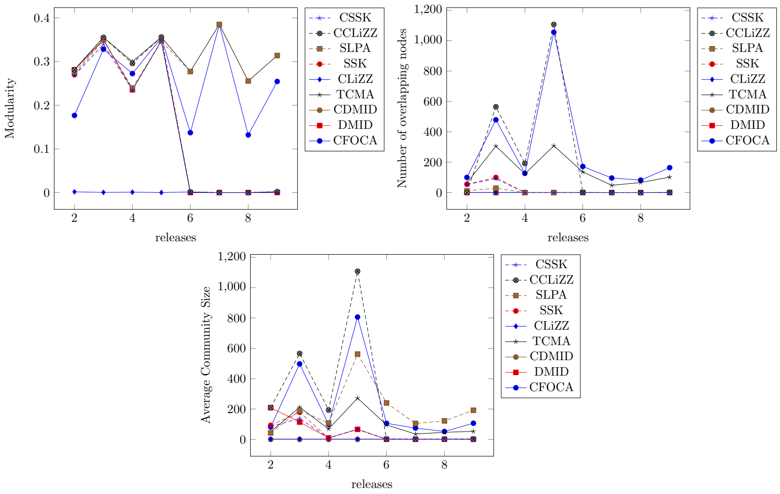

Figure 2: Modularity, number of overlapping nodes and average community size for different structural and content-based OCD algorithms on BioJava dataset.

CCLiZZ and CFOCA obtain the highest number of overlapping nodes in comparison to others. In other words, they identify most of the network nodes as overlapping among different communities. However, there are releases that the number of overlapping nodes are zero for these algorithms. Then, TCMA follows the number of overlapping nodes with more stable results with the maximum of 308 nodes as overlapping. SSK has 54 and 100 for the first two releases and afterward damps to zero. Similarly, CSSK achieves 53 and 92 and suddenly damps to zero number of overlapping nodes. DMID and CDMID did not detect any nodes as overlapping and it may show that the detection of community structures might be ambiguous for some of the algorithms. Finally, SLPA had a similar situation to SSK with 11 and 29 as the number of overlapping nodes followed by zero for the next releases. Regarding average community size, CFOCA and CCliZZ obtained the highest values. This can be because of the high number of overlapping nodes detected for these two algorithms. SLPA has the next position for average community size; however, it is slightly unstable with the peak at release five similar to CFOCA and CCliZZ. TCMA obtains relatively high average community sizes with a peak value of 271. DMID and CDMID overlap entirely on each other and have a maximum of 209 as the average community size. SSK and CSS more or less obtain similar average community sizes, in which from release six on assign each node to a separate community which is not realistic. This observation also happens for DMID and CDMID at releases 7 and 8. In general, average community sizes depend on the internal structure of the network; however, content-based techniques tend to detect bigger communities with more considerable overlaps also on forums corresponding to open source software development.

BioJmol. In Figure 3, we can observe the average community sizes by applied OCD algorithms versus release times for BioJmol dataset. Almost structural-based algorithms were unable to detect overlapping communities due to lack of thread communications among members during the first five releases. This behavior can be justified as the developers were working structurally more isolated than cooperative, and thus leads into separated content generations. Usually, in the beginning, people struggle to understand the elements of the project and get familiarized with it, in which may cause a cold start situation. However, their generated content could reveal several communities with the applied CFOCA and TCMA. This finding indicates that when explicit structural information of the network is missing, content-based algorithms dealing with the actual context of the social network may be more informative. In other words, when cold start problem exists, content-based methods are still able to detect overlapping communities.

