Leiden Coloring Algorithm for Influencer Detection

Handrizal[1]*, Poltak Sihombing[2], Erna Budhiarti Nababan[3], Mohammad Andri Budiman[4]
Student Doctoral Program in Computer Science[1]
Department of Computer Science-Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia[1, 2, 4]
Department of Information Technology-Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia[3]

***Abstract*—In today's digital age, the role of influencers, especially on social media platforms, has grown significantly. A commonly used feature by business professionals today is follower grouping. However, this feature is limited to identifying influencers based solely on mutual followership, highlighting the need for a more sophisticated approach to influencer detection. This study proposes a novel method for influencer detection that integrates the Leiden coloring algorithm and Degree centrality. This approach leverages network analysis to identify patterns and relationships within large-scale datasets. Initially, the Leiden coloring algorithm is employed to partition the network into various communities, considered potential influencer hubs. Subsequently, Degree centrality is utilized to identify nodes with high connectivity, indicating influential individuals. The proposed method was validated using data crawled from Twitter (X) social media, employing the keyword "GarudaIndonesia." The data was collected using Tweet-Harvest between January 1, 2020, and October 16, 2024, resulting in a dataset of 22,623 rows. The dataset was subjected to two experimental scenarios: 1,000 and 5,000 rows. Compared to the Louvain coloring method, the proposed approach demonstrated an increase in the modularity value of the Leiden coloring algorithm by 0.0306, a reduction in time processing by 14.4848 seconds, and a decrease in the number of communities by 1,290.**

***Keywords*—Influencer; Louvain coloring; Leiden; Leiden coloring**

I. Introduction

The role of influencers, especially on social media platforms, has grown significantly, necessitating the development of more sophisticated influencer detection methods [1]. Influencer detection is a part of community detection. Businesses must accurately identify customers and respond to their needs to remain competitive [2]. As the business landscape evolves, businesses are increasingly adopting digital marketing strategies to keep pace with the competition [3].

Digital marketing involves promoting and disseminating information, as well as searching for markets, through digital media by utilizing various means such as social media [4]. Digital marketing simplifies the process for businesses to connect with and satisfy the desires of potential consumers [5]. From the perspective of potential consumers, digital marketing provides easy access to product information through cyberspace, making it faster and more convenient to search for information [6]. To increase their customer base, businesses need to identify individuals or groups who can help market their products or services. These individuals or groups, with the potential to attract more consumers, are known as influencers [7].

Contemporary digital marketing applications, particularly on Twitter (X) [8], provide features such as follower grouping to assist businesses in promoting their products.

However, the follower grouping feature is limited to identifying influencers based solely on mutual followership, without providing information about the topics discussed by the group or the size and influential members of the group. This can be challenging for business owners who are new to using social media as a promotional tool. Therefore, a method is needed to detect influencers based on specific topics or keywords using Social Network Analysis (SNA) on platforms like Twitter (X), represented graphically. One widely used SNA method is community detection.

There are several community detection algorithms, such as the Louvain algorithm [9] and the Leiden algorithm [10]. The latter was created to identify communities in large networks with complex modularity patterns. The Leiden algorithm is one of the approaches used in social network analysis [11], aiming to identify subgraph groups with strong internal connections. This can help understand the social structure and relationships within a social network, but the algorithm has limitations, such as difficulty in interpreting community visualizations due to the lack of color coding. Graph coloring can accelerate the process by assigning unique colors to nodes, similar to indexing in a relational database. Based on this, this study aims to influencer detection using the Leiden algorithm combined with graph coloring. By incorporating color coding into community detection, we can improve the interpretability and effectiveness of the results.

