Privacy-Preserving User Clustering: The Application of Anonymized Data to Community Detection in Large Organizations

1st Igor Jakovljevic
ISDS
Graz University of Technology
Graz, Austria
e-mail: igor.jakovljevic@cern.ch

2nd Martin Pobaschnig
ISDS
Graz University of Technology
Graz, Austria
e-mail: martin.pobaschnig@student.tugraz.at

3rd Christian Gütl
ISDS
Graz University of Technology
Geneva, Switzerland
e-mail: c.guetl@tugraz.at

4th Andreas Wagner
IT Department
CERN
Graz, Austria
e-mail: andreas.wagner@cern.ch

***Abstract*—This paper is an extension of our previous work on privacy-protected user clusters identification in large organizations. Oversharing exposes risks, such as improved targeted advertising and leakage of sensitive information. Requiring only the bare minimum of data reduces these risk factors, while simultaneously increasing the privacy of each user. Using anonymized data to find communities opens up new possibilities for large organizations under strong data protection regulations. Although related work often focuses on privacy-preserving community detection algorithms, including differential privacy, in this paper the focus was on the anonymized data itself. Channel membership information was used to build a weighted social graph and groups of interest were identified using popular community detection algorithms. Graphs based on channel membership data resembled interest groups within the network satisfactorily but failed to capture the organizational structure. Furthermore, a statistical evaluation and a user study were conducted to measure the performance of the recommender prototype. The statistical evaluation showed promising results, while the user study yielded mediocre satisfaction of the participants and revealed various potential shortcomings and limitations of the recommender system and user dataset retrieved from the notification system.**

Index Terms*—*Data Privacy; Open Data; Large Organizations; Clustering

I. Introduction

This paper is an extension of our previous work on privacy-protected identification of user clusters in large organizations, presented in [1].

Large organizations are estimated to generate a median of 300 terabytes (TB) of data weekly [2]. Data are generated from the use of various methods of communication (chat, email, face-to-face, phone, short message service, social media) between organization members, data sharing tools, internal processes, different hardware units (mobile phones, tablets, laptops, etc.) and more [2]. The publication of these data to be used for analysis and research has been an excellent source of information for researchers, promoting innovation and advancements in various areas and facilitating cooperation between various groups [3][4]. In this context, the term used to describe the data available freely to anyone to use for analysis and research is open data [5]. There have been different initiatives for collaboration based on open data, such as the Netflix Prize, OpenStreetMap, CERN (Conseil européen pour la recherche nucléaire) Open Science Initiative, Open City Initiatives, and more [3][5][6]. The purpose of these projects has been to improve existing technologies and algorithms and facilitate innovation and collaboration [3]. In addition to these projects, organizations internally analyze user behavior and user data and create new or improve existing services, generally relying on continuous user surveys and behavior tracking while invading their privacy [7].

The sharing of personal data that contain identifiers, quasi-identifiers, and sensitive attributes has been identified as a common issue with similar projects [3]. Sensitive and personal data should not be accessed freely; organizations must protect and secure them. To achieve this, organizations usually secure themselves and do not release this type of data. By doing so, the possible benefits available from private data are not explored. To avoid privacy breaches and publish organizational data, multiple data privacy preservation techniques were developed. Most of them are based on pseudo-anonymization or complete anonymization of the data [8]. The use of anonymized private data led to privacy-preserving data analysis methods. These methods offer a way to use private data safely, considering privacy requirements [9].

CERN has always stood for the principles of open data and open science, facilitating collaborative, transparent, and reproducible research and development whose results are publicly

available [6]. One such initiative is the CERN anonymized Mattermost dataset, which contains anonymized user data, relationships between users, organizations, buildings, teams, and channels. The goal of this dataset is to facilitate innovation for channel recommendations, user clustering, feature extractions, and others [10].

This paper, which is an extension of our previous work on privacy-protected identification of user clusters in large organizations, aims to analyze the provided CERN datasets and determine the privacy aspects and attributes that can be used for privacy-sensitive clustering methods and applications in recommender systems [1]. Based on the observations stated above, more specifically, the main research questions are as follows:

- **RQ1:** Which user information can be extracted from the anonymized Mattermost organizational open data?
- **RQ2:** Is it possible to detect user groups without invading user privacy?
- **RQ3:** What is the performance of clustering algorithms when applied to sparse anonymized user data?

The remainder of this paper is organized as follows. Section II covers the literature overview and discusses current topics in privacy-preserving data mining, open data, sparse data, clustering methodologies, and clustering evaluation metrics. In Section III, we discuss and describe the CERN Mattermost dataset. Section IV focuses on the analysis of various clustering methodologies and algorithms on the previously mentioned data and evaluates the best performing algorithms on the notification system dataset. Section V describes the user evaluation and analysis of the application of clustering algorithms on sparse anonymized user data from the notification system. Section VI discusses the findings of the statistical and user evaluation and explains the use of clustering methodologies. We conclude the work in Section VII with a discussion of the research questions and future work.

II. Background and Related Work

A. Networks and Graphs

Networks are defined as interconnected or interrelated chains, groups, or systems and can be found in a variety of areas, such as the World Wide Web, connections of friends, connections between cities, connections in our brain, power line links, and citation links. In essence, a network is a set of interconnected entities, which we call nodes, and their connections, which we call links. The nodes describe all types of entities, such as people, cities, computers, Web sites, and so on. Links define relationships or interactions between these entities, such as connections between people, flights between airports, links between Web pages, connections between neurons, and more. A special type of network is a social network. It is a group of people connected by a type of relationship (friendship, collaboration, or acquaintance) [11].

The data structure commonly used for the representation of networks is called a graph. A graph is defined as a set of connected points, called vertices (or nodes), that are connected via edges, also called links. The set of vertices is denoted as $V = \{v_1, v_2, v_3, \dots\}$, while the set of edges is denoted as $E = \{e_1, e_2, e_3, \dots\}$. The resulting graph G consists of a set of vertices V and a set of edges E that connect them and can be written as $G = (V, E)$. Two vertices connected by an edge are called adjacent or neighbors, and all vertices connected to a vertex are called neighborhood [12].

Graphs have a variety of measures associated with them. These measures can be classified as global measures and nodal measures. Global measures refer to the global properties of a graph, whereas nodal measures refer to the properties of nodes. The most important measures are degree measures, strength measures, modularity measures, and clustering coefficient measures. The degree measure is a nodal. It is the sum of edges connected to a node. The sum of the weights of all edges connected to a node is defined as the strength measure, while the extent to which a graph divides into clearly separated communities (that is, subgraphs or modules) is described by modularity measures [13].

