A Case Study Comparing Twitter Communities Detected by the Louvain and Leiden Algorithms During the 2022 War in Ukraine

Karolina Sliwa
Vienna University of Economics and Business (WU)
Vienna, Austria
karolina.malgorzata.sliwa@wu.ac.at

Ema Kušen
University of Vienna, Faculty of Informatics
Vienna University of Economics and Business (WU)
Vienna, Austria
ema.kusen@univie.ac.at

Mark Strembeck
Vienna University of Economics and Business (WU)
Secure Business Austria (SBA)
Complexity Science Hub (CSH)
Vienna, Austria
mark.strembeck@wu.ac.at

ABSTRACT

This paper presents a case study regarding a comparative examination of the Louvain and Leiden community detection algorithms. The case study was conducted on a real-world communication network consisting of 3,222,623 nodes and 27,423,553 edges. In particular, the network in our case study models the communication between Twitter users during the initial four weeks of the 2022 war in Ukraine. In addition, we also applied dynamic topic modeling in order to examine differences in the detected communities.

CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in collaborative and social computing**; • **Information systems** → **Social networks**.

KEYWORDS

Community Detection, Louvain, Leiden, Topic Modeling, Twitter

ACM Reference Format:
Karolina Sliwa, Ema Kušen, and Mark Strembeck. 2024. A Case Study Comparing Twitter Communities Detected by the Louvain and Leiden Algorithms During the 2022 War in Ukraine. In *Companion Proceedings of the ACM Web Conference 2024 (WWW '24 Companion), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3589335.3651892

1 INTRODUCTION

Online social media platforms enable users to engage in diverse interactions, sharing information, opinions, common emotional sentiments, and fostering connections across the globe. Users with shared interests often form (virtual) online communities. In this context, community detection involves identifying sets of nodes within a network that share similar properties [29],[6],[23]. The community concept helps to identify the hidden structures of various networks [20].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
WWW '24 Companion, May 13–17, 2024, Singapore, Singapore.
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0172-6/24/05…$15.00
https://doi.org/10.1145/3589335.3651892

Identifying communities within complex networks is a challenging task. Since community detection is a non-deterministic polynomial time hard (NP-hard) problem (see, e.g., [2], [6], [19]), obtaining an optimal solution quickly becomes computationally more demanding with the growth of network size. The temporal aspect of community detection adds another layer of complexity, acknowledging that community membership may evolve over time [8],[31].

Another set of challenges arises in the comparison and selection of community detection algorithms, particularly in the context of large real-world networks. Moreover, in many real-world applications, the absence of a 'ground truth', which involves a distinct assignment of nodes to communities, may further complicate an evaluation of the reliability of community detection procedures. Also note that the majority of benchmarking techniques predominantly center on static networks [18]. When applied to real-world networks, these benchmarks typically deal with either undirected or small-scale networks [16],[17],[21], [28].

One of the most prominent algorithms in community detection is the Louvain algorithm [1]. Its effectiveness has been verified in the context of large-scale directed networks [4]. The Louvain algorithm optimizes modularity of a network, seeking to maximize the density of connections within communities while minimizing connections between them. When applying the Louvain algorithm, the number of candidate communities notably decreases after only a few iterations, concentrating the majority of the running time on the initial iteration [15].

Yet, a drawback of this algorithm is its tendency to generate large communities that consist of a significant portion of nodes. As indicated by [27], this even occurs in situations where smaller communities might be expected, potentially resulting in poorly connected or even disconnected communities. For these reasons, the Leiden algorithm [27] has been introduced which aims at improving the discovered partitions in comparision to the Louvain algorithm.

In this paper, we compare the results produced by the Louvain and Leiden algorithms on a directed weighted large-scale social network derived from a Twitter dataset. In particular, our focus is on analyzing communities that surfaced during the initial phase of the 2022 Ukraine War, spanning from February 24, 2022, to March 25, 2022. The paper extends our previous work [26] and is guided by two research questions:

RQ 1: *How do the Leiden and Louvain algorithms compare in effectively detecting Twitter communities?* Given the absence of a ground-truth in our real-world social network, benchmarking options are limited. However, we adhere to standard metrics such as modularity, daily recognition of communities, and execution time commonly employed in evaluating various community detection algorithms on real-world networks, as outlined in [28].