II. Materials and Methods

A. Leiden Algorithm

The Leiden algorithm is an algorithm used for community detection in networks or graphs. Designed to work efficiently, the Leiden algorithm detects communities in networks or graphs based on modularity optimization [13]. Modularity is a measure that describes the extent to which the density of connections within a community exceeds that of a random network. The modularity value is a scale value ranging from [-1,1]. The modularity of a network is calculated using the following formula:

*Corresponding Author. Email ID: handrizal@usu.ac.id

$$Q = \frac{1}{2m}\sum A_{i,j} - \frac{k_i k_j}{2m}\delta(c_i, c_j)$$

where:

- $A_{i,j}$ represents the edge weights between nodes i and j
- k_i and k_j are the sum of the weights of the edges attached to nodes i and j
- m is the sum of all edge weights in the graph
- c_i and c_j are the node communities
- δ is the Kronecker delta function
- $(\delta(c_i, c_j) = 1$ if $c_i = c_j$, otherwise 0

The Leiden algorithm comprises three primary stages:

1) Moving nodes: Iteratively moving nodes between communities to optimize the overall modularity of the network.

2) Refinement: Dividing the network into distinct, connected components by separating unconnected communities

3) Aggregation: Form a hierarchical network by iteratively aggregating nodes into communities and then treating each community as a single node in the next level of the hierarchy

Steps in the Leiden Algorithm:

1) Initialization: Create a representation of the graphunder analysis. This can be achieved using either an adjacency list or an adjacency matrix. Subsequently, each node within the graph is initialized as an independent community

2) Iteration: Repeat the following steps iteratively until convergence is attained:

a) Local Move Step

- Assess the feasibility of relocating each node within the graph to a different community.
- Calculate the change in modularity resulting from the movement of the node
- If a community exists that provides a greater increase in modularity, relocate the node accordingly.

b) Community Aggregation

- Once all nodes have been evaluated, nodes within the same community are merged into a single new node
- Construct a new graph in which each node corresponds to a community generated in the preceding step

c) Weight Update

- Recompute the edge weights in the new graph according to the number of nodes represented in each community.

3) Termination: The iteration process continues until a stable state is reached, where neither community assignments nor modularity values change significantly.

B. Graph Coloring

Graph coloring, a technique introduced in 1979 [14], involves assigning colors to vertices in a graph such that no adjacent vertices share the same color. This study explores the application of graph coloring [12] within the context of network analysis. Consider a graph G(V, E), where V represents the set of vertices and E represents the set of edges in the graph. We aim to assign a color, w(i) ∈ {1, 2, ..., k}, to each vertex i ∈ V, where k is the number of colors used, such that:

$$\forall (i,j) \in G, w(i) \neq w(j)$$

C. Leiden Coloring Algorithm

In 2024, the study in [15] stated that one of the shortcomings of the Leiden algorithm is its inability to efficiently process large networks, resulting in relatively long processing times.To address this issue, the Leiden algorithm was modified using graph coloring, hereinafter referred to as Leiden Coloring.

The steps of the Leiden coloring algorithm are as follows:

- Phase 1: Community Detection

Community detection within the graph is performed using the Leiden algorithm. The implementation strictly adheres to the algorithmic steps and mathematical formulations definedby the Leiden algorithm.

- Phase 2: Community Coloring

After community identification, the graph coloring principle is applied to the resulting community structure. Each community is assigned a unique color, ensuring that no more colors are used than the number of identified communities.

This process effectively integrates the Leiden algorithm with graph coloring, resulting in each node being labeled with both its community membership and a unique color.

The stages of influencer detection research using Leiden coloring are as follows:

- Business understanding

This stage involves the collection of information about influencers, encompassing indicators that can identify potential influencers, relevant phenomena, and pertinent facts.

- Twitter data collection

Data collection for this study was conducted on Twitter (X) using the Tweet-Harvest service. Tweet-Harvest was employed to retrieve tweets related to the keyword "GarudaIndonesia" from January 1, 2020, to October 16, 2024. This process resulted in a dataset of 22,623 tweets.

- Network construction

This stage of dataset processing encompasses data selection based on research objectives, subsequent data cleaning to eliminate irrelevant entries, and finally, data transformation into a graph format for further analysis.

- Community detection

After the preceding stage, the dataset will undergo further analysis for influencer detection utilizing Social Network

Analysis methodologies. Both the Leiden and Leiden Coloring algorithms will be employed in this phase.

- Analysis results

This stage involves analyzing, concluding, and evaluating the results of influencer detection from the previous step. The information generated may include graph visualizations, the number of communities formed, and the identification of influencers within each community.