B. Clustering Methods

Fundamental tasks in data mining are clustering and classification, among others. Clustering is applied mostly for unsupervised learning problems, while classification is used as a supervised learning method. The goal of clustering is descriptive, and that of classification is predictive [14].

Clustering is used to discover new sets of groups from samples. It groups instances into subsets using different measures. Measures used to determine similar or dissimilar instances are classified into distance measures and similarity measures. Different clustering methods have been developed, each of them using different principles. Based on research, clustering can be divided into five different methods: hierarchical, partitioning, density-based, model-based clustering, and grid-based methods [14][15].

Hierarchical Methods - Clusters are constructed by recursively partitioning items in a top-down or bottom-up fashion. For example, each item is initially a cluster of its own; then the clusters are merged based on a measure until the desired clusters are formed [15].

Partitioning Methods - These methods typically require a predetermined number of clusters. Items are moved between different predetermined clusters based on different metrics (error-based metrics, similarity metrics, distance metrics) until desired clusters are formed. To achieve the optimal cluster distribution, extensive computation of all possible partitions is required. Greedy heuristics are used for this computation because it is not feasible to calculate all possible partitions under time constraints [14].

Density-Based Methods - These methods are based on the assumption that clusters are formed according to a specific probability distribution. The aim is to identify clusters and their distribution parameters. The distribution is assumed to be a combination of several distributions [16].

Model-based Clustering methods - Unlike the previously mentioned methods, which group items based on similarity

and distance metrics, these methods attempt to optimize the fit between the input data and a given mathematical model [17].

Grid-based methods - The previous clustering methods were data-driven, while grid-based methods are space-driven approaches. They partition the item space into cells disconnected from the distribution of the input. The grid-based clustering approach uses a multiresolution grid data structure. It groups items into a finite number of cells that form a grid structure on which all clustering operations are performed. The main advantage of the approach is its faster processing time [18].

C. Evaluation Metrics

According to the literature, there are two main types of evaluation metrics for recommendation systems; they are statistical accuracy metrics (SAM) and decision support accuracy metrics (DSAM) [19]. SAM methods such as Mean Absolute Error (MAE) evaluate the precision of a recommender system by comparing the predicted values with the actual ratings of the original predictions and ratings [20][21]. DSAM determines the effectiveness of a prediction engine by helping users select relevant items from the available ones. The most common measures are sensitivity, specificity, and precision. Using the right model validation techniques helps to understand the models and estimate the performance of a model [19].

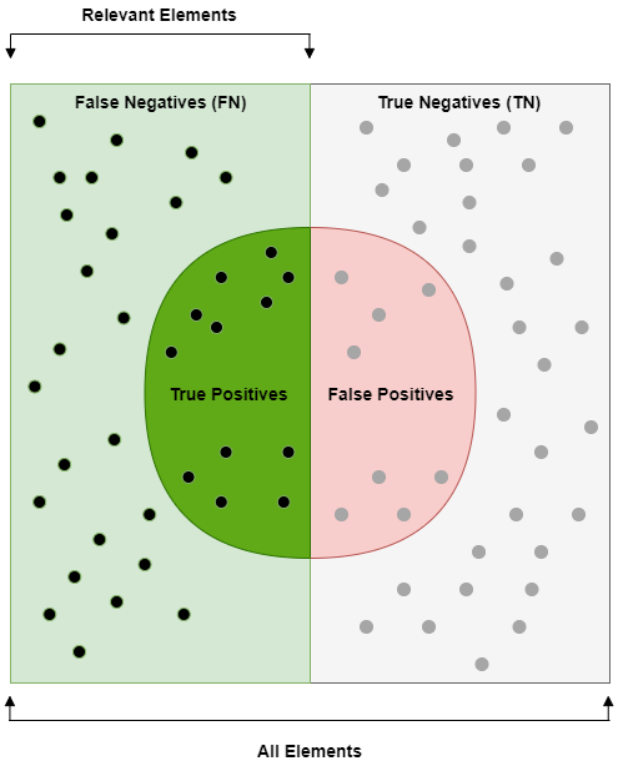

Figure 1. Classification Evaluation Representation based on [22]

Figure 1 illustrates the main elements used for classification evaluation. True positive values are values when the actual and predicted conditions are positive. False positive values are states in which the predicted value is positive, but the actual value is negative. The true negative value indicates that the actual and predicted conditions are the same and are both negative. The state in which the actual condition is positive but the prediction is negative is referred to as false negative values. [23].

Actual positive (AP) values, as seen in equation (1), refer to the number of true positives (TP) together with the number of false positives (FP).

$$AP = TP + FN \tag{1}$$

Actual negative (AN) values, as seen in equation (2), refer to the number of false positives (FP) together with the number of true negatives (TN).

$$AN = FP + TN \tag{2}$$

Predicted positive (PP) values, as seen in equation (3), refer to the number of true positives together with the number of false negatives.

$$PP = TP + FP \tag{3}$$

The predicted negative (PN), as seen in equation (4), refers to the number of false negatives along with the number of true negatives [23].

$$PN = FN + TP \tag{4}$$

Sensitivity describes the ratio of correct predictions to all actual positive conditions and is calculated as shown in equation (5) [24].

$$sensitivity = \frac{TP}{AP} \tag{5}$$

The specificity describes the ratio of correct rejections to all actual negative conditions and is calculated as shown in equation (6) [24].

$$specificity = \frac{TN}{AN} \tag{6}$$

The precision describes the ratio of correct predictions out of all positively predicted values and is calculated as shown in equation (7) [25].

$$precision = \frac{TP}{PP} \tag{7}$$

According to [26], the f-score is a measure of the accuracy of the prediction. It is the harmonic mean between precision and sensitivity and is calculated as shown in Equation (8).

$$f_{score} = 2\frac{precision * sensitivity}{precision + sensitivity} \tag{8}$$

D. Open Data, Sparse Data, and Privacy-aware Data Analysis

Open Data describes data available without restrictions for anyone to use for analysis and research [5]. Open innovation is defined as the use of purpose-oriented inputs and outputs of knowledge to stimulate internal innovation while increasing the demands for external use of innovation, respectively. The goal of open innovation and open data is to increase accountability and transparency while providing new and efficient services [27].

Sparse data is characterized by a relatively high percentage of variables that do not contain actual information. These variables contain values such as "empty" or NA [28]. Sparse data bias is a statistical bias that results from unevenly distributed data. Models trained on sparse data can be biased towards more common observations, leading to poor performance on less common observations. It can occur in unbalanced datasets or when dealing with missing data [29].

