Privacy Protected Identification of User Clusters in Large Organizations based on Anonymized Mattermost User and Channel Information

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

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

Christian Gütl
ISDS
Graz University of Technology
Graz, Austria
e-mail: c.guetl@tugraz.at

Andreas Wagner
IT Department
CERN
Geneva, Switzerland
e-mail: andreas.wagner@cern.ch

***Abstract*—Oversharing exposes risks such as improved targeted advertising and sensitive information leakage. Requiring only the bare minimum of data diminishes these risk factors while simultaneously increasing the privacy of each individual user. Using anonymized data for finding communities enables new possibilities for large organizations under strong data protection regulations. While related work often focuses on privacy-preserving community detection algorithms including differential privacy, in this paper, the focus was set 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 satisfactorily resembled interest groups within the network but failed to capture the organizational structure.**

***Keywords*—Data Privacy. Open Data. Large Organizations. Clustering.**

I. INTRODUCTION

It is estimated that a median of 300 terabytes (TB) of data is generated by large organizations on a weekly basis [1]. The data is 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 [1]. Publishing this data to be used for analysis and research has been a excellent source of information for researchers, promoting innovation and advancements in various areas while facilitating cooperation between diverse groups [2][3]. In this context, the term used to describe data available freely for anyone to use for analysis and research is open data [4]. 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 [2][4][5]. The purpose of these projects has been to improve existing technologies and algorithms and facilitate innovation and collaboration [2]. Besides these projects, organizations internally analyze user behavior and user data and create new or improve existing services, usually relying on continuous user surveying and behavior tracking while invading their privacy [6].

Sharing of personal data that contains identifiers, quasi-identifiers, and sensitive attributes has been identified as a common issue with similar projects [2]. Sensitive and personal data should not be accessed freely; organizations have to protect and secure it. To achieve this, organizations usually secure and do not release this type of data. By doing so, possible benefits available from private data are not explored. To avoid privacy breaches and to publish organizational data, multiple privacy-preserving techniques for data were developed. Most of them are based on pseudo-anonymization or full anonymization of data [7]. Utilization of anonymized private data gave rise to privacy-preserving data analytics methods. These methods offer a way to utilize private data safely, by considering privacy requirements [8].

CERN always stood for principles of open data and open science, facilitating research and development that is collaborative, transparent and reproducible and whose outputs are publicly available [5]. One such initative is the CERN anonymized Mattermost data set, which contains anonymized user data, relationships between users, organizations, building, teams and channels. The goal of this data set is to facilitate innovation for channel recommendations, user clustering, feature extractions, and others [9].

This research aims to analyze the provided CERN data set and determine privacy aspects and attributes that can be used for privacy aware clustering methods. Based on the observations stated above, more specifically, the main research questions are:

- **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?

The remainder of this paper is organized as follows: Section II covers the literature overview and discusses current topics

 ISBN: 978-1-61208-994-2

in privacy-preserving data mining, open data, and clustering methodologies. In Section III, we discuss and describe the CERN Mattermost data set. Section IV focuses on the findings from the data set and explains the usage of clustering methodologies on the previously mentioned data. We conclude the work in Section V with the discussion of the research questions and future works.

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. 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 among 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) [10].

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 that are connected by an edge are called adjacent or neighbors and all vertices that are connected to a vertex are called neighborhood [11].

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, while 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 (i.e., subgraphs or modules) is described by modularity measures [12].

B. Clustering Methods

Fundamental tasks in data mining are clustering and classifications, 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 [13].

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 [13][14].

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 clusters are merged based on a measure until desired clusters are formed [14].

Partitioning Methods - These methods typically require a pre-determined number of clusters. Items are moved between different pre-determined 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 [13].

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 [15].

Model-based Clustering methods - Unlike the previously mentioned methods, which cluster items based on similarity and distance metrics, these methods attempt to optimize the fit between the input data and a given mathematical model [16].

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 multi-resolution grid data structure. It groups items into a finite number of cells that form a grid structure on which all of the operations for clustering are performed. The main advantage of the approach is its faster processing time [17].

C. Open Data and Privacy-aware Data Analysis

Open Data describes data available without restrictions for anyone to use for analysis and research [4]. Open innovation is defined as the use of purposive inflows and outflows 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 [18].