- Evaluation

At this stage, the algorithm's performance is evaluated using three matrices: modularity, time processing, and the number of communities to determine the algorithm's overall effectiveness.

III. Results and Discussion

This study employed the Leiden coloring algorithm for influencer detection, conducting tests with two datasets: a smaller dataset of 1,000 rows and a larger dataset of 5,000 rows. The objective was to analyze how the identification of influential individuals within a social network might be affected by the size and scale of the available data.

A. Detection of Influencers using the Leiden Coloring Algorithm

This section presents the results obtained from the proposed Leiden coloring algorithm. These results encompass the influencer detection matrix, modularity values, processing times, and the number of identified communities. Two dataset testing scenarios were conducted in this study. The first scenario utilized a dataset comprising 1,000 rows, while the second scenario employed a dataset consisting of 5,000 rows.

1) Results of influencer detection with dataset 1000 rows: This section presents the results obtained from the proposed Leiden coloring algorithm. These results are presented as an influencer detection matrix using a dataset of 1,000 rows.

TABLE I. Results of Influencer Detection with Dataset 1000 Rows

No.	Leiden Coloring Algorithm
1	IndonesiaGaruda
2	GarudaCares
3	wandiseptian11
4	PinterPoin
5	idbcpr

Table I shows that in the dataset containing 1000 rows, the username 'IndonesiaGaruda' holds the top influencer position, while 'idbcpr' ranks fifth.

2) Result of influencer detection with dataset 5000 rows: This section presents the results obtained from the proposed Leiden coloring algorithm. These results are presented as an influencer detection matrix using a dataset of 5,000 rows.

TABLE II. Result of Influencer Detection with Dataset 5000 Rows

No.	Leiden Coloring Algorithm
1	IndonesiaGaruda
2	disemuacom
3	GarudaCares
4	astuceclover
5	TiketPesawatPro

According to Table II, in the 5000-row dataset, the username 'IndonesiaGaruda' is identified as the top influencer, while 'TiketPesawatPro' is ranked fifth in terms of influencer.

B. Result of Matrix Modularity

This section presents the results obtained from the proposed Leiden coloring algorithm. The results are shown as a modularity matrix using datasets containing 1,000 rows and 5,000 rows.

TABLE III. Results of Matrix Modularity

No.	Dataset	Modularity of Leiden Coloring
1	1000 rows	0.9396
2	5000 rows	0.9381
Average		0.9388

Table III shows the modularity values for the 1000-row dataset (0.9396) and the 5000-row dataset (0.9381). Additionally, the matrix above is presented graphically in Fig. 1.

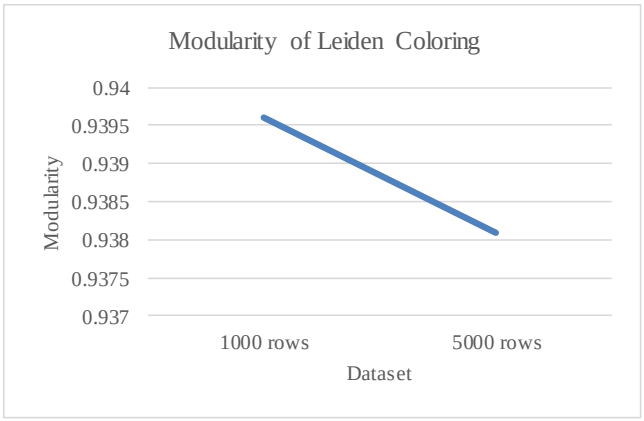

Fig. 1. Results of matrix modularity.

C. Result of Matrix Time Processing

This section presents the results obtained from the proposed Leiden coloring algorithm. The results are shown as a time-processing matrix using datasets containing 1,000 rows and 5,000 rows.

TABLE IV. Result of Matrix Time Processing

No.	Dataset	Time Processing Leiden Coloring (second)
1	1000 rows	29.5493
2	5000 rows	434.1838
Average		231,8666

In Table IV, it can be seen that for a dataset with 1000 rows, the processing time is 29.5491 seconds, and for a dataset with 5000 rows, it is 434.1838 seconds. The matrix above is also presented in graphical form, as shown in Fig. 2.