RQ 2: *Do topical similarities exist among the detected communities?* In this paper, we especially focus on finding topical similarities in the five largest communities occurring during the final five days of the observation period. We focused on the five largest communities due to constraints imposed by the hardware that was available at the time of writing. Our main objective is to understand how Louvain and Leiden communities differ with respect to topical content.

The remainder of this paper is organized as follows: Section **2** discusses some background information on the Leiden and Louvain algorithms. Section **3** delves into the specifics of the research approach. Section **4** presents the results of our community detection analysis, followed by a comprehensive discussion. We conclude with implications, future directions, and the broader significance of our work.

2 BACKGROUND

Before going into the details of both approaches, it is worth mentioning the modularity function, which serves as a foundation for both algorithms. Modularity serves as a metric assessing the effectiveness of dividing a graph into communities. Represented by a function Q with values ranging from [-1, 1], higher values indicate that the partition of communities is relevant for the network [20]. A common problem with modularity is the resolution limit which suggests that the modularity influenced by the network's edge count makes it difficult to identify small communities that might be merged into larger ones [7].

The Louvain algorithm optimizes modularity through two fundamental phases [1]: local moving of nodes and aggregation of the network. In the local moving phase, individual nodes shift to the community that maximizes the increase in the quality function. The aggregation phase involves creating an aggregate network based on the obtained partition, repeating these steps until further improvements in quality cannot be achieved. Despite its simplicity, the Louvain algorithm may generate arbitrarily badly connected communities, highlighting a crucial challenge [27].

In response to this limitation, the Leiden algorithm emerges as a refinement and improvement over the Louvain algorithm proposed by [27]. Partly based on the smart local move algorithm, the Leiden algorithm incorporates ideas such as speeding up the local moving of nodes and moving nodes to random neighbors. It comprises three phases: local moving of nodes, refinement of the partition, and aggregation of the network. Notably, the Leiden algorithm provides explicit guarantees, ensuring connected communities and convergence to locally optimally assigned subsets of all communities, especially when applied iteratively.

Evaluation of community detection algorithms takes different forms, depending on the availability of reference communities. In scenarios with known reference communities, such as in supervised settings, measures like centrality metrics (e.g., Betweenness Centrality, Closeness Centrality, and Degree Centrality) play a crucial role in determining node assignments and assessing their influence within the network [11]. The Fraction of Correctly Classified nodes (FCC), introduced by [10], evaluates the agreement between a node's estimated community and the majority of nodes in its reference community, providing insights into how well the algorithm's identified communities align with the ground truth.

Evaluations relying on networks with a predefined 'ground truth' have constrained advantages, as these networks do not accurately mirror the complex nature of real-world empirical networks. In particular, benchmark networks exhibit a relatively straightforward structure, while empirical networks showcase a more diverse and rich architecture. Unsupervised community detection lacks a predefined reference, while supervised evaluation allows for a direct comparison with known community structures. Unsupervised approaches focus on revealing inherent structures without predefined ground-truth. This demands a nuanced evaluation framework tailored to the nature of the analysis. Both approaches may contribute valuable insights into the performance and effectiveness of community detection algorithms [30].

The work by [27] has contributed significantly to examining the differences the Louvain and Leiden algorithms. Our case study will focus on assessing algorithmic performance based on the quality of the community structure.

3 RESEARCH PROCEDURE

Our procedures are structured in the following manner.

Phase 1: Data extraction. To extract the data we used Twitter's Search API with academic access and a list of hashtags and keywords that were selected after monitoring the discourse about the 2022 Ukraine war. For more details please refer to [14, 26]. The dataset analyzed in this paper covers the time from 24 February until 25 March 2022 and includes 189 million tweets in English language authored by 3.2 million Twitter users.

Phase 2: Network Modeling. To derive the @-mention network, we used the source (user's screen name sending the tweet with an @mention), target (user's screen name being mentioned), and timestamp of the tweets in our dataset. Our network is directed and weighted, with weights indicating message quantity between nodes. The @-mention network consists of 3,222,623 nodes and 27,423,553 edges (see also [14]). Given the noise in online social connections, particularly in large networks, backboning is recommended to identify and remove redundant or randomly formed connections [3]. As illustrated in [25], a redundant edge is characterized by its lack of statistically significant deviations from a null model in the local assignment of weights to edges. In our study, we opted to employ the publicly available implementation[1] to utilize the *Disparity Filter (DF)* technique (see [25]). The goal of backboning is to identify the statistically significant connections that form the backbone of the network, without disregarding small-scale interactions. As a result of this process, our network was reduced to 27,352,653 edges, representing a decrease of 70,900 edges.