Privacy-preserving analytics are a set of methods for collecting, measuring, and analyzing data that respect individual privacy rights. These methods allow data-driven decisions while still giving individuals control over personal data. Restricting access to the data could be found to restrict support for various types of data analysis. Adopting approaches to restricting information in the data so that they are free of identifiers and free of content with a high risk of individual identification. Techniques have been proposed to release data without revealing sensitive information for various applications. Interest in the development of privacy-preserving data mining algorithms has been growing over the years [30].

III. Dataset

The Mattermost dataset was extracted from an internal PostgreSQL (Structured Query Language) database and is accessible as a JSON (JavaScript Object Notation) formatted file [10]. It includes data from January 2018 to November 2021 with 21231 CERN users, 2367 Mattermost teams, 12773 Mattermost channels, 151 CERN buildings, and 163 CERN organizational units. The dataset states the relationships between Mattermost teams, Mattermost channels, and CERN users. It contains various pieces of information, such as channel creation, channel deletion times, user channel joining, and leave times. It also includes user-specific information, such as building and organizational units, messages, and the mention count. To hide identifiable information (e.g., team name, user name, channel name), the dataset was anonymized using a combination of techniques such as omitting attributes, hashing string values, and removing connections between users, teams, and channels. It is important to note that there are various other anonymization techniques, including pseudonymization, differential privacy, and k-anonymity, that could affect the results of privacy-preserving analytics in different ways. The usage of these methods can cause algorithms applied to anonymized datasets to perform differently, since each method introduces a certain level of information loss.

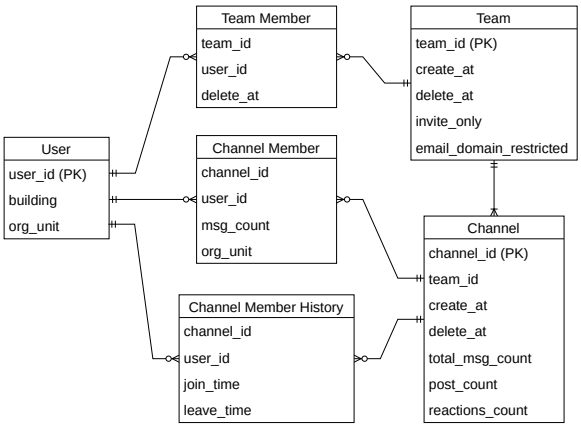

Figure 2. CERN Mattermost dataset Entity Relation Diagram

The entity relationship diagram shown in Figure 2 describes entities with data attributes and relationships between entities.

A. Data Transformation

The dataset was analyzed and prepared to filter out superfluous teams, channels, and users. According to the analysis, approximately 22.6% teams consist of one single person and can be removed as they form isolated nodes that do not contribute to the community structure.

Table I shows the five-number summary of the number of members within teams with more than one member. The five-number summary consists of three quartiles, Q_1, Q_2 or median, and Q_3, which divide the dataset into two parts with the lower part having 25%, 50% and 75% of the dataset's values, respectively. The other two values of the five-number summary consist of the minimum and maximum value of the dataset.

Using the quartiles from the five-number summary, the lower and upper team size fences can be calculated, which act as boundaries above or below which teams are considered outliers. The upper fence can be calculated by $UpperFence = Q_3 + 1.5 * IQR$, where IQR represents the interquartile range. IQR is defined as $IQR = Q_3 - Q_1$. This results in an upper bound of 51.5.

Table I
FIVE-NUMBER SUMMARY OF TEAMS WITH MORE THAN ONE MEMBER.

	Minimum	Q_1	Median	Q_3	Maximum
Team Members	2	4	10	23	4512

When counting the number of teams above that threshold, approximately 87.7% of the teams have less than 52 members. The lower fence is calculated by $LowerFence = Q_1 - 1.5 * IQR$ and yields -24.5. Since we do not have negative team sizes, we can limit the lower bound to 2, since team sizes of 1 are isolated nodes.

B. Graph Creation

Channel membership relations were used to generate graphs that act as a basis for community detection and user group analysis. A weighted edge is added between two users if they share the same channel, and the weight of the edge is increased for each additional channel they share. The idea behind channel membership for graph creation is that team members within CERN join channels related to their organization and work interest. Consequently, the more channels members have in common, the more likely they are to belong to the same organizational structure. The goal is to find the best communities that resemble the CERN organizational structure and communities.

IV. Evaluation

A. Algorithm Evaluation

Following the procedure described in Section III-B with an upper team threshold of 52, a weighted graph was produced. The igraph implementation of the Large Graph Layout (LGL) with 2000 iterations was used to visualize it [31].

LGL was used because it creates good layouts for a large number of vertices and edges and produces well-observable clusters. The graph produced is shown in Figure 3.

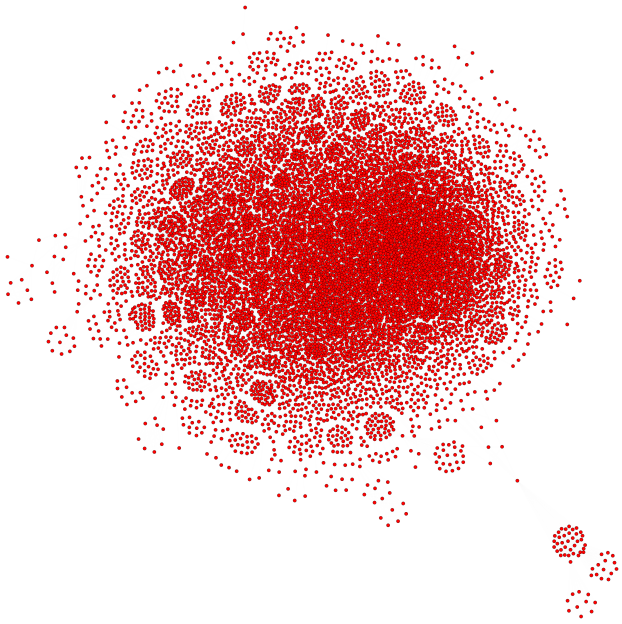

Figure 3. Graph based on channel membership relationship.

Table II lists several advanced clustering algorithms utilized to detect communities in the Mattermost dataset along with the corresponding evaluation results. The mentioned clustering algorithms were selected for evaluation because they were the commonly used algorithms for clustering and also available in the ones in the igraph library. Evaluation metrics considered include modularity, similarity, and the number of communities identified by each algorithm.