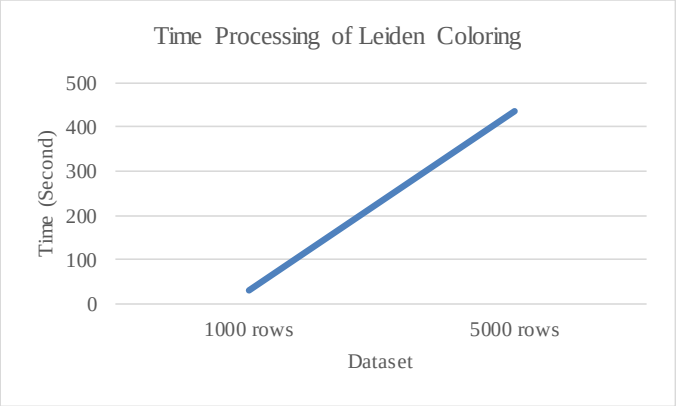

Fig. 2. Results of matrix time processing.

D. Results of Matrix Number Communities

This section presents the results obtained from the proposed Leiden coloring algorithm. The results are shown as a number of community matrix using datasets containing 1,000 rows and 5,000 rows.

TABLE V. RESULT OF MATRIX NUMBER OF COMMUNITIES

No.	Dataset	Number Communities Leiden Coloring
1	1000 rows	505
2	5000 rows	1969
Average		1237

As shown in Table V, the number of communities for the 1000-row dataset is 505, whereas for the 5000-row dataset, it is 1969. The corresponding matrix is graphically depicted in Fig. 3.

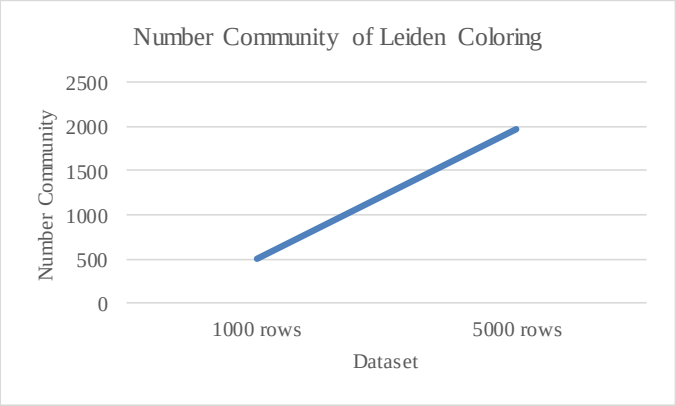

Fig. 3. Results of matrix number communities.

In the graph in Fig. 3 above, it can be seen that the number of communities for the 1,000-row dataset is 505, and it increases to 1,969 for the 5,000-row dataset.

E. Comparison of Louvain Coloring and Leiden Coloring Algorithm

This section presents the results of a comparison between the proposed Leiden coloring algorithm and the Louvain coloring algorithm, focusing on influencer detection, modularity value, time processing, and the number of identified communities. Two testing scenarios were conducted with varying dataset sizes: the first using a dataset with 1,000 rows, and the second using a dataset with 5,000 rows.

1) Comparison of influencer detection with dataset 1000 rows: This section presents a comparison of the results obtained using the proposed Leiden coloring algorithm with the Louvain coloring algorithm based on the influencer detection metric, using datasets containing 1,000 rows.

TABLE VI. COMPARISON OF INFLUENCER DETECTION WITH DATASET 1000 ROWS

Louvain Coloring Algorithm	Leiden Coloring Algorithm
IndonesiaGaruda	IndonesiaGaruda
GarudaCares	GarudaCares
astuceclover	wandiseptian11
TiketPesawatPro	PinterPoin
disemuacom	idbcpr

Table VI shows that the Louvain and Leiden coloring algorithms produce identical influencer detections for the first and second ranks in the 1000-row dataset: 'IndonesiaGaruda' and 'GarudaCares', respectively. However, the rankings diverge starting from the third rank.

2) Comparison of influencer detection with dataset 5000 rows: This section presents a comparison of the results obtained using the proposed Leiden coloring algorithm with the Louvain coloring algorithm based on the influencer detection metric, using datasets containing 5,000 rows.