Phase 3: Community Detection using the Louvain & Leiden Algorithms. In this phase, we use the Louvain and Leiden algorithms to discover the underlying community structures within our backbone network. The application of these algorithms will

[1]https://github.com/malcolmvr/backbone_network

result in the partitioning of nodes into distinct communities, allowing us to gain insight into the structural organization of the Twitter @-mention network.

Phase 4: Comparison of Algorithms with Evaluation Metrics. Once we have applied both the Louvain and Leiden algorithms to our backbone network, we will proceed to a comparative analysis of their performance. We will utilize a set of evaluation metrics to assess the quality of the community structures produced by each algorithm. These metrics include modularity, execution time, quantity of communities and membership size. The objective is to explore differences between algorithms in partitioning the network into coherent and distinguishable communities.

Phase 5: Dynamic Topic Modeling. In the final phase, our aim is to compare the largest communities identified by both algorithms, using them as a 'context metric' to examine differences within the same timeframe as detected by different algorithms. Specifically, when analyzing topical differences between the Leiden and Louvain communities, we concentrate on the five communities recognized in the final 5-day snapshot. We use BERTopic Dynamic Topic Modeling (DTM) in analyzing the evolution of topics over time.

4 RESULTS

4.1 Network Modeling

Metric	μ	std	min	max
Density	6.35E-06	3.04E-06	2.24E-06	1.69E-05
Reciprocity	0.034	0.006	0.019	0.047
Transitivity	0.022	0.007	0.011	0.038
Avg. CC	0.055	0.005	0.039	0.066
Avg. In-Deg.	2.115	0.212	1.707	2.666
Deg. Assort.	-0.068	0.015	-0.101	-0.041
#SCCs	11523.63	12425.78	17.00	43836.00
#WCCs	365064.60	192061.49	82190.00	821043.00

Table 1: Average metrics of the network.

We apply classical structural measures to our backbone network to identify fundamental topological characteristics. These measures include *Transitivity* and *Average Local Clustering Coefficient* which offer global and local perspectives on triadic closure. The *Degree Assortativity* assesses the propensity of nodes to connect with peers of similar degrees. Additionally, *Strongly* and *Weakly Connected Components* provide insights into the network's connectivity in terms of isolated subgraphs.

With a density of μ = 6.35E-06, the network has fewer connections relative to the total possible connections. The reciprocity value of μ = 0.034 suggests a moderate level of mutual connections, indicating that a portion of interactions involves bidirectional relationships. Transitivity, at μ = 0.022, highlights a limited tendency for interconnected nodes to form triadic relationships. The average clustering coefficient of μ = 0.055 reflects a moderate degree of local connectivity among neighboring nodes. The average in-degree is μ = 2.115. The negative degree assortativity (μ = -0.068) implies a preference for nodes to connect to those with different degrees, resulting in a disassortative network. In terms of connected components, the network comprises an average of μ = 11,523.63 strongly connected components (SCCs), indicating the presence of numerous internally cohesive subgroups. Additionally, the average number of weakly connected components (WCCs) is substantially higher at μ = 365,064.60, emphasizing the broader connectivity of the network.

In summary (see Table 1), the network is in a state of ongoing development, with a dispersed and less interconnected structure. The lower density, average degree, and transitivity, along with the presence of small disconnected components, suggest that the network is constantly evolving, particularly in terms of link formation among newly emerged users during the initial stages of the conflict. The low clustering coefficient further supports the observation of a network with limited local clustering.

4.2 Community Detection

Despite the observed variability in the previously analyzed backbone network, the exploration of community detection remains crucial. While the network may exhibit diverse and variable connected components, the detection of communities within these structures can uncover hidden patterns and clusters that might not be immediately apparent. The outcomes of the Leiden and Louvain community detection algorithms reveal significant differences in the community structures they identified within the same dataset. Specifically, the average number of communities detected by the Louvain algorithm was denoted as μ=75544.73, sd=38078.31 communities, whereas the Leiden algorithm identified a substantially smaller average number of communities, denoted as μ=22385.43, sd=11482.10 communities emerging per day. Figure 2 shows the daily community count.

