e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i2.29143

Analysis of X and Threads Responses Based on Single Keywords Using Graph Neural Network

Rifky Fahriza Sinaga[1), Rizal[2), Said Fadlan Anshari[3)
[1,2,3) Program Studi Teknik Informatika, Fakultas Teknik, Universitas Malikussaleh
Jl.Kampus Bukit Indah, Blang Pulo, Kota Lhokseumawe, Aceh 24355, Indonesia
E-Mail : rifky.220170134@mhs.unimal.ac.id [1); rizal@unimal.ac.id [2); saidfadlan@unimal.ac.id [3);

ABSTRACT

This study analyzes user responses on the social media platforms X and Threads regarding radicalism, based on the single keyword "radicalism." Differing interaction characteristics between the two platforms motivate a comparison of network structure using a graph-based approach and Graph Neural Networks (GNN). Data were collected through scraping of public content on X and Threads, with 5,000 raw posts each, yielding 1,327 reply interactions on X and 1,273 on Threads. Research stages included data collection, preprocessing, construction of the user-post graph, network metric analysis, and implementation of a Graph Autoencoder with a Graph Convolutional Network (GCN) encoder to generate node embeddings. The resulting graphs comprised 1,471 nodes and 1,223 unique edges for X, and 1,491 nodes and 1,081 unique edges for Threads, with X showing a denser structure (density 0.000565; average degree 1.663) than Threads (density 0.000487; average degree 1.450), while Threads was more fragmented (412 weak components versus 299 on X). The Graph Autoencoder was evaluated via link prediction using AUC and Average Precision (AP): X achieved AUC 0.5397 and AP 0.5879, slightly above the random-guessing baseline, while Threads achieved AUC 0.4987 and AP 0.5507, indicating a structure harder to reconstruct due to fragmentation. These quantitative results reinforce the network-metric findings that X forms a more connected network while Threads fragments into smaller groups. Practically, the findings offer an empirical basis for a decision-support system monitoring radicalism-related discourse, favoring dominant-cluster monitoring on X and parallel, cross-cluster monitoring on Threads. This study does not aim to detect or label accounts or content as radical, but to analyze interaction patterns and network characteristics of user responses.

Keywords – X, Threads, Radicalism, Social Network Analysis, Graph Neural Network

1. INTRODUCTION

The development of information and communication technology has given rise to a digital ecosystem that has transformed the way society interacts and consumes information. Social media has become a public space where social issues are formed, debated, and disseminated massively, driving a shift from one-way communication toward a participatory model in which every user simultaneously acts as both consumer and producer of information (Zhafira, 2024). Among various platforms, X (Twitter) and Threads are two text-based conversational platforms widely used for real-time public discussion. X (Twitter) is known as a microblogging platform for the open exchange of opinions and news, while Threads, launched by Meta in July 2023, recorded the fastest user growth at its launch despite experiencing a decline in daily active users following the initial euphoria (Jayaningsih et al., 2024). The interaction characteristics of the two platforms also differ: X is more effective at rapidly disseminating information despite being less verified, whereas Threads tends to encourage deeper and more structured discussion (Jayaningsih et al., 2024), making a comparison between the two relevant to examine in the context of sensitive and reactive issues.

One issue that consistently triggers debate on social media is radicalism. Data from the Ministry of Communication and Information Technology recorded the shutdown of 470 sites and accounts indicated to be spreading radical propaganda throughout 2022, reflecting the seriousness of this threat within Indonesia's digital space. Social media can be exploited to spread radical ideology across various platforms, and a social media analytics approach can help in understanding its dissemination patterns (Adek et al., 2021). This condition is further exacerbated by social media algorithms that reinforce users' exposure to radical content, as demonstrated by a systematic review of 82 studies (Akram & Nasar, 2023). In Indonesia, the detection of radicalism-related speech in Indonesian-language tweets using Convolutional Neural Networks (CNN) has achieved an accuracy of 87% (Putra & Sibaroni, 2022), proving that computational approaches are highly relevant to develop further.