Of all the available algorithms presented in Table II, infomap (2), label propagation (3), and random walk (7) delivered the best performance with respect to modularity, similarity, and communities, as shown in Table II.

Random walk algorithms are based on the idea that a random walk on a graph tends to stay within a community and rarely cross over to other communities. It uses a spectral clustering approach to partition the graph into communities by performing a random walk on the graph and using the resulting probability distribution to compute a normalized Laplacian matrix. This algorithm is known for its ability to detect overlapping communities and has been shown to perform well on large-scale networks [38].

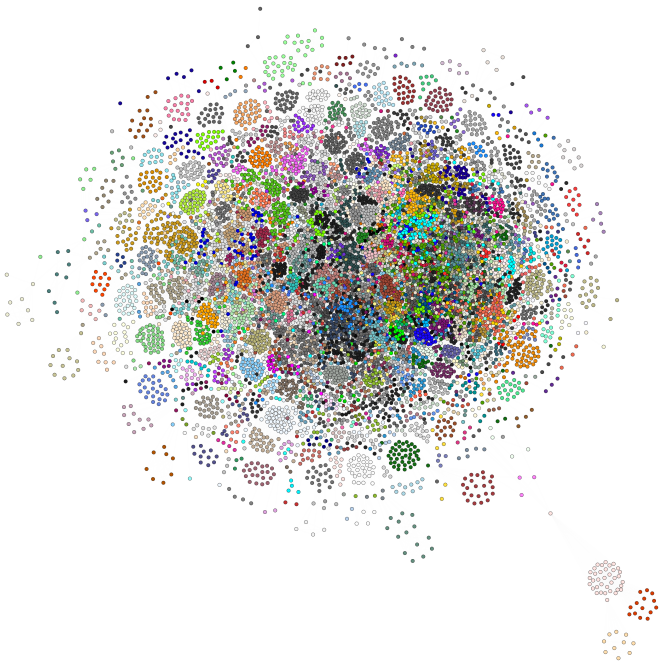

Figure 4. Communities detected by using the label propagation algorithm. A clear separation between individual clusters in the outer part of the graph can be observed.

The infomap algorithm utilizes random walks to assign special codes, known as Huffman codes, to each vertex and organizes them in a way that minimizes the description length measured in bits per vertex. These Huffman codes are binary strings assigned to objects based on their frequency, ensuring that objects visited more frequently are assigned shorter encodings, while less frequently visited objects receive longer ones. This algorithm has demonstrated its effectiveness in detecting hierarchical community structures and is widely recognized for its high accuracy [33].

The Label Propagation Algorithm begins by assigning a unique community label to each node in the network. These labels are propagated through the network iteratively. During each iteration, a node updates its label to the one that the majority of its neighbours have. The algorithm continues to propagate labels until convergence, where each node adopts the majority label of its neighbours or if the maximum number of iterations specified is reached. As the labels are propagated, densely connected nodes quickly reach a consensus on the

Table II
RESULTS INCLUDING FIVE-NUMBER SUMMARY OF SIMILARITIES BETWEEN MATTERMOST TEAMS AND FOUND COMMUNITY WITH DIFFERENT ALGORITHMS. VALUES WITHIN COLUMNS REPRESENT THE MEAN AND STANDARD DEVIATION OVER 25 ITERATIONS.

Algorithm	Communities	Modularity	Minimum [%]	Q_1 [%]	Median [%]	Q_3 [%]	Maximum [%]
1. Community structure through greedy optimization of modularity [32]	41 ± 0	0.75 ± 0.00	7.85 ± 0.00	23.43 ± 0.00	45.24 ± 0.00	66.67 ± 0.00	100 ± 0.00
2. Infomap community finding [33]	414 ± 3	0.71 ± 0.00	18.13 ± 1.18	46.52 ± 0.19	61.75 ± 0.68	75.97 ± 0.61	100.00 ± 0.00
3. Finding communities based on propagating labels [34]	463 ± 8	0.70 ± 0.00	15.68 ± 2.23	48.18 ± 1.07	61.25 ± 0.81	75.08 ± 0.28	100.00 ± 0.00
4. Community structure detecting based on the leading eigenvector of the community matrix [35]	43 ± 0.00	0.67 ± 0.00	5.85 ± 0.00	15.17 ± 0.00	26.92 ± 0.00	52.48 ± 0.00	95.65 ± 0.00
5. Finding community structure of a graph using the Leiden algorithm [36]	1290 ± 3	0.64 ± 0.00	2.04 ± 0.00	20.00 ± 0.00	42.86 ± 0.00	66.67 ± 0.00	100.00 ± 0.00
6. Finding community structure by multi-level optimization of modularity [37]	40 ± 2	0.78 ± 0.00	8.80 ± 0.77	14.79 ± 1.12	21.75 ± 1.64	50.87 ± 6.80	86.51 ± 6.57
7. Computing communities using random walks [38]	344 ± 0	0.72 ± 0.00	8.33 ± 0.00	55.56 ± 0.00	66.67 ± 0.00	80.00 ± 0.00	100.00 ± 0.00
8. Community detection based on statistical mechanics [39]	25 ± 0	0.77 ± 0.00	8.10 ± 0.71	11.23 ± 0.79	14.06 ± 1.05	17.700 ± 1.39	31 ± 8.51

label, leading to the disappearance of many labels. At the end of the propagation, only a few labels remain, and nodes with the same label are considered to belong to the same community. This algorithm is known for its simplicity, efficiency, and scalability [34] .

Calculating the community structure with the highest modularity value (community_optimal_modularity) and community structure detection based on edge betweenness (community_edge_betweenness) were not feasible in practice, since the runtime was too long. Figure 4 displays the result of the label propagation algorithm applied to the graph created previously. Each community is assigned a unique color to observe the separation of individual clusters. The label propagation algorithm finds communities with slightly less similarity than the infomap algorithm, which performs best with respect to similarity measurement. However, it finds many and much more detailed communities.

Figure 5 represents the similarities of the users between the communities found and the Mattermost teams while Figure 6 illustrates the results of 10 iterations as violin plots. An upper threshold of 52 for the teams was used for this figure, as described later in this section.
Of all communities detected, 75% have similarities above 47.79%, 50% have similarities above 61.18%, and 25% have similarities above 74.99%. Similarities are measured by comparing the discovered community with all Mattermost teams and counting the common members in both sets. The percentage value of the Mattermost team with the most common members is used.