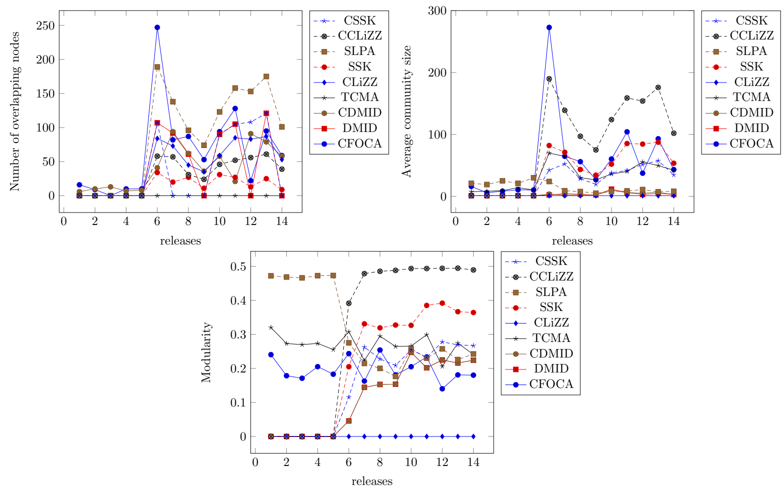

Figure 3: The number of overlapping nodes, modularity, average community size and execution time for structural and content-based overlapping community detection algorithms on the BioJmol dataset.

Furthermore, DMID and CDMID detect low average community sizes. One may also spot fewer fluctuations among detected communities while applying structural methods such as SLPA, DMID, SSK, and CLiZZ. This can be explained as structural methods only track the communication threads; however, forums are more context-dependent and conversations' topic and content change over the period of releases. Content-based algorithms such as CFOCA and TCMA further reflect this issue. We can observe other findings, for instance, CLiZZ is unable to detect community structures when there is no content, on the contrary, content-based weighting version can identify communities. Additionally, in release six, the highest average community sizes belong to CFOCA (273), CCLiZZ (190), SSK (82.1), TCMA (69.8) and CSSK (42.33). Other algorithms like DMID (2.28), CDMID (3.53), SLPA (24) detect smaller average community sizes. This observation indicates the apparent difference in community resolutions for different algorithms.

Figure 3 also indicates the number of overlapping nodes generated by different algorithms for the BioJmol dataset. Here, structural methods were unable to detect overlapping nodes when there is lack of edges for the first five releases. As for DMID, adding of content negligibly affects on the number of overlapping nodes. For instance, in periods 10 and 11 both DMID and CDMID detected respectively 105 overlapping members. Concerning CFOCA, the number of overlapping nodes are also higher than other algorithms. For instance, CFOCA detected 247, 82, 87 and 53 overlapping nodes for periods 6-9; however, TCMA as a content-based method discovered 41, 94, 62 and 36 overlapping nodes. Although DMID detects low average community sizes, the number of overlapping nodes are as high as CFOCA and TCMA. Another issue is about adding of content to structural-based techniques. Adding to content increases the number of overlapping nodes in almost all of the structural-based approaches except SSK. This finding can be justified while content and context reveal broader boundary overlaps among the covers. As for SLPA, the number of overlapping nodes are much lower than other algorithms. Additionally, the number of overlapping nodes are similar to average community sizes. This effect may be due to the availability of content, which causes identification of most of the nodes as overlapping. In other words, members may participate in different contexts and disciplines of the project.

Regarding modularity in BioJmol, for the first five releases, structural methods generate almost zero modularity due to lack of communication threads in the network. In five snapshots, SLPA makes approximately high values for modularity, but it does not have the proper resolution while each node is assigned to a single community which is not realistic. TCMA obtains higher modularity values in comparison to CFOCA; however, results for SLPA indicates that modularity is not the only dominant factor. Because considering nodes in single communities even generate high modularity values; however, they are not essential communities. We can also observe that content-based algorithms such as CCLiZZ, TCMA, and CFOCA also reach satisfactory values for modularity. However, we recognize that modularity is not the only principal factor, in which we need to consider other factors such as meaningfulness of the communications.