The Leiden algorithm, known for its emphasis on identifying cohesive communities and overcoming the resolution limit, detected a maximum of 60,507 communities. In contrast, the Louvain algorithm, which is more sensitive to local connectivity variations, identified a significantly larger number, reaching a maximum of 180,571 communities. One interpretation is that Leiden's communities may be more tightly knit and densely packed with members, while the surplus of communities detected by Louvain might be characteristic of its ability to identify numerous, potentially sparsely connected, local structures.

We then further analyzed the membership characteristics of these communities (see Figure 1). The Leiden algorithm identified 60,507 unique communities, with the largest one comprising 118,731 members. During the initial day of the conflict, an average of 15.88 members sd=761.54 per community was observed. On the following day, there was an increase in the average community membership to 18.48 sd=791.01, indicating a growth in size over the 14-day period. Despite this upward trend, there was a decline in the overall number of communities. Throughout the entire analysis period, community sizes continued to expand, accompanied by a notable decrease in the number of communities. On the last day of the study, community membership remained relatively stable, only dropping to 16.18 members sd=284.66.

In contrast, the Louvain algorithm generated considerably smaller communities, consisting of an average of 5.32 members sd=76.35 on the first day and identifying 180,571 unique communities. The average size of Louvain communities remained relatively constant,

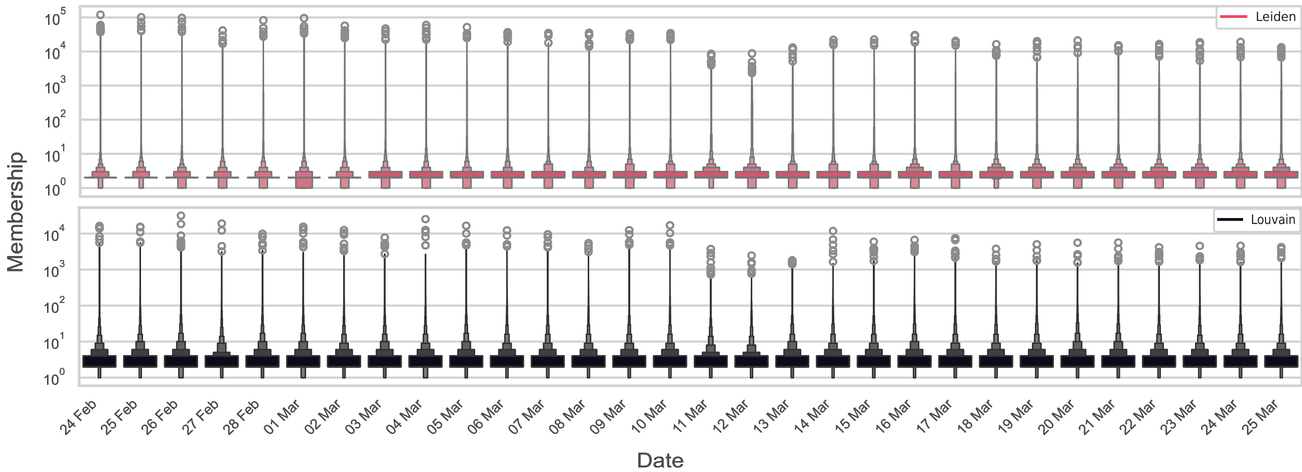

Figure 1: Daily Membership Size Comparison between Louvain and Leiden Communities.

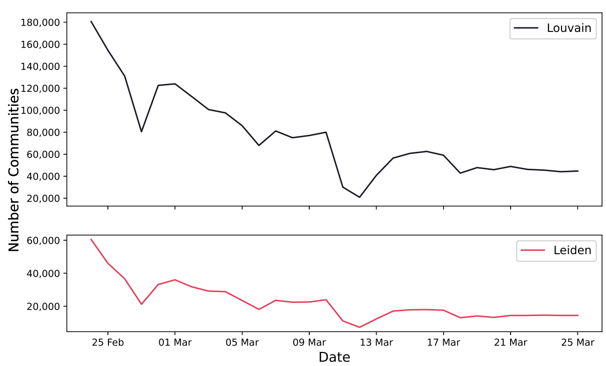

Figure 2: Daily Community Count Comparison between Louvain and Leiden Algorithms.

never falling below an average of 5 members, with less fluctuation compared to the Leiden communities.