Privacy-friendly analytics is a set of methods for collecting, measuring, and analyzing data respecting individual privacy rights. These methods allow for data-driven decisions while still giving individuals control over personal data. Restricting access to the data could be found to restrict to support of various kinds of data analysis. Adopting approaches of restricting information in the data so that they are free of identifiers and free of content with a high risk of individual

 ISBN: 978-1-61208-994-2

identification. Techniques for releasing data without disclosing sensitive information have been proposed for various applications. Interest in developing data mining algorithms that are privacy-preserving has been growing over the years [19].

III. Dataset

The Mattermost data set was extracted from an internal Postgre SQL (Structured Query Language) database and is accessible as JSON (JavaScript Object Notation) formatted file [9]. 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 data set states the relationships between Mattermost teams, Mattermost channels, and CERN users, and holds 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 mention count. To hide identifiable information (e.g., Team Name, User Name, Channel Name, etc.) the data set was anonymized. The anonymization was done by omitting attributes, hashing string values, and removing connections between users/teams/channels.

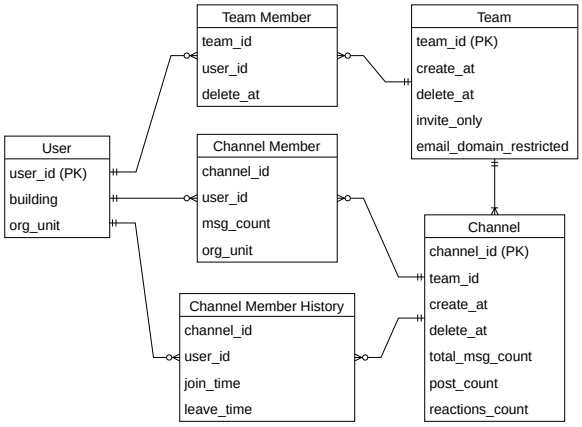

Fig. 1. CERN Mattermost data set Entity Relation Diagram

The entity relationship diagram shown in Figure 1 describes the entities with data attributes and relationships between the entities.

A. Data Transformation

The data set was analyzed and prepared to filter out superfluous teams, channels, and users. Based on the analysis, approximately 22.6% teams consist of only one 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 count 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, that divide the data set into two parts with the lower part having 25%, 50% and 75% of the data set's values, respectively. The other two values of the five-number summary consist of the minimum and the maximum value of the data set.

Using the quartiles from the five-number summary, the lower and upper team size fences can be calculated, which act as a boundaries above or below which teams are considered outliers. The upper fence can be calculated by $UpperFence = Q_3 + 1.5 * IQR$, where IQR stands for 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 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, as 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 between two users is added 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 the 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 belong to the same organizational structure. The goal is to find the best communities that resemble CERN's organizational structure and communities.

IV. Findings and Discussion

Following the procedure described in Section III-B with an upper team threshold of 52, a weighted graph was produced. The igraph's implementation of the Large Graph Layout (LGL) with 2000 iterations was used to visualize it [20]. LGL was used as it creates good layouts for large number of vertices and edges and produces well-observable clusters. The produced graph is displayed in Figure 2.

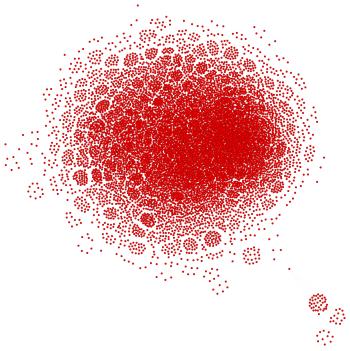

Fig. 2. Graph based on channel membership relationship.

 ISBN: 978-1-61208-994-2

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

Algorithm	Communities	Modularity	Minimum [%]	Q_1 [%]	Median [%]	Q_3 [%]	Maximum [%]
1. Community structure via greedy optimization of modularity [21]	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 [22]	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 [23]	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 [24]	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 [25]	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 [26]	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 [27]	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 [28]	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

To evaluate community detection algorithms and their effect on different modularity scores, the following ones were assessed:

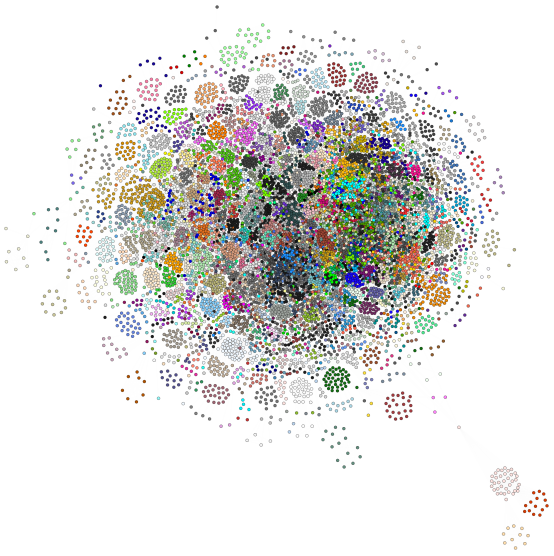

Fig. 3. Communities detected by using the label propagation algorithm. A clear separation between individual cluster in the outer part of the graph can be observed.