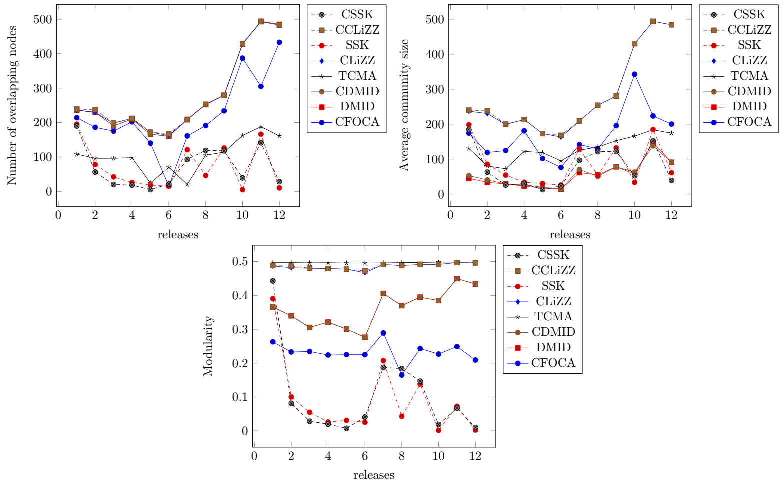

Figure 4: Modularity, average community size and the number of overlapping nodes generated based on structural and content-based overlapping community detection algorithms on URCH dataset.

URCH. Figure 4 indicates average community size over the months of the year 2004. The numbers on the x axis correspond to months of the year 2004. As we can observe the highest community sizes belong to CCLiZZ over all the months, which is followed by TCMA and CFOCA. Other algorithms including CLiZZ, CSSK, SSK, DMID, and CDMID have lower average community size of approximately below 100. This observation indicates that content-based methods detect communities of larger sizes. One may observe fluctuations among all of the algorithms in various months, which may show different levels of activities in different months in URCH. This finding may be because of either exam times, which it distributes over the period of a year or the university application deadlines that are usually two times per year. Besides, we plot the number of overlapping nodes versus months in the year 2004 of URCH dataset. The number of overlapping nodes indicates the connections among communities. Overlapping members contribute to the diffusion of information and innovations across communities. Here, DMID and CDMID identify approximately the highest number of overlapping nodes. Adding of content to DMID did not increase the number of overlapping nodes. Similar to the community sizes, CFOCA and TCMA result in higher number of overlapping nodes in comparison to other algorithms. This observation indicates that content-based approaches detect higher levels of communications. Other algorithms experience lower values for the number of overlapping nodes over the period of months. Regarding modularity, TCMA, CLiZZ, CCLiZZ have the highest values, which are followed by DMID and CD-

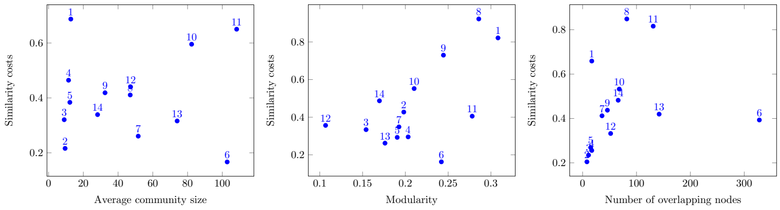

Figure 5: This figure shows similarity costs versus the number of overlapping nodes, average community size, and modularity for Bio-Jmol dataset.

MID. CFOCA has an average modularity between 0.2 and 0.3, which is ranked after DMID. Finally, SSK and CSSK obtain the lowest modularity values on URCH dataset.

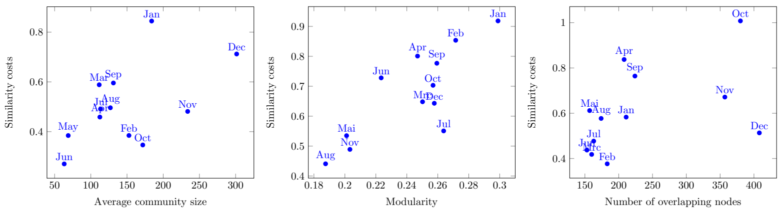

Figure 6: This figure shows similarity costs versus the number of overlapping nodes, average community size, and modularity for URCH dataset.