TABLE VII. COMPARISON OF INFLUENCER DETECTION WITH DATASET 5000 ROWS

Louvain Coloring Algorithm	Leiden Coloring Algorithm
IndonesiaGaruda	IndonesiaGaruda
GarudaCares	disemuacom
TiketPesawatPro	GarudaCares
disemuacom	astuceclover
astuceclover	TiketPesawatPro

In Table VII, it can be seen that the Louvain coloring and Leiden coloring algorithms on the 5000-row dataset produce the same influencer detection for the first rank, namely the username 'IndonesiaGaruda.' Meanwhile, the second to fifth ranks produce different influencer detections.

Analysis of Tables VI and VII reveals that both the Louvain and Leiden coloring algorithms identify 'IndonesiaGaruda' as the top-ranked influencer. However, discrepancies in influencer rankings emerge from the second to the fifth positions.

F. Comparison of Matrix Modularity

This section presents a comparison of the results obtained using the proposed Leiden coloring algorithm with the Louvain coloring algorithm based on the modularity metric, using datasets containing 1,000 rows and 5,000 rows (Table VIII).

TABLE VIII. COMPARISON OF MATRIX MODULARITY

Dataset	Modularity	
	Louvain Coloring	Leiden Coloring
1000 rows	0.9114	0.9396
5000 rows	0.9050	0.9381
Average	0.9082	0.9388

The Leiden coloring algorithm consistently demonstrated higher modularity values compared to the Louvain coloring algorithm across all two test scenarios. The modularity values for the Leiden coloring algorithm ranged from a minimum of 0.9367 to a maximum of 0.9396, with an average of 0.9388. In contrast, the Louvain coloring algorithm exhibited a lower range, with a minimum of 0.9050 a maximum of 0.9252, and an average of 0.9082. This translates to an average increase in modularity of 0.0306 for the Leiden coloring algorithm. A comparative graph illustrating this modularity matrix is presented in Fig. 4.

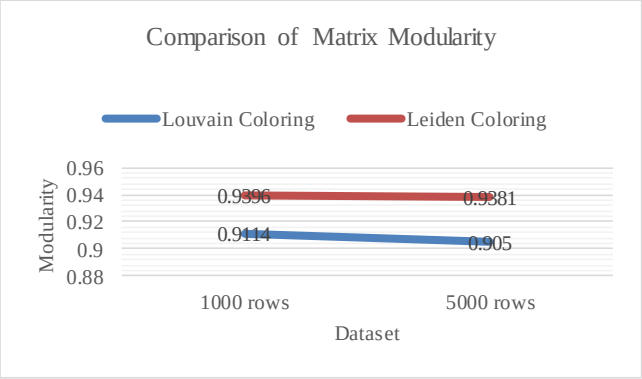

Fig. 4. Comparison of matrix modularity.

G. Comparison of Matrix Time Processing

This section presents a comparison of the results obtained using the proposed Leiden coloring algorithm with the Louvain coloring algorithm based on the time processing metric, using datasets containing 1,000 rows and 5,000 rows (Table IX).

TABLE IX. COMPARISON OF MATRIX TIME PROCESSING

Dataset	Time Processing (second)	
	Louvain Coloring	Leiden Coloring
1000 rows	41.85	29.5493
5000 rows	450.86	434.1838
Average	246,355	231,8666

The time processing of the Leiden coloring algorithm is better than that of the Louvain coloring algorithm. In the two test scenarios conducted, the Leiden coloring algorithm outperforms the Louvain coloring algorithm in all scenarios. The processing time for the Leiden coloring algorithm ranges from a minimum of 29.5493 seconds to a maximum of 434.1838 seconds, with an average of 246.355 seconds. Meanwhile, the Louvain coloring algorithm has a minimum value of 41.85 seconds, a maximum of 450.86 seconds, and an average of 231.8666 seconds. Thus, the Leiden coloring algorithm shows a reduction in processing time of 14.4848 seconds. The comparison of these processing times is also illustrated in the graph in Fig. 5.