Depending on the number of communities found, there may be overlaps such that one team fits multiple communities as the best match. This might be the case where the size of communities is smaller than the size of teams, such that communities form subgroups of the teams. However, less than 0.01% of the communities discovered correspond to the same Mattermost team. The average size of the communities discovered is 20 ± 23, the minimum is 2, the first quartile Q_1 is 6, the median is 13, the third quartile Q_3 is 26, and the maximum is 421.

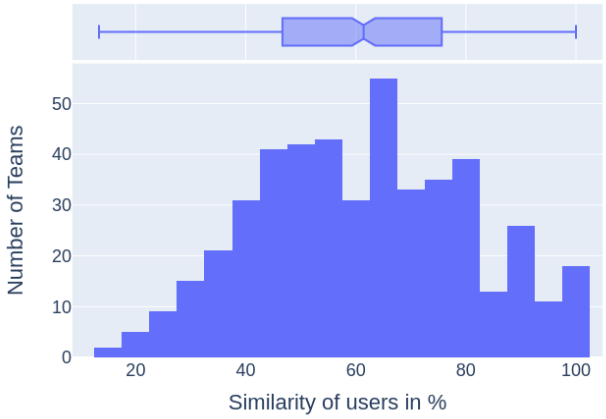

Figure 5. Sample run showing similarities of users between found communities and Mattermost teams.

Figure 7 shows the similarities of users between the detected communities and the organizational units with a threshold of 52, and Table III shows the parameters of this figure in detail. We can observe that the similarities are relatively low, with 75% of communities having at most 5.07% similarity. This indicates that the discovered communities generally do not resemble organizational units very well. The main reason is that Mattermost teams often consist of members of different organizational units. This is especially the case where users form groups of interest that are not related to work. This results in discovered communities that capture the teams and structure

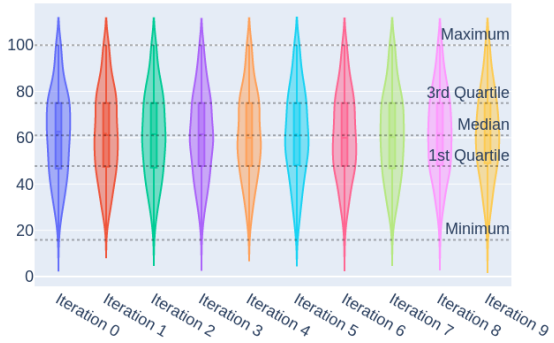

Figure 6. Similarities between discovered communities and Mattermost teams over iterations with threshold 52.

within Mattermost rather than the organizational structure of CERN.

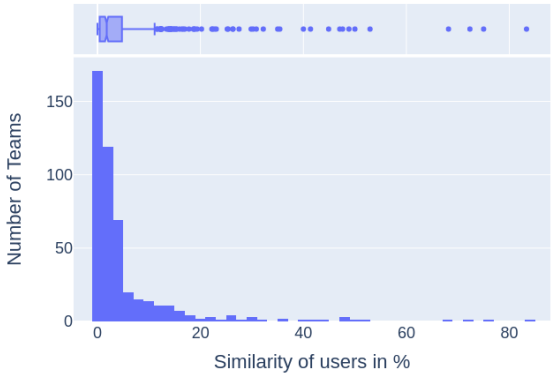

Figure 7. Sample run showing similarities of users between found communities and organizational units.

When creating the graph, two different methods were used and compared for filtering teams and channels. With the first method, the threshold was used as an upper limit for team members, i.e., only the channels of the teams below the threshold are considered for creating the graph.

Table III
FIVE-NUMBER SUMMARY OF SIMILARITIES BETWEEN ORGANIZATIONAL UNITS AND DISCOVERED COMMUNITIES USING LABEL PROPAGATION ALGORITHM. VALUES WITHIN COLUMNS REPRESENT MEAN AND STANDARD DEVIATION OVER 25 ITERATIONS IN PERCENT.

Minimum	Q_1	Median	Q_3	Maximum
0.0 ± 0.0	0.42 ± 0.04	1.77 ± 0.04	5.07 ± 0.29	74.68 ± 4.55

Due to the random nature of the label propagation algorithm, the results of each run differ slightly. The mean and standard derivation over 25 runs were calculated to obtain more precise results. With the second method, the threshold was used as an upper limit for channel members, i.e., all channels below the threshold are considered for creating the graph. The second method yields more nodes, but fewer communities, and slightly less similarity to the first. Due to this, the first method was preferred.

B. Statistical Evaluation

For the statistical evaluation, the user-channel subscription information of the most recent snapshot of the notification system database was collected. The dataset was then divided into a training set and a validation set. For a correct recommendation, an edge contained in the validation set should be in the list of recommendations for a given user. This is a classification problem as described in Section II-C. The graph is created from the training set and communities are discovered. The entire dataset contains 1270 user-channel edges. When splitting them, 1016 edges fall into the training set, while the remaining 254 edges fall into the validation set. However, there are some points that need to be considered:

1) Some users might only be part of one channel. If they are put in the validation set, they will never be recommended.
2) Some users are removed when creating the graph, as they might only be in groups above the upper user limit. If they are removed from the graph, the user-channel edges of the validation set will not be recommended.
3) There are recommendations that might be justified even though they are not in the validation set.

The first two points can be addressed while creating the graph and the recommendations, while the third point cannot. However, the third point has a direct influence on the results, as many of these recommendations fall into the false positive group, directly influencing metrics such as precision and f-score. The results of the statistical evaluation are shown in Table IV.

Table IV
RESULTS OF THE STATISTICAL EVALUATION.

Algorithm	Label Propagation		Infomap	
	Mean	Standard Deviation	Mean	Standard Deviation
True positive	42	6	40	6
True negative	35026	1484	35741	1095
False positive	3320	1484	2605	1096
False negative	12	4	13	4
Specificity	0.91	-	0.93	-
Sensitivity	0.78	-	0.75	-
Precision	0.012	-	0.015	-
F-Score	0.024	-	0.029	-

The results show a correct hit rate (Table IV - Sensitivity) of 78% for the Label Propagation algorithm and 75% for the Infomap algorithm. The correct rejection rate (Table IV - Specificity) for the Label Propagation algorithm is 91% and 93% for the Infomap algorithm. The number of false positives is high due to the user-channel edges that are in the list of recommendations, but not in the validation set.