5.2 Experiments on Content and Community Properties

To investigate the relation between the content generated in forum communications and the community structure, we plotted the similarity costs versus the average community size, modularity and number of overlapping nodes. Higher similarity costs indicate fewer similarities among people, on the contrary, lower costs are signs of higher similarity. Figure 5 also shows the content similarity versus average community size in Bio-Jmol dataset. We can observe that periods 1, 10 and 11 have the highest similarity costs (lowest content similitude). Release 1 may indicate small communities of people at the beginning of the project. In contrast, releases 10 and 11 show bigger communities and still lower similarities, which might indicate lower activities at the end of the project. Although we observe small communities at releases 2, 3, 4 and 5, content similarities enhance for them. We can observe the highest content similarity for release 6, in which increment of members from 30 to 352 members, can be a reason for this. Furthermore, we can observe that average community sizes between 10 to 50 have the highest similarity of content. This behavior has an exception with release six with more than 100 average community size property. Regarding similarities of content and overlapping nodes, we can perceive that highest content similarities can be observed for releases 2, 3, 4 and 5, and this may be due to few number of members at the beginning of the project. When there is few number of members (around 20 members), overlapping nodes are limited to 2 or 3 members which seems realistic. There is an increase in the number of nodes at release 6, and correspondingly in the number of overlapping nodes, that may indicate communities have high mixing tendency. As for releases 7, 9, 14 and 10, there are somehow high content similarity and around 36 to 68 overlapping members. At these release periods, the number of members is about 100, which indicates half of the nodes are overlapping among communities.

When there is high content similarity among members, they have some ideas to share. Moreover, overlapping nodes can further propagate information over the network. Furthermore, we observe a reverse relation between similarity and modularity values of the detected communities by CFOCA algorithm. For most of the releases, for instance, 3, 5, 13, 7, 2, etc., the similarity costs are low, which indicate high similarity of content among members that lead to little modularity in detected communities. Other releases including 1, 8 and 9 have high similarity costs and high modularity structures that indicate lower similarity contents - the identified communities tend to be more modular.

Figure 6 shows similarity costs versus the number of overlapping nodes, average community size, and modularity for URCH dataset. It shows modularity values versus months in the URCH dataset during the year 2004. We can observe that the highest modularity values for identified communities belong to TCMA and CCLiZZ. They steadily reach higher values, which indicate that these algorithms perform better on URCH dataset regarding modularity. It is followed by CDMID and DMID, which show that content could not improve DMID as for modularity. CFOCA achieves lower values of modularity in comparison to the mentioned algorithms, but still satisfactorily performant. SSK, CSSK, and SLPA obtained worse modularity values for the learner networks over the period of months. Although modularity is a comparative metric, putting all the nodes in one community generates high values of modularity that is not realistic in social networks. Thus, we require innovating better methods for the evaluation of OCD algorithms.

Moreover, we can observe the relation between the similarity costs and modularity based on CFOCA algorithm. In CFOCA, higher similar costs indicate lower content similarities and thus higher content similarity is related to January, in which the network structure is modular. As we can observe during first three months of 2004 - January, February, and March - the learner network has low content similarity among users. At the end of the year, it also faces a low similarity content, i.e., September, October, and December. During this period the network has quite high values for modularities. In other words, high values of modularity show to some extent reverse relation with the content similarity of users. The additional months including April, June, and July have a similar situation. On the other hand, November, May, and August have high content and lower modular network. We attributed the structure of the networks to the events that happen in learning networks, and we require to investigate such correlations in future - events such as university application deadlines or test timelines. Additionally, we can observe the similarity costs versus the number of overlapping nodes for URCH. As one may notice the communication among learner networks at the end of the year 2004 has increased to a certain extent, which indicates the high amount of information diffusion among communities. Moreover, one may notice the high similarity of content in December, January, February, March, June, and July that their similarity cost values are lower than 0.6. Moreover, we can observe the lower similarity of content for other months. We can attribute the increased number of overlapping nodes and content similarity to the events happening in the URCH dataset in which we require to identify these events.

6 Conclusion

In this paper, we evaluated OCD algorithms on two forums corresponding to open source software development named BioJmol and BioJava, and two forums related to learning topics, i.e., called URCH and Academic Exchange. We considered temporal versions of these datasets for the analysis of results. We investigated the number of overlapping nodes, average community sizes, and modularity to extract community properties. Furthermore, we sketched the number of overlapping nodes, average community sizes, and modularity versus similarities. Our experiments suffer from several shortcomings that we need to address them in future works. The cost function generates a global similarity cost value that may not reflect the actual similarities among members. To gain more realistic and fine-grained dynamics, we need to investigate and define some local similarity values. Besides, we observed that structural methods achieved more stable community results in comparison to the content-based approaches, but we need to investigate this on more datasets. Content-based approaches may show the more realistic appearance of communities. Thus, we intend to study the stabilities of overlapping communities on further datasets and based on other measures. In some cases, assigning each node to single communities yielded the highest values for modularity. Although it does not dissatisfy the definition of overlapping communities, it is unfair to get the highest values of modularity.