4.3 Comparison of algorithms

Our following examination relies on modularity with a resolution parameter $\gamma = 1$. Figure 3 presents the highest modularity achieved using both the Louvain and Leiden algorithms for each network. On average, the modularity of Leiden communities surpasses that of Louvain, reaching μ=0.7453 compared to Louvain's μ=0.5942. This suggests that, on average, the community structures identified by Leiden exhibit a higher degree of internal cohesion and separation from the rest of the network, reinforcing the notion of Leiden's emphasis on more densely connected and distinct communities.

The execution time analysis underscores the considerable efficiency gap between the Leiden and Louvain algorithms, at least regarding the real-world network that we analyzed in our case study. On average, Leiden demonstrates significantly shorter execution times, averaging 22.52 seconds with low variability (sd = 14.63 seconds). In contrast, Louvain exhibits longer execution times, averaging 80.45 seconds, and a wider range of computational durations (sd = 50.21 seconds). These findings align with expectations, highlighting Leiden's efficiency and scalability compared to Louvain.

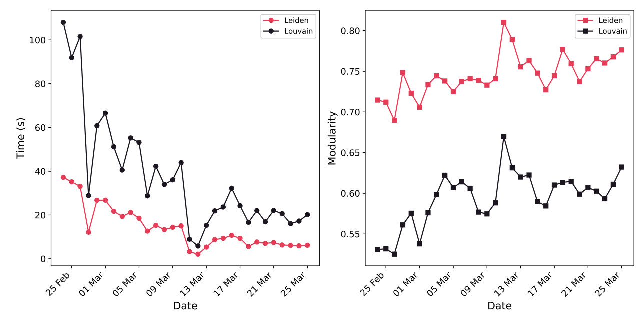

Figure 3: Comparison of Modularity and Execution Times between Leiden and Louvain Algorithms.

The conventional approach to community detection relies on the use of ground truth for result validation. However, research by [22] emphasizes the limitations of this method, particularly in real-world networks, where the inadequacy of ground truth validation becomes apparent. This limitation is evident in our experimental validation, where algorithms produce varying numbers of communities. Essentially, our findings indicate that the choice of community detection algorithms should be informed by both the characteristics of the dataset and the computational efficiency of the algorithms.

4.4 Dynamic Topic Modeling

Our analysis targets topic similarities in the five largest communities from the latest snapshot (5 days) detected by Leiden and Louvain algorithms. In our future work we aim at assessing if these smaller communities tend to stabilize, in contrast to the literature's indication of deviations in communities at the beginning [24].

To unveil latent topics characterizing users' tweets within these communities, we employed BERTopic Dynamic Topic Modeling (DTM). DTM allows us to track the evolution of topics over time, capturing shifts in how subjects like environmental awareness are discussed from one period to another. BERTopic method has been proven to be particularly effective in topic extraction, as demonstrated by [5]. The BERTopic modeling involves three key stages: transforming documents into a high-dimensional embedding, dimensionality reduction, and clustering of low-dimensional vector representations.

To make our model more efficient we experimented with various combinations of hyperparameters, leading to the adoption of the following configuration. The chosen representation model, *MaximalMarginalRelevance*, incorporates a diversity parameter of 0.5, underscoring its emphasis on selecting the most relevant and diverse documents for each topic. We selected the embedding model *all-mpnet-base-v2* maps sentences and paragraphs to a 768-dimensional dense vector space to enhance the overall representation of textual information. To decrease the computational time of model execution, we employed a *min_topic_size* of 175 documents and the activation of low memory mode. Notably, the calculation of probabilities for document-topic assignments was disabled, while verbose mode was enabled to provide detailed insights during the model fitting process.

In Figures 2 and 4, we explore the dynamic evolution of BERTopic topics within various communities for both Louvain and Leiden communities. In total, 355 topics were identified in Louvain communities, while Leiden had 424 topics. The disparity can be attributed to the smaller community sizes in Louvain and larger ones in Leiden. As depicted in Figure 4, Topic 1, with a frequency surpassing 15k, underwent a decline and eventually vanished within Leiden communities, raising questions about whether members merged with another community or simply disappeared. A temporal detection of communities could provide insights into this phenomenon. Other topics in Leiden communities appeared relatively stable within the considered timeframe.

Examining the Dynamic Topic Modeling (DTM) for Louvain communities, we observe significantly lower frequency for the communities detected by the Louvain algorithm. Additionally Topic 5 emerged only in the midday of March 21, featuring content praising Putin. This topic rose for a day before dropping significantly, prompting further exploration into whether this shift is linked to members joining or leaving the community. Understanding these dynamics contributes valuable information about community discourse evolution over time.