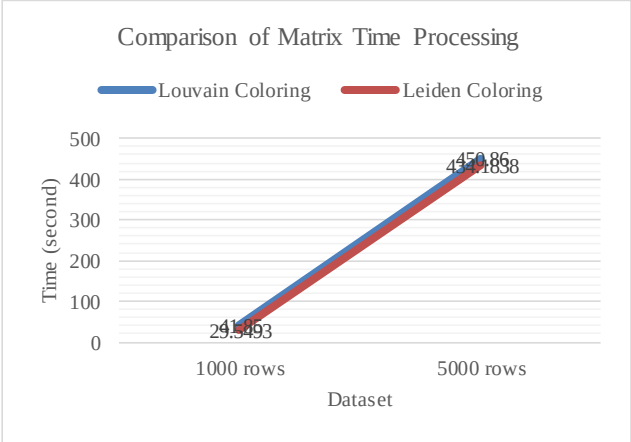

Fig. 5. Comparison of matrix time processing.

H. Comparison of Matrix Number Communities

This section presents a comparison of the results obtained using the proposed Leiden coloring algorithm with the Louvain coloring algorithm based on the number of communities metric, using datasets containing 1,000 rows and 5,000 rows (Table X).

TABLE X. COMPARISON OF MATRIX NUMBER COMMUNITIES

Dataset	Number Communities	
	Louvain Coloring	Leiden Coloring
1000 rows	936	505
5000 rows	4119	1969
Average	2527	1237

The Leiden coloring algorithm produced a significantly lower number of communities compared to the Louvain coloring algorithm across the two test scenarios. The number of communities detected by the Leiden algorithm ranged from 505 to 1969 with an average of 1237, while the Louvain coloring algorithm produced a range of 936 to 4119 with an average of 2527. This resulted in a reduction of 1290 communities on average. A comparison graph of the community numbers for both algorithms is presented in Fig. 6.

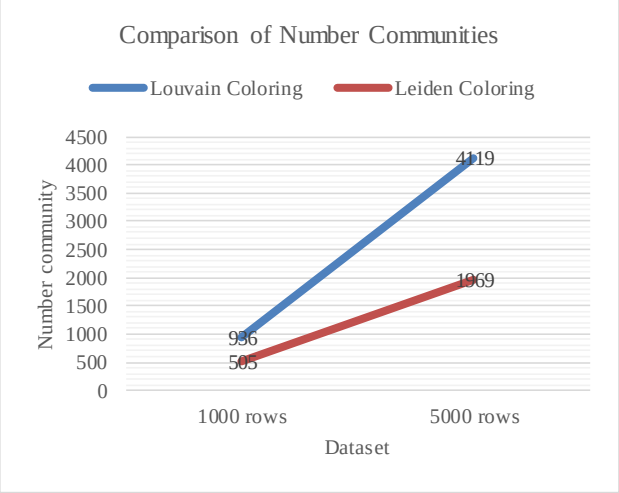

Fig. 6. Comparison of matrix number communities.

IV. Conclusion

This study successfully applied the Leiden coloring algorithm to identify influencers on Twitter (X). Influencer detection was conducted using the Leiden coloring algorithm on a dataset of Twitter (X) interactions related to "GarudaIndonesia." Data was collected using Tweet-Harvest between January 1, 2020, and October 16, 2024, resulting in a dataset of 22,623 tweets.

The dataset was evaluated using two experimental scenarios: one with 1,000 tweets and another with 5,000 tweets. The analysis identified five influential usernames in each scenario. For the 1,000-tweet dataset, the identified influencers were IndonesiaGaruda, GarudaCares, Wandiseptian11, PinterPoin, and idbcpr. For the 5,000-tweet dataset, the identified influencers were IndonesiaGaruda, disemuacom, GarudaCares, astuceclover, and TiketPesawatPro.

Compared to the Louvain coloring algorithm, the Leiden coloring algorithm demonstrated improved performance. The Leiden coloring algorithm exhibited a 0.0306 increase in modularity value, a 14.4848-second reduction in processing time, and a 1290-community reduction.

The primary contributions of this study include improved modularity, reduced processing time, and a more concise community structure when using the Leiden coloring algorithm for influencer detection.

Several limitations and suggestions have emerged during this study, such as the rapid development of Twitter (X) data due to bold processing methods, highlighting the need for new approaches to detect influencers. Therefore, future research could focus on applying the Leiden coloring algorithm to real-time Twitter (X) data.