V. User Study

For the user study, specific users of the CERN notification system were invited to participate by joining a particular channel and their user data was collected for community detection. Their data was anonymized to respect their privacy. Then, the user-channel subscription information of the most recent snapshot of the notification system database was collected and used to create the user graph and discover communities. The results of the statistical evaluation in Section IV were used to select the algorithm to group the users and create graphs. Figure 8 shows the graph created from the user-channel information. The colors represent the communities, and the numbers on the nodes represent the selected anonymized users from CERN. Since the CERN IT department was the first to fully adopt the new notification system, most of the chosen users come from this department. This can also be seen in Figure 8, where most of the users are grouped together into the same community.

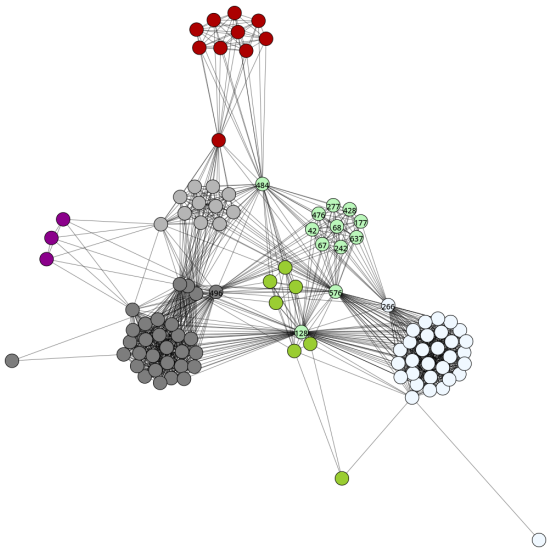

Figure 8. Graph with communities of the most recent user-channel information.

For each community, a list of channels is created to which the community members subscribed and the number of community members in each channel is enumerated. The list of channels is sorted by popularity in descending order, where the first channel contains most of the community members. This list acts as a list of recommendations for the community. The recommendations are created for each notification system user by choosing the first five channels of the most popular ones in the community to which the user has not already subscribed. The survey containing the recommended channels was sent to all participants through the notification system. Each participant was asked to rate the recommendations as personally relevant or irrelevant.

The survey was sent to 15 users of the notification system. Of the initial users, 13 responded, making the response rate 86.66%. Table V shows the results of the user study. Users were marked as active or inactive depending on their interaction with the system.

Table V
RESULTS OF THE USER EVALUATION.

User Id	Active	Relevant Channels	Irrelevant Channels	Precision
128	x	1	4	0.333
476	x	3	2	0.6
484	x	5	0	1
496	x	1	4	0.2
42		3	2	0.6
67		1	4	0.2
68		3	2	0.6
177		1	4	0.2
242		4	1	0.8
266		1	4	0.2
428		2	3	0.4
576		4	1	0.8
637		3	2	0.6

Active users are long-time members who use the system on a daily basis. While inactive users are users who recently joined the notification system or users who do not use the system daily.

The average precision on all participants for relevant and irrelevant channels is 50%, with no significant impact on user activity. Figure 9 shows the results as a graphical representation on a graph.

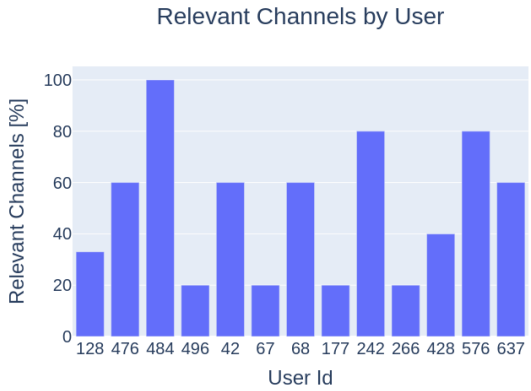

Figure 9. Relevant channels by user as shown in table V.

VI. Findings and Discussion

Based on Section IV with a higher threshold, more users are within teams and channels, increasing the edge weight between many different users. Due to this, the weight difference between the edges of the communities within and outside becomes smaller, resulting in fewer communities. Table VI shows the number of users, edges and the average and standard deviation of edge weights over different thresholds. Higher thresholds result in more nodes and edges, but the average weight decreases, as many users are only part of a few channels and teams. Without a threshold, the average weight increases due to channels increasing the weight for numerous users.

Table VI
NUMBER OF NODES, EDGES, AND AVERAGE AND STANDARD DEVIATION OF EDGE WEIGHTS OVER DIFFERENT THRESHOLDS.

Threshold	Nodes	Edges	Weight
52	9520	151501	2.94 ± 2.35
200	14906	809012	2.82 ± 2.25
500	17124	1909964	2.65 ± 1.88
1000	17948	3104814	2.53 ± 1.66
1500	18721	5000668	2.34 ± 1.58
None	19682	15194697	2.44 ± 1.62

Higher thresholds do not improve community discovery, as the typical size of teams is up to 52, as previously stated. On the basis of our experiments, the clustering tendency depicted by the modularity value decreased with higher thresholds and fewer communities were found.

The results of the user study show mediocre performance in terms of relevance, which, however, might not be a good indication of performance due to various factors that affect the outcome:

1) Low number of participants
2) Low number of interactions between participants
3) Participants are primarily from the IT department
4) High chance that participants already subscribed to channels that are relevant to them
5) Low channel diversity across the whole notification system

Since these points mentioned above can only be addressed when the notification system has more diverse channels and a higher number of users, especially from other departments, the user study gives a good first impression of the prototype.

VII. Conclusion and Future Work

Data privacy preservation is one of the key issues in open innovation and open data. This research aims to analyze the provided CERN dataset and determine privacy aspects and attributes that can be extracted and used for privacy-protected identification of user clusters in large organizations. Information such as user group matching has been the focus of this research. Different clustering algorithms were used for user group detection without invading user privacy. To achieve this, only user communication and interaction data from the CERN Mattermost dataset was used for cluster formation. The dataset includes 21231 CERN users, 2367 Mattermost teams, 12773 Mattermost channels, 151 CERN buildings, and 163 CERN organizational units. It was expected to rediscover an organizational structure that closely matches the organizational hierarchical structures (Organizational Units, Departments, Groups, Sections, etc.). Our research shows that fitting detected clusters to existing organizational structures was not successful and yielded poor results. Matching detected clusters with interest groups, such as Mattermost teams, produced satisfactory results. The main reason for this finding is that users interact and communicate with individuals who share their interests (the same channels or Mattermost teams). These individuals might not be in the same organizational units, or users from different organizational units might be in the same channel, introducing noise to the data.