Previous research on the spread of issues on social media has generally been limited to content analysis or aggregate interaction metrics, even though user responses form complex networks of relationships between accounts. Social Network Analysis (SNA) has emerged as a method for representing accounts as nodes and interactions as edges. Research analyzing the spread of the Natuna issue on Twitter using SNA successfully mapped 71,477 nodes and 147,066 edges while identifying key actors (Thayyibi & Mansur, 2021), while SNA

DOI : 10.30872/jim.v21i2.29143 e-ISSN 2597-4963 dan p-ISSN 1858-4853

has also proven capable of analyzing relationships between individuals in GSM-based communication networks (Fadlan & Ramdani, 2022), confirming SNA as a relevant methodological foundation before being further developed using Graph Neural Networks. Recent research has begun integrating Graph Neural Networks (GNN) to analyze information dissemination networks in greater depth, as GNN is capable of capturing complex relational patterns between nodes along with their attributes (Wu et al., 2021). This capability has been demonstrated in the identification of opinion leaders within information dissemination networks (Jain et al., 2023), graph-structure-based fake news detection (Phan et al., 2023), and the simultaneous capture of textual and relational patterns in fake news detection (Patel & Sutrakar, 2025).

2. Literature Review

A. *X* (Twitter) & *Threads*

X (Twitter) is a microblogging platform that enables the rapid dissemination of information through reply, repost, and mention features, making it more effective than Threads in spreading information (Jayaningsih et al., 2024). X's interaction data is also rich for network analysis, as every tweet, reply, mention, and retweet represents a relationship between accounts that can be mapped as a network structure, as demonstrated by Efendi et al. (2023) in their analysis of the #Pilpres2024 hashtag communication network, which successfully identified public communication patterns. Meanwhile, Threads is a text-based platform developed by Meta, launched in July 2023, and integrated with Instagram to support public conversation. Jayaningsih et al. (2024) noted that X tends to be used for the rapid and widespread dissemination of information, while Threads is more oriented toward deeper discussion, making a comparison between the two relevant for examining user responses to sensitive social issues such as radicalism. This study focuses on representing user interactions on both platforms as networks in order to analyze the patterns of dissemination of issues related to radicalism through graph-based analysis.

B. Radicalism in Digital Space

Radicalism in digital space can be understood as the dissemination of extreme ideologies, narratives, or calls to action through digital media, particularly social media. Social media is exploited by radical and extremist groups to spread ideology, build online communities, and expand communication reach, while also serving as a critical space in the radicalization process because it supports the rapid and widespread dissemination of extremist narratives (Akram & Nasar, 2023). In the Indonesian context, deep learning approaches have been applied to detect radical rhetoric in Indonesian-language tweets, demonstrating that computational approaches to radicalism remain highly relevant for further development (Adek et al., 2021).

Research in Indonesia has also moved toward machine learning- and deep learning-based user behavior analysis to classify levels of understanding of radicalism based on digital data, with results showing high classification performance (Lindawati et al., 2026). From a security standpoint, the role of intelligence in early detection of potential radicalism threats in digital space has become an important concern for preventing further spread (Roffi, 2023), while the optimization of AI-based cyber monitoring systems has also been recommended as a public information governance strategy to curb the spread of extremist propaganda on social media (Surono1 et al., 2026).

C. *Social Network Analysis* (SNA)

Social Network Analysis (SNA) is an analytical approach for mapping and measuring relationships and information flows between entities within a social network, in which entities are represented as nodes and the interactions between them as edges, thereby enabling researchers to identify key actors, detect relational structures, and map patterns of information dissemination within the network (Thayyibi & Mansur, 2021). SNA has been widely applied to understand relational patterns among actors; Thayyibi and Mansur (2021), for example, applied SNA to analyze the spread of the Natuna issue on Twitter, successfully mapping the network structure of nodes and edges while identifying influential actors within it. Fadlan and Ramdani (2022) further demonstrated that SNA can be used to analyze GSM-based communication networks in a criminal context, emphasizing that network analysis considers not only the number of connections but also the position and role of actors within the structure. Beyond identifying individual actors, SNA also offers various centrality measures, such as degree, betweenness, and closeness centrality, which allow researchers to assess an actor's influence from different perspectives, whether based on direct connections, control over information flow, or proximity to other nodes in the network. These findings support the use of SNA as a foundation for constructing the user interaction graph in this study.