Out of all available algorithms, algorithms 2, 3, and 7 delivered the best performances concerning modularity, similarity, and communities, as shown in Table II. 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 3 displays the result of the label propagation algorithm applied to the previously created graph. Each community gets assigned a unique color, so the separation of individual clusters can be observed. The label propagation algorithm finds communities with slightly less similarity than the infomap algorithm, which performs best concerning similarity measurement. However, it finds many and much more detailed communities.

Figure 4 represents the similarities of users between found communities and the Mattermost teams and Figure 5 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 detected communities, 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.

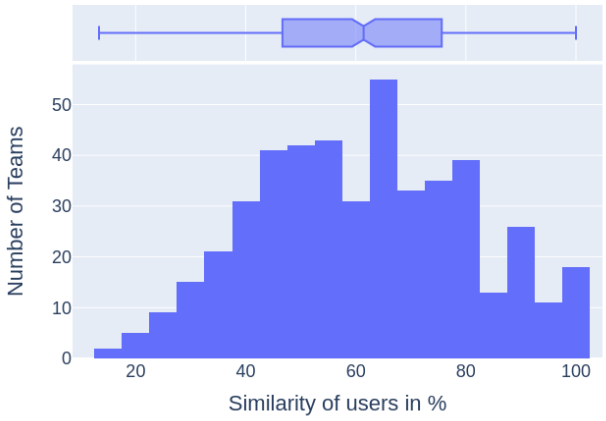

Fig. 4. Sample run showing similarities of users between found communities and Mattermost teams.

Depending on the number of communities found, there might be overlaps, such that one team fits multiple communities as the best match. This might be the case where the size

 ISBN: 978-1-61208-994-2

of communities is smaller than the size of teams, such that communities form subgroups of the teams. However, less than 0.01% of discovered communities are matched against the same Mattermost team. The average size of discovered communities is 20 $\pm$ 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 6 shows the similarities of users

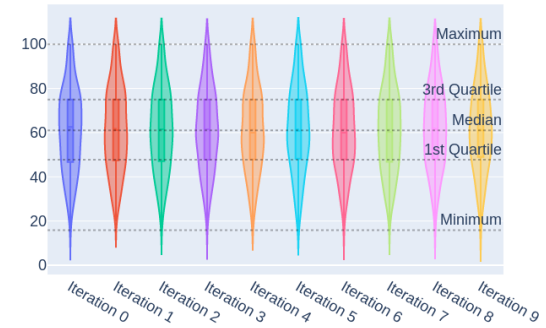

Fig. 5. Similarities between discovered communities and Mattermost teams over iterations with threshold 52.

between detected communities and the organizational units with a threshold of 52, and Table III states 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 capturing the teams and structure within Mattermost instead of the organizational structure of CERN.

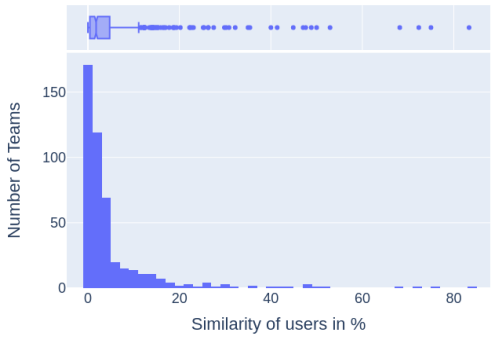

Fig. 6. Sample run showing similarities of users between found communities and organizational units.

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 $\pm$ 0.0	0.42 $\pm$ 0.04	1.77 $\pm$ 0.04	5.07 $\pm$ 0.29	74.68 $\pm$ 4.55

Because of the random nature of the label propagation algorithm, the results of each run slightly differ. The mean and standard derivation over 25 runs were calculated to get 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 than the first. Because of this, the first method was preferred.

TABLE IV
NUMBER OF NODES, EDGES, AND AVERAGE AND STANDARD DEVIATION OF EDGE WEIGHTS OVER DIFFERENT THRESHOLDS.