Even applying of a combined version of modularity using of both content and structure did not resolve the issue and it is required to investigate a better way to evaluate the goodness of the algorithms. Furthermore for some algorithms, adding of content were beneficial for the structural-based techniques and resulted in better performance of the algorithm. In this regard, we would like to investigate which structural-based OCD techniques may be improved by adding content.

References

[Ahn et al., 2010] Ahn, Y., Bagrow, J. P., and Lehmann, S. (2010). Link Communities Reveal Multiscale Complexity in Networks. Nature, 466(7307):761–764.

[Akoglu et al., 2012] Akoglu, L., Tong, H., Meeder, B., and Faloutsos, C. (2012). PICS: Parameter-free Identification of Cohesive Subgroups in Large Attributed Graphs Mining, Anaheim, California, USA, April 26-28. pages 439–450.

[Baek et al., 2009] Baek, S. C., Kang, S., Noh, H., and Kim, S. W. (2009). Contents-Based Analysis of Community Formation and Evolution in Blogspace. In *2009 IEEE 25th International Conference on Data Engineering*, pages 1607–1610.

[Balasubramanyan and Cohen, 2014] Balasubramanyan, R. and Cohen, W. W. (2014). Block-LDA: Jointly Modeling Entity-Annotated Text and Entity-Entity Links. In *Handbook of Mixed Membership Models and Their Applications.*, pages 255–273.

[Bouguessa et al., 2008] Bouguessa, M., Dumoulin, B., and Wang, S. (2008). Identifying Authoritative Actors in Question-Answering Forums: The Case of Yahoo! Answers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 866–874, New York, NY, USA. ACM.

[Cazabet et al., 2010] Cazabet, R., Amblard, F., and Hanachi, C. (2010). Detection of Overlapping Communities in Dynamical Social Networks. In *Proceedings of the 2010 IEEE Second International Conference on Social Computing*, SOCIALCOM '10, pages 309–314, Washington, DC, USA. IEEE Computer Society.

[Coscia et al., 2012] Coscia, M., Rossetti, G., Giannotti, F., and Pedreschi, D. (2012). DEMON: a Local-First Discovery Method for Overlapping Communities. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 615–623, New York, NY, USA. ACM.

[Dang and Viennet, 2012] Dang, T. A. and Viennet, E. (2012). Community Detection in Social Networks with Attribute and Relationship Data. In *Extraction et gestion des connaissances (EGC'2012), Actes, janvier 31 - février 2012, Bordeaux, France*, pages 563–564.

[Ge et al., 2008] Ge, R., Ester, M., Gao, B. J., Hu, Z., Bhattacharya, B., and Ben-Moshe, B. (2008). Joint Cluster Analysis of Attribute Data and Relationship Data: The Connected K-center Problem, Algorithms and Applications. *ACM Transactions on Knowledge Discovery from Data*, 2(2):7:1–7:35.

[Greene et al., 2010] Greene, D., Doyle, D., and Cunningham, P. (2010). Tracking the Evolution of Communities in Dynamic Social Networks. *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 176–183.

[Leskovec et al., 2007] Leskovec, J., Kleinberg, J. M., and Faloutsos, C. (2007). Graph Evolution: Densification and Shrinking Diameters. *ACM Transactions on Knowledge Discovery from Data*, 1(1).

[Li et al., 2012] Li, H.-J., Zhang, J., Liu, Z.-P., Chen, L., and Zhang, X.-S. (2012). Identifying Overlapping Communities in Social Networks Using Multi-scale Local Information Expansion. *The European Physical Journal B*, 85(6).