References

[1] A. Rababah, L. Al-Haddad, M. S. Sial, Z. Chunmei, and J. Cherian, "Analyzing the effects of COVID-19 pandemic on the financial performance of Chinese listed companies," J. Public Aff., vol. 20, no. 4, 2020, doi: 10.1002/pa.2440.

[2] D. Ushakov, E. Dudukalov, E. Kozlova, and K. Shatila, "The Internet of Things impact on smart public transportation," Transp. Res. Procedia, vol. 63, pp. 2392–2400, 2022, doi: 10.1016/j.trpro.2022.06.275.

[3] Y. J. Purnomo, "Digital Marketing Strategy to Increase Sales Conversion on E-commerce Platforms," J. Contemp. Adm. Manag., vol. 1, no. 2, pp. 54–62, Aug. 2023, doi: 10.61100/ADMAN.V1I2.23.

[4] M. T. Khanom, "Using social media marketing in the digital era: A necessity or a choice," Int. J. Res. Bus. Soc. Sci. (2147- 4478), vol. 12, no. 3, pp. 88–98, 2023, doi: 10.20525/ijrbs.v12i3.2507.

[5] M. K. Peter and M. Dalla Vecchia, The Digital Marketing Toolkit: A Literature Review for the Identification of Digital Marketing Channels and Platforms, vol. 294, no. March. Springer International Publishing, 2021. doi: 10.1007/978-3-030-48332-6_17.

[6] D. R. Piranda, D. Z. Sinaga, and E. E. Putri, "ONLINE MARKETING STRATEGY IN FACEBOOK MARKETPLACE AS A DIGITAL MARKETING TOOL," J. Humanit. Soc. Sci. Bus., vol. 1, no. 3, pp. 1–8, Mar. 2022, doi: 10.55047/JHSSB.V1I2.123.

[7] D. Vrontis, A. Makrides, M. Christofi, and A. Thrassou, "Social media influencer marketing: A systematic review, integrative framework and future research agenda," Int. J. Consum. Stud., vol. 45, no. 4, pp. 617–644, Jul. 2021, doi: 10.1111/IJCS.12647.

[8] S. S. Veleva and A. I. Tsvetanova, "Characteristics of the digital marketing advantages and disadvantages," IOP Conf. Ser. Mater. Sci. Eng., vol. 940, no. 1, 2020, doi: 10.1088/1757-899X/940/1/012065.

[9] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," J. Stat. Mech. Theory Exp., vol. 2008, no. 10, Mar. 2008, doi: 10.1088/1742-5468/2008/10/p10008.

[10] V. Traag, L. Waltman, and N. J. van Eck, "From Louvain to Leiden: guaranteeing well-connected communities," Sci. Rep., vol. 9, no. 1, Oct. 2018, doi: 10.1038/s41598-019-41695-z.

[11] F. Nguyen, "Leiden-Based Parallel Community Detection," no. September, 2021, [Online]. Available: www.kit.edu

[12] Ü. V. Çatalyürek, J. Feo, A. H. Gebremedhin, M. Halappanavar, and A. Pothen, "Graph coloring algorithms for multi-core and massively multithreaded architectures," Parallel Comput., vol. 38, no. 10–11, pp. 576–594, Oct. 2012, doi: 10.1016/J.PARCO.2012.07.001.

[13] S. H. H. Anuar et al., "Comparison between Louvain and Leiden Algorithm for Network Structure: A Review," J. Phys. Conf. Ser., vol. 2129, no. 1, p. 012028, Dec. 2021, doi: 10.1088/1742-6596/2129/1/012028.

[14] D. Brélaz, "New methods to color the vertices of a graph," Commun. ACM, vol. 22, no. 4, pp. 251–256, Apr. 1979, doi: 10.1145/359094.359101.

[15] S. Sahu, K. Kothapalli, and D. S. Banerjee, "Fast Leiden Algorithm for Community Detection in Shared Memory Setting," ACM Int. Conf. Proceeding Ser., pp. 11–20, 2024, doi: 10.1145/3673038.3673146.