Delving into whether these alterations can be traced back to the presence of leaving members or the influence of new members joining the community adds an extra layer of complexity. This exploration could unravel patterns and dynamics that shape the community discourse, paving the way for a more nuanced comprehension of how communities evolve over time.

5 CONCLUSION

Our study compared the communities detected by the Louvain and Leiden algorithms in a large-scale social network of over 27 million edges and 3.2 million nodes. Leiden runs faster and consistently generated denser communities than Louvain. Additionally, our analysis delved into dynamic topic modeling, enhancing our comprehension of how selected topics evolve within specific communities. This exploration also pointed towards an intriguing future direction for topic-aware temporal dynamic community detection. This research fills a gap in understanding algorithm outcomes in real-world networks, emphasizing the importance of network structure and community formation in information exchange.

Moving forward, our research agenda will focus on temporal community detection building on the research of [9, 12, 13]. Recognizing the temporal dimension's significance, particularly in crisis scenarios, will contribute to a more nuanced understanding of community dynamics. Furthermore, our future work will extend into Dynamic Topic Modeling, enabling us to capture the evolving nature of topics within these dynamic communities over time. The incorporation of such models will provide a more comprehensive analysis of the evolving social network structures, allowing for a deeper understanding of the nuanced interactions and discussions within these online communities.

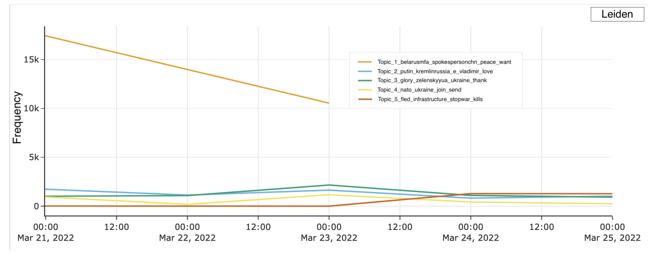

Figure 4: DTM for Leiden Communities.

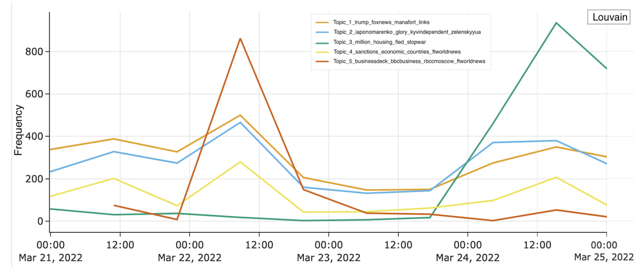

Figure 5: DTM for Louvain Communities.

REFERENCES

[1] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 2008, 10 (2008), P10008.
[2] Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Gorke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner. 2007. On modularity clustering. *IEEE transactions on knowledge and data engineering* 20, 2 (2007), 172–188.
[3] Lucio La Cava, Luca Maria Aiello, and Andrea Tagarelli. 2023. Drivers of social influence in the Twitter migration to Mastodon. *Scientific Reports* 13, 1 (2023), 21626.
[4] Nicolas Dugué and Anthony Perez. 2015. *Directed Louvain: maximizing modularity in directed networks.* Ph. D. Dissertation. Université d'Orléans.
[5] Roman Egger and Joanne Yu. 2022. A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts. *Frontiers in sociology* 7 (2022), 886498.

[6] Santo Fortunato. 2010. Community detection in graphs. *Physics reports* 486, 3-5 (2010), 75–174.

[7] Santo Fortunato and Marc Barthelemy. 2007. Resolution limit in community detection. *Proceedings of the national academy of sciences* 104, 1 (2007), 36–41.

[8] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. A survey on concept drift adaptation. *ACM computing surveys (CSUR)* 46, 4 (2014), 1–37.

[9] Xubo Gao, Qiusheng Zheng, Didier A Vega-Oliveros, Leandro Anghinoni, and Liang Zhao. 2020. Temporal network pattern identification by community modelling. *Scientific Reports* 10, 1 (2020), 240.

[10] Michelle Girvan and Mark EJ Newman. 2002. Community structure in social and biological networks. *Proceedings of the national academy of sciences* 99, 12 (2002), 7821–7826.