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

With a higher threshold, more users are within teams and channels, increasing edge weight between many different users. Because of this, the weight difference of the edges within and outside communities gets smaller, resulting in fewer communities. Table IV 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. With no threshold, the average weight increases due to channels increasing the weight for numerous users. Higher thresholds do not improve community discovery, as the typical size of teams is up to 52, as stated previously. Based on our experiments, the clustering tendency depicted by the modularity value decreased with higher thresholds, with fewer communities found.

V. Conclusion and Future Work

In conclusion, this research investigates which user information can be extracted from anonymized open data [7]. 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 communication and interaction user data was used for cluster formation. It was expected to rediscover organizational structure that closely matches the organizational hierarchical structures (organizational Units, Depart-

 ISBN: 978-1-61208-994-2

ments, 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 that share their interests (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.

Future work might include the usage of novel clustering algorithms that are based on neural networks. Additionally, new metrics for weighting user-to-user connections could be used to identify not only interest groups but also organizational connections between users. Besides these improvements, the data could be brought into connection with external data to identify certain teams, users, or organizational structures and the level of communication between them.

REFERENCES

[1] 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.

[2] 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.

[3] G. Navarro-Arribas, V. Torra, A. Erola, and J. Castellà-Roca, "User k-anonymity for privacy preserving data mining of query logs," *Inf. Process. Manag.*, vol. 48, no. 3, pp. 476–487, 2012. doi: 10.1016/j.ipm.2011.01.004. [Online]. Available: https://doi.org/10.1016/j.ipm.2011.01.004.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] S. R. M. Oliveira and O. R. Zaiane, "A privacy-preserving clustering approach toward secure and effective data analysis for business collaboration," *Comput. Secur.*, vol. 26, no. 1, pp. 81–93, Feb. 2007, issn: 0167-4048. doi: 10.1016/j.cose.2006.08.003. [Online]. Available: https://doi.org/10.1016/j.cose.2006.08.003.

[9] 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).

[10] F. Menczer, S. Fortunato, and C. A. Davis, *A first course in network science*. Cambridge University Press, 2020, isbn: 9781108471138.

[11] V. Voloshin, *Introduction to graph theory*. 2009, isbn: 9781606923740.

[12] L. Tang and H. Liu, *Community Detection and Mining in Social Media*, 1. Jan. 2010, vol. 2, Publisher: Morgan & Claypool Publishers. doi: 10.2200/S00298ED1V01Y201009DMK003. [Online]. Available: https://www.morganclaypool.com/doi/abs/10.2200/S00298ED1V01Y201009DMK003 (visited on 08/14/2022).

[13] L. Rokach and O. Maimon, "Clustering methods," in Jan. 2005, pp. 321–352. doi: 10.1007/0-387-25465-X_15.

[14] 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).

[15] 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).

[16] 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).

[17] 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.

[18] 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.

[19] 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).

[20] 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.

[21] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, p. 026 113, Feb. 2004, arXiv: cond-mat/0308217, issn: 1539-3755, 1550-2376. doi: 10.1103/PhysRevE.69.026113. [Online]. Available: http://arxiv.org/abs/cond-mat/0308217.

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

[23] A. Rezaei, S. M. Far, and M. Soleymani, "Near linear-time community detection in networks with hardly detectable community structure," ASONAM '15, pp. 65–72, 2015. doi: 10.1145/2808797.2808903. [Online]. Available: https://doi.org/10.1145/2808797.2808903.

[24] 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/.

[25] 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, arXiv: 1810.08473, issn: 2045-2322. doi: 10.1038/s41598-019-41695-z. [Online]. Available: http://arxiv.org/abs/1810.08473.

[26] 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, P10008–P10020, Oct. 2008, arXiv: 0803.0476 version: 2, issn: 1742-5468. doi: 10.1088/1742-5468/2008/10/P10008. [Online]. Available: http://arxiv.org/abs/0803.0476.

[27] P. Pons and M. Latapy, "Computing communities in large networks using random walks," ISCIS'05, pp. 284–293, 2005. doi: 10.1007/11569596_31. [Online]. Available: https://doi.org/10.1007/11569596_31.

[28] J. Reichardt and S. Bornholdt, "Statistical Mechanics of Community Detection," *Physical Review E*, vol. 74, no. 1, p. 016 110, Jul. 2006, arXiv: cond-mat/0603718, issn: 1539-3755, 1550-2376. doi: 10.1103/PhysRevE.74.016110. [Online]. Available: http://arxiv.org/abs/cond-mat/0603718.

 ISBN: 978-1-61208-994-2