D. *Graph Neural Network* (GNN)

Graph Neural Network (GNN) is a class of deep learning models designed to process graph-structured data, in which relationships between entities (nodes and edges) are learned through a message-passing mechanism that captures dependencies within the graph (Phan et al., 2023). Structurally, GNNs work by aggregating information from neighboring nodes in a layered manner, so that each node can represent both its own attributes and the structural context of the surrounding network; this layered aggregation allows the model to progressively capture higher-order relationships, from local patterns to global patterns within the graph (Wu et al., 2021). In social

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i2.29143

media analysis, GNNs can identify opinion leaders based on graph structure and node features, and are widely applied in fake news detection due to their ability to leverage relationships between nodes and information propagation patterns (Phan et al., 2023).

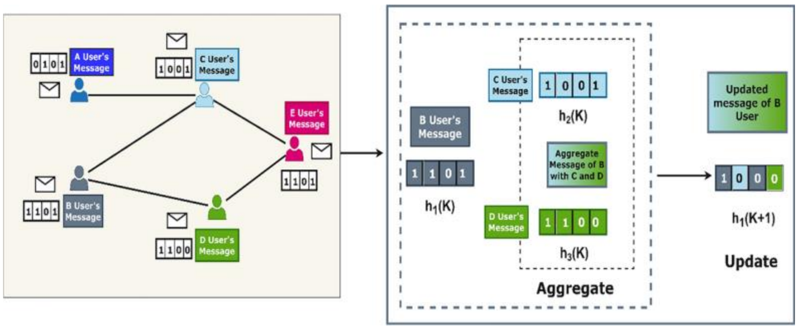


Figure 1. Message passing in a Graph Neural Network (GNN)

The process of updating node representations by combining messages from neighboring nodes in a layered manner at each layer forms the basis of this study in examining the structure of the user interaction network on X and Threads, in which user accounts are represented as nodes and interactions such as replies, reposts, retweets, and mentions are represented as edges, so that the resulting node representations reflect the position and characteristics of accounts within the network. Architecturally, this study refers to the fundamental concept of Graph Convolutional Network (GCN), which operates through convolution operations on graph structures; GraphSAGE, which applies sampling and aggregation of neighboring node features for large-scale graphs; Graph Attention Network (GAT), which applies an attention mechanism to assign different weights to each neighboring node; and Graph Autoencoder, an unsupervised approach that learns latent node representations through encoding-decoding, comprehensively discussed by Khemani et al. (2024) as the main theoretical foundation for selecting the GNN approach. The application of graph-based approaches to Indonesian-language social media text data has also been proven effective by Mukti, Adek, and Agusniar (2026) through the Multi-View Graph Attention Network (MV-GAT) model, which combines a semantic content graph, a user interaction graph, and a temporal propagation graph for political topic classification on Twitter/X data, further strengthening the justification for using a similar graph-based approach in this study.

3. Research Method

This research was conducted at the Computer Engineering Study Program, Faculty of Engineering, Universitas Malikussaleh, Lhokseumawe, with data collection and processing carried out online using the researcher's personal device, from February 2026 until completion. The hardware used was the researcher's Lenovo 82KT laptop (AMD Ryzen 3 5300U, 8 GB RAM, Windows 11 Pro 64-bit), which was used throughout the entire process, from data collection to the evaluation of analysis results. The software used included Python 3.X, Apify for scraping data from X and Threads, Visual Studio Code, Pandas, NetworkX, PyTorch Geometric for implementing the GNN, Matplotlib, Gephi, and Microsoft Excel.