[McAuley and Leskovec, 2012] McAuley, J. and Leskovec, J. (2012). Learning to Discover Social Circles in Ego Networks. In *Advances in Neural Information Processing Systems 25*, pages 548–556.

[Milo et al., 2002] Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., and Alon, U. (2002). Network Motifs: Simple Building Blocks of Complex Networks. *Science*, 298(5594):824–827.

[Moser et al., 2009] Moser, F., Colak, R., Rafiey, A., and Ester, M. (2009). Mining Cohesive Patterns from Graphs with Feature Vectors SDM 2009, April 30 - May 2, 2009, Sparks, Nevada, USA. pages 593–604.

[Palla et al., 2005] Palla, G., Derényi, I., Farkas, I., and Vicsek, T. (2005). Uncovering the Overlapping Community Structure of Complex Networks in Nature and Society. *Nature*, 435(7043):814–818.

[Palla et al., 2009] Palla, G., Pollner, P., Barabási, A.-L., and Vicsek, T. (2009). Social Group Dynamics in Networks. In *Adaptive networks: NECSI. Thilo Gross ; Hiroki Sayama ed*, New England Complex Systems Institute book series, pages 11–38. Springer, Berlin and Heidelberg.

[Petrushyna et al., 2011] Petrushyna, Z., Kravcik, M., and Klamma, R. (2011). Learning Analytics for Communities of Lifelong Learners: A Forum Case. In *Advanced Learning Technologies (ICALT), 2011 11th IEEE International Conference on*, pages 609–610.

[Rosvall et al., 2009] Rosvall, M., Axelsson, D., and Bergstrom, C. T. (2009). The Map Equation. *Phys. J. Spec. Top*, 178:13–23.

[Ruan et al., 2013] Ruan, Y., Fuhry, D., and Parthasarathy, S. (2013). Efficient Community Detection in Large Networks Using Content and Links. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1089–1098, New York, NY, USA. ACM.

[Shahriari et al., 2016] Shahriari, M., Haefele, S., and Klamma, R. (2016). Contextualized versus Structural Overlapping Communities in Social Media. In *In Proceedings of the First Workshop on Recommender Systems and Big Data Analytics co-located with I-KNOW'2016. Online under: http://socialcomputing.know-center.tugraz.at/rs-bda/papers/RS-BDA16_paper_4.pdf.*

[Shahriari et al., 2015a] Shahriari, M., Krott, S., and Klamma, R. (2015a). Disassortative Degree Mixing and Information Diffusion for Overlapping Community Detection in Social Networks (DMID). In *Proceedings of the 24th International Conference on World Wide Web Companion, WWW 2015, Florence, Italy, May 18-22, 2015 - Companion Volume*, pages 1369–1374.

[Shahriari et al., 2015b] Shahriari, M., Krott, S., and Klamma, R. (2015b). WebOCD: A RESTful Web-based Overlapping Community Detection Framework. In *Proceedings of the 15th International Conference on Knowledge Technologies and Data-driven Business, I-KNOW '15, Graz, Austria, October 21-23, 2015*, pages 51:1–51:4.

[Stanoev et al., 2011] Stanoev, A., Smilkov, D., and Kocarev, L. (2011). Identifying Communities by Influence Dynamics in Social Networks. *Physical Review*, 84(4).

[Xie et al., 2011] Xie, J., Szymanski, B. K., and Liu, X. (2011). SLPA: Uncovering Overlapping Communities in Social Networks via a Speaker-Listener Interaction Dynamic Process. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, pages 344–349.

[Xu et al., 2012] Xu, Z., Ke, Y., Wang, Y., Cheng, H., and Cheng, J. (2012). A Model-Based Approach to Attributed Graph Clustering of Data, SIGMOD 2012, Scottsdale, AZ, USA, May 20-24. pages 505–516.

[Yang et al., 2013] Yang, J., McAuley, J., and Leskovec, J. (2013). Community Detection in Networks with Node Attributes. In *2013 IEEE 13th International Conference on Data Mining*, pages 1151–1156.

[Zhou et al., 2009] Zhou, Y., Cheng, H., and Yu, J. X. (2009). Graph Clustering Based on Structural/Attribute Similarities. *PVLDB*, 2(1):718–729.