[11] Muhammad Aqib Javed, Muhammad Shahzad Younis, Siddique Latif, Junaid Qadir, and Adeel Baig. 2018. Community detection in networks: A multidisciplinary review. *Journal of Network and Computer Applications* 108 (2018), 87–111.

[12] Ema Kušen and Mark Strembeck. 2021. Building blocks of communication networks in times of crises: Emotion-exchange motifs. *Computers in Human Behavior* 123 (2021).

[13] Ema Kušen and Mark Strembeck. 2021. Emotional Communication During Crisis Events: Mining Structural OSN Patterns. *IEEE Internet Computing* 25, 02 (March 2021), 58–65.

[14] Ema Kušen and Mark Strembeck. 2023. The Effects of Multiple Exposure to Highly Emotional Social Media Content During the Early Stages of the 2022 War in Ukraine. *Springer Nature Computer Science* 4, 5 (August 2023).

[15] Andrea Lancichinetti and Santo Fortunato. 2009. Community detection algorithms: a comparative analysis. *Physical review E* 80, 5 (2009), 056117.

[16] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. 2008. Benchmark graphs for testing community detection algorithms. *Physical review E* 78, 4 (2008), 046110.

[17] Jure Leskovec, Kevin J Lang, and Michael Mahoney. 2010. Empirical comparison of algorithms for network community detection. In *Proceedings of the 19th international conference on World wide web*. 631–640.

[18] Pranav Nerurkar, Madhav Chandane, and Sunil Bhirud. 2019. A comparative analysis of community detection algorithms on social networks. In *Computational Intelligence: Theories, Applications and Future Directions-Volume I: ICCI-2017*. Springer, 287–298.

[19] Mark EJ Newman. 2006. Modularity and community structure in networks. *Proceedings of the national academy of sciences* 103, 23 (2006), 8577–8582.

[20] Mark EJ Newman and Michelle Girvan. 2004. Finding and evaluating community structure in networks. *Physical review E* 69, 2 (2004), 026113.

[21] Günce Keziban Orman, Vincent Labatut, and Hocine Cherifi. 2012. Comparative evaluation of community detection algorithms: a topological approach. *Journal of Statistical Mechanics: Theory and Experiment* 2012, 08 (2012), P08001.

[22] Leto Peel, Daniel B Larremore, and Aaron Clauset. 2017. The ground truth about metadata and community detection in networks. *Science advances* 3, 5 (2017), e1602548.

[23] Mason Alexander Porter, Jukka-Pekka Onnela, Peter J Mucha, et al. 2009. Communities in networks. (2009).

[24] Naw Safrin Sattar, Aydin Buluc, Khaled Z Ibrahim, and Shaikh Arifuzzaman. 2023. Exploring temporal community evolution: algorithmic approaches and parallel optimization for dynamic community detection. *Applied Network Science* 8, 1 (2023), 64.

[25] M Ángeles Serrano, Marián Boguná, and Alessandro Vespignani. 2009. Extracting the multiscale backbone of complex weighted networks. *Proceedings of the national academy of sciences* 106, 16 (2009), 6483–6488.

[26] Karolina Sliwa, Ema Kušen, and Mark Strembeck. 2023. An Analysis of Twitter Communities Related to the 2022 War in Ukraine. In *Proc. of the 8th International Conference on Complexity, Future Information Systems and Risk, (COMPLEXIS)*.

[27] Vincent A Traag, Ludo Waltman, and Nees Jan Van Eck. 2019. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific reports* 9, 1 (2019), 5233.

[28] Meng Wang, Chaokun Wang, Jeffrey Xu Yu, and Jun Zhang. 2015. Community detection in social networks: an in-depth benchmarking study with a procedure-oriented framework. *Proceedings of the VLDB Endowment* 8, 10 (2015), 998–1009.

[29] Bo Yang, Dayou Liu, and Jiming Liu. 2010. Discovering communities from social networks: Methodologies and applications. *Handbook of social network technologies and applications* (2010), 331–346.

[30] Zhao Yang, René Algesheimer, and Claudio J Tessone. 2016. A comparative analysis of community detection algorithms on artificial networks. *Scientific reports* 6, 1 (2016), 30750.

[31] Indrė Žliobaitė, Mykola Pechenizkiy, and Joao Gama. 2016. An overview of concept drift applications. *Big data analysis: new algorithms for a new society* (2016), 91–114.