A. Research Scheme

This research was carried out through six main stages. The first stage was a literature study to review theories and prior research related to social network analysis, Graph Neural Networks (GNN), and the spread of radicalism ideology on social media. Next, data was collected through a crawling process on the X (Twitter) and Threads platforms using Apify and Nifty with the keyword "radikalisme" (radicalism), yielding 5,000 post data from each platform, as well as 1,327 X reply data and 1,273 Threads reply data used as the basis for graph construction. The data then went through a preprocessing stage (validation, attribute adjustment, removal of duplicate data and empty attributes, text normalization) in preparation for graph construction in the next stage.

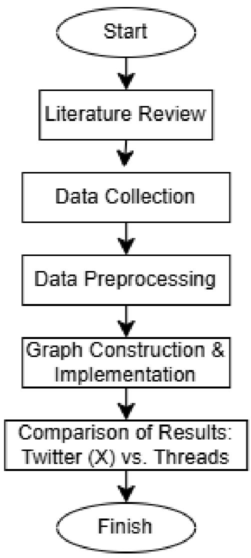


Figure 2. Research Scheme

The next stage was graph construction and GNN implementation. Interaction data was

DOI : 10.30872/jim.v21i2.29143 e-ISSN 2597-4963 dan p-ISSN 1858-4853

represented as a graph G=(V,E), with user nodes representing users who gave replies, post nodes representing the original posts, and edges representing the reply relationship between them. Graphs were constructed separately for X (Twitter) and Threads so they could be compared independently. The graph was then processed using a Graph Autoencoder model with a Graph Convolutional Network (GCN) encoder through PyTorch Geometric to produce node embeddings. The final stage was a comparison of results across platforms based on metrics such as degree centrality, density, average degree, weak components, largest component ratio, and the node embeddings produced by the model, with model performance evaluated using AUC and Average Precision.

B. Data Collection Method

Data collection was carried out through web scraping techniques using Apify and Nifty. Apify was used to collect X (Twitter) post data, X (Twitter) reply data, and Threads reply data, while Nifty was used specifically for Threads post data because Apify has limitations in the access and scraping configuration stages for retrieving large volumes of Threads post data. The data is secondary data from public content, collected using a single keyword, "radikalisme," as the search query on both platforms to keep the issue focus consistent. Data components include post text, account identity, upload time, and reply interactions. The total data collected was 10,000 post data (5,000 per platform), plus 1,327 X (Twitter) reply data and 1,273 Threads reply data that served as the basis for graph construction.

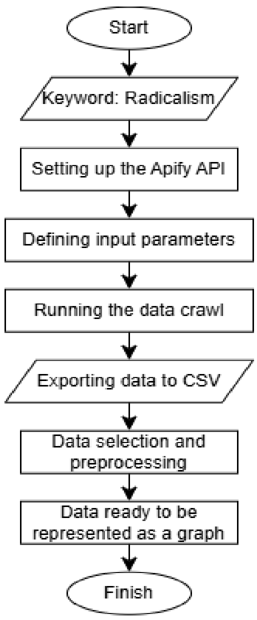


Figure 3. Data Collection Flow Diagram

The data collection process began with keyword selection, actor configuration according to platform (Twitter X Tweets Scraper and Threads Scraper on Apify, as well as Nifty for Threads posts), input parameter settings (keyword, target volume, time range), followed by crawling and exporting results to CSV format. The data then underwent preprocessing consisting of removal of duplicate data, removal of data with empty attributes, and filtering of irrelevant data. Given that X has 23 attributes while Threads has 32 attributes, normalization of column names, time formats, and interaction types was performed so that both datasets could be processed within the same analytical pipeline. To handle relationship duplication, if a user gave more than one reply to the same post, that relationship was represented as a single unique edge with a weight indicating the number of repeated interactions, resulting in 1,223 unique edges on X and 1,081 unique edges on Threads. The cleaned data was then organized into a format ready to be represented as a graph.

C. System Scheme

The system process begins with data input in the form of crawling results from X and Threads in CSV format using Apify and Nifty. This data then undergoes a preprocessing stage that includes removal of duplicate data, text normalization, and extraction of interaction relationships between accounts. Next, a graph is constructed using a Social Network Analysis (SNA) approach, in which each user giving a reply is represented as a user node, the original post is represented as a post node, and the reply interaction is represented as an edge (E). Graphs are constructed separately for the X and Threads platforms to enable comparison of network characteristics on each platform.