The algorithm evaluation results also showed that the clustering tendency depicted by the modularity value decreased with higher thresholds and fewer communities were found. In addition, new metrics for weighting user-to-user connections could be used to identify not only interest groups, but also organizational connections between users.

Furthermore, the findings of the analysis of the CERN Mattermost dataset were applied to a new dataset retrieved from the CERN notification system. Since this dataset resembled the Mattermost dataset, it was expected that the clustering algorithms produce similar results on this dataset. The user study showed that the average precision of the best-performing clustering algorithm is 50%. The decrease in performance could be a product of the high level of sparsity in the dataset, the low number of existing channels to recommend, and the high level of users already subscribed to existing channels.

Future work might include the use of novel neural network-based clustering algorithms. Additionally, new metrics for weighting user-to-user connections could be used to identify not only interest groups, but also organizational connections between users. In addition to these improvements, the data could be connected to external data to identify certain teams, users, or organizational structures, and the level of communication between them.

To fully evaluate the effectiveness of channel recommendations, it would be beneficial to provide a baseline of relevant channels. The baseline would be formulated per user by proposing random channels to users and checking their relevance. This would allow for a better understanding of the impact of the recommendation and provide better means of evaluation. We suggest that future research includes such an evaluation when the notification system is fully adopted by other groups at CERN and when more users engage with the system.

References

[1] I. Jakovljevic, M. Pobaschnig, C. Gütl, and A. Wagner, "Privacy aware identification of user clusters in large organisations based on anonymized mattermost user and channel information", in *Proceedings of the 11th International Conference on Data Science, Technology and Applications - IARIA DATA ANALYTICS*, 2022, pp. 62–67, ISBN: 978-1-61208-994-2. [Online]. Available: https://www.thinkmind.org/index.php?view=article&articleid=data_analytics_2022_2_80_60050.

[2] I. Jakovljevic, A. Wagner, and C. Gütl, "Open search use cases for improving information discovery and information retrieval in large and highly connected organizations", 2020. DOI: 10.5281/zenodo.4592449. [Online]. Available: https://doi.org/10.5281/zenodo.4592449.

[3] J. Zhang, Y. Wang, Z. Yuan, and Q. Jin, "Personalized real-time movie recommendation system: Practical prototype and evaluation", *Tsinghua Science and Technology*, vol. 25, pp. 180–191, Apr. 2020. DOI: 10.26599/TST.2018.9010118.

[4] G. Navarro-Arribas, V. Torra, A. Erola, and J. Castellà-Roca, "User k-anonymity for privacy preserving data mining of query logs", *Information Processing Management*, vol. 48, no. 3, pp. 476–487, 2012, Soft Approaches to IA on the Web, ISSN: 0306-4573. DOI: https://doi.org/10.1016/j.ipm.2011.01.004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306457311000057.

[5] S. Antony and D. Salian, "Usability of open data datasets", in Oct. 2021, pp. 410–422, ISBN: 978-3-030-89021-6. DOI: 10.1007/978-3-030-89022-3_32.

[6] K. Naim *et al.*, "Pushing the Boundaries of Open Science at CERN: Submission to the UNESCO Open Science Consultation", Jul. 2020. DOI: 10.17181/CERN.1SYT.9RGJ. [Online]. Available: http://cds.cern.ch/record/2723849.

[7] P. Rao, S. Krishna, and A. Kumar, "Privacy preservation techniques in big data analytics: A survey", *Journal of Big Data*, vol. 5, pp. 1–12, Sep. 2018. DOI: 10.1186/s40537-018-0141-8.

[8] I. Jakovljevic, C. Gütl, A. Wagner, and A. Nussbaumer, "Compiling open datasets in context of large organizations while protecting user privacy and guaranteeing plausible deniability", *In Proceedings of the 11th International Conference on Data Science, Technology and Applications (DATA 2022)*, pp. 301–311, 2022, ISSN: 2184-285X. DOI: 10.5220/0011265700003269.

[9] S. R. M. Oliveira and O. R. Zaiane, "A privacy-preserving clustering approach toward secure and effective data analysis for business collaboration", *Computers Security*, vol. 26, no. 1, pp. 81–93, Feb. 2007, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2006.08.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404806001222.

[10] I. Jakovljevic, C. Gütl, A. Wagner, M. Pobaschnig, and A. Mönnich, "Cern anonymized mattermost data", version 1, Mar. 2022. DOI: 10.5281/zenodo.6319684. [Online]. Available: https://doi.org/10.5281/zenodo.6319684 (visited on 06/27/2022).

[11] F. Menczer, S. Fortunato, and C. A. Davis, *A first course in network science*. Cambridge University Press, 2020, ISBN: 9781108471138. DOI: 10.1017/9781108653947. [Online]. Available: https://www.cambridge.org/highereducation/books/first-course-in-network-science/EE22722F27519D8BB1443C7225C57BAF.

[12] V. Voloshin, *Introduction to Graph and Hypergraph Theory*. Nova Kroshka Books, Jan. 2013, p. 231, ISBN: 978-1606923726.

[13] L. Tang and H. Liu, *Community Detection and Mining in Social Media*. Morgan Claypool Publishers, Jan. 2010, vol. 2. DOI: 10.2200/S00298ED1V01Y201009DMK003. [Online]. Available: https://www.morganclaypool.com/doi/abs/10.2200/S00298ED1V01Y201009DMK003 (visited on 08/14/2022).

[14] L. Rokach and O. Maimon, "Clustering methods", in *Data Mining and Knowledge Discovery Handbook*. Springer US, 2005, pp. 321–352, ISBN: 978-0-387-25465-4. DOI: 10.1007/0-387-25465-X_15. [Online]. Available: https://doi.org/10.1007/0-387-25465-X_15.

[15] C. Hennig, "An empirical comparison and characterisation of nine popular clustering methods", *Advances in Data Analysis and Classification*, vol. 16, no. 1, pp. 201–229, Mar. 2022, ISSN: 1862-5355. DOI: 10.1007/s11634-021-00478-z. [Online]. Available: https://doi.org/10.1007/s11634-021-00478-z (visited on 06/27/2022).

[16] J. D. Banfield and A. E. Raftery, "Model-based gaussian and non-gaussian clustering", *Biometrics*, vol. 49, no. 3, pp. 803–821, 1993, ISSN: 0006341X, 15410420. [Online]. Available: http://www.jstor.org/stable/2532201 (visited on 06/27/2022).

[17] P. D. McNicholas, "Model-based clustering", *Journal of Classification*, vol. 33, no. 3, pp. 331–373, Oct. 2016, ISSN: 1432-1343. DOI: 10.1007/s00357-016-9211-9. [Online]. Available: https://doi.org/10.1007/s00357-016-9211-9 (visited on 06/27/2022).