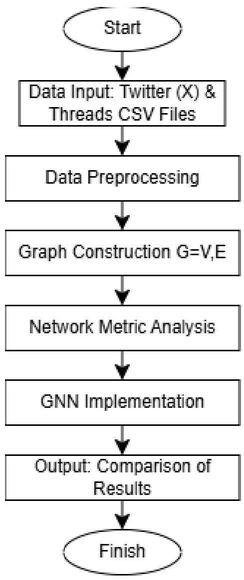


Figure 4. System Scheme

After the graph is formed, the system calculates network metrics such as degree centrality, density, average degree, weak components, and largest component ratio. In addition, the graph is also processed using a Graph Autoencoder model with a Graph Convolutional Network (GCN) encoder to produce node embeddings as a representation of the network structure, along with model evaluation

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i2.29143

values in the form of AUC, Average Precision, and graph reconstruction loss. The system output is presented in the form of preprocessing results, a list of nodes and edges, graph visualization, network metric values, node embedding results, and a comparison of interaction network characteristics between X and Threads.

D. Graph Neural Network (GNN) Workflow

The GNN workflow begins with the graph resulting from preprocessing, consisting of user and post nodes as well as reply edges. The model used is based on a Graph Autoencoder with a Graph Convolutional Network (GCN) encoder. Node features are initialized based on interaction attributes from the graph produced by the SNA approach. This research focuses on the structure of reply relationships between nodes, so the node features used are structural/topological in nature and do not involve sentiment analysis or analysis of the content of users' conversation text. These initial features form the basis for the model to learn relationship patterns through a message-passing mechanism between connected nodes, repeated layer by layer until all GNN layers have been processed.

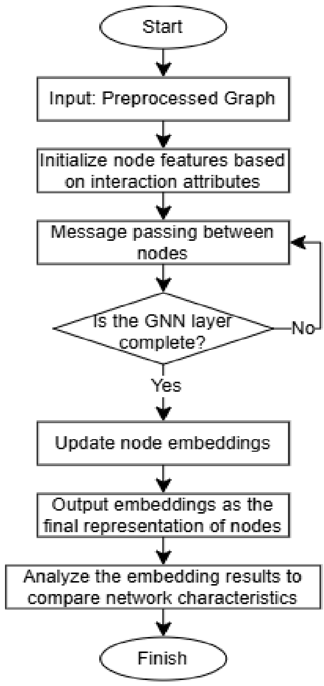


Figure 5. Graph Neural Network (GNN) Workflow

After all layers have been processed, the system updates the node embeddings as the final representation of each node, depicting the position, role, and characteristics of an account within the network structure. Model performance is evaluated using AUC and Average Precision, with the training process monitored through a graph reconstruction loss chart that shows a declining trend from the early to the final epochs of training on both platforms, indicating that the training process proceeded stably and moved toward convergence (detailed results are presented in the Results and Discussion section). These embeddings are then analyzed to compare the characteristics of user interaction networks between X and Threads, revealing differences in interaction patterns, connection density, and community structure on each platform. Thus, the GNN in this research functions as a graph representation learning method, not as a classification method or a means of evaluating individual user accounts.

4. Results and Discussion

This study develops a system for analyzing social media user responses on the X and Threads platforms toward the issue of radicalism based on a single keyword, "radicalism," using a Social Network Analysis (SNA) approach and Graph Neural Networks (GNN). Post and reply data were collected through scraping techniques (using Apify and Nifty), then processed into user post interaction graphs, analyzed using network metrics, and subsequently evaluated for representation using a Graph Autoencoder model based on Graph Convolutional Networks (GCN).

A. System Architecture & Workflow

The system is designed as a Streamlit-based dashboard consisting of four main components: data sources (X and Threads platforms), data collection process, analysis dashboard (data upload, validation, preprocessing, graph construction, network metric analysis, and GNN implementation), and system output of graph visualizations, metric values, embedding results, and exported analysis result files. The system architecture is shown in Figure 6.