[18] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2012, ISBN: 0123814790. [Online]. Available: http://www.amazon.de/Data-Mining-Concepts-Techniques-Management/dp/0123814790/ref=tmm_hrd_title_0?ie=UTF8&qid=1366039033&sr=1-1.

[19] A. Bobic, I. Jakovljevic, C. Gütl, J. Le Goff, and A. Wagner, "Implicit user network analysis of communication platform open data for channel recommendation", in *9th International Conference on Social Networks Analysis, Management and Security - SNAMS 2022*, Apr. 2022, pp. 1–8. DOI: 10.1109/SNAMS58071.2022.10062597. [Online]. Available: https://ieeexplore.ieee.org/document/10062597.

[20] N. Good *et al.*, "Combining collaborative filtering with personal agents for better recommendations", in *Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*, American Association for Artificial Intelligence, 1999, pp. 439–446, ISBN: 0262511061. [Online]. Available: https://dl.acm.org/doi/10.5555/315149.315352.

[21] T. Chai and R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?", *Geoscientific Model Development*, vol. 7, no. 3, pp. 1247–1250, Jan. 2014. DOI: 10.5194/gmdd-7-1525-2014. [Online]. Available: https://gmd.copernicus.org/articles/7/1247/2014/.

[22] "Sensitivity and specificity". (Jan. 2023), [Online]. Available: https://en.wikipedia.org/wiki/Sensitivity_and_specificity.

[23] T. Fawcett, "An introduction to ROC analysis", *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006, ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2005.10.010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016786550500303X.

[24] W. Mumtaz, S. S. Ali, A. Mohd Zasin, and A. Malik, "A machine learning framework involving eeg-based functional connectivity to diagnose major depressive disorder (MDD)", *Medical biological engineering computing*, vol. 56, Jul. 2017. DOI: 10.1007/s11517-017-1685-z.

[25] T. de Greef, S. Masroor, M. A. Peletier, and R. Pendavingh, "Precision and sensitivity in detailed-balance reaction networks", *SIAM Journal on Applied Mathematics*, vol. 76, no. 6, pp. 2123–2153, 2016. DOI: 10.1137/15M1054869. eprint: https://doi.org/10.1137/15M1054869. [Online]. Available: https://doi.org/10.1137/15M1054869.

[26] N. Ye, K. M. A. Chai, W. S. Lee, and H. L. Chieu, "Optimizing f-measures: A tale of two approaches", in *Proceedings of the 29th International Coference on International Conference on Machine Learning*, Omnipress, 2012, pp. 1555–1562, ISBN: 9781450312851. [Online]. Available: http://dblp.uni-trier.de/db/conf/icml/icml2012.html#NanCLC12.

[27] J. West, A. Salter, W. Vanhaverbeke, and H. Chesbrough, "Open innovation: The next decade", *Research Policy*, vol. 43, no. 5, pp. 805–811, Jun. 2014, ISSN: 0048-7333. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0048733314000407 (visited on 08/14/2022).

[28] Oracle. "Oracle9i olap services". (2023), [Online]. Available: https://docs.oracle.com/cd/A91034_01/DOC/olap.901/a86720/esdatao6.htm.

[29] S. Greenland, M. A. Mansournia, and D. G. Altman, "Sparse data bias: A problem hiding in plain sight", *British Medical Journal*, vol. 353, Apr. 2016. DOI: 10.1136/bmj.i1981. [Online]. Available: https://www.bmj.com/content/352/bmj.i1981.

[30] I. Pramanik *et al.*, "Privacy preserving big data analytics: A critical analysis of state-of-the-art", *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 11, pp. 207–218, Jan. 2021. DOI: https://doi.org/10.1002/widm.1387. [Online]. Available: https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/widm.1387 (visited on 08/14/2022).

[31] A. Adai, S. Date, S. Wieland, and E. Marcotte, "Lgl: Creating a map of protein function with an algorithm for visualizing very large biological networks", *Journal of molecular biology*, vol. 340, pp. 179–90, Jul. 2004. DOI: 10.1016/j.jmb.2004.04.047. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022283604004851.

[32] M. Newman and M. Girvan, "Finding and evaluating community structure in networks", *Physical Review E*, vol. 69, no. 2, p. 026113, Feb. 2004, ISSN: 1539-3755, 1550-2376. DOI: 10.1103/PhysRevE.69.026113. [Online]. Available: http://www.cse.cuhk.edu.hk/~cslui/CMSC5734/newman_community_struct_networks_phys_rev.pdf.

[33] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure", *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, Jan. 29, 2008, ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.0706851105. arXiv: 0707.0609. [Online]. Available: http://arxiv.org/abs/0707.0609.

[34] A. Rezaei, S. M. Far, and M. Soleymani, "Near linear-time community detection in networks with hardly detectable community structure", *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, pp. 65–72, 2015. DOI: 10.1145/2808797.2808903. [Online]. Available: https://doi.org/10.1145/2808797.2808903.

[35] M. E. J. Newman, "Modularity and community structure in networks", *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 23, pp. 8577–8582, Jun. 2006, ISSN: 0027-8424. DOI: 10.1073/pnas.0601602103. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1482622/.

[36] V. Traag, L. Waltman, and N. J. van Eck, "From louvain to leiden: Guaranteeing well-connected communities", *Scientific Reports*, vol. 9, no. 1, pp. 5233–5233, Dec. 2019, ISSN: 2045-2322. DOI: 10.1038/s41598-019-41695-z. [Online]. Available: http://arxiv.org/abs/1810.08473.

[37] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks", *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, pp. 10008–10020, Oct. 2008, ISSN: 1742-5468. DOI: 10.1088/1742-5468/2008/10/P10008. [Online]. Available: http://arxiv.org/abs/0803.0476.

[38] P. Pons and M. Latapy, "Computing communities in large networks using random walks", *Proceedings of the 20th International Conference on Computer and Information Sciences*, pp. 284–293, 2005. DOI: 10.1007/11569596_31. [Online]. Available: https://doi.org/10.1007/11569596_31.

[39] J. Reichardt and S. Bornholdt, "Statistical Mechanics of Community Detection", *Physical Review E*, vol. 74, no. 1, p. 016110, Jul. 2006, ISSN: 1539-3755, 1550-2376. DOI: 10.1103/PhysRevE.74.016110. [Online]. Available: http://arxiv.org/abs/cond-mat/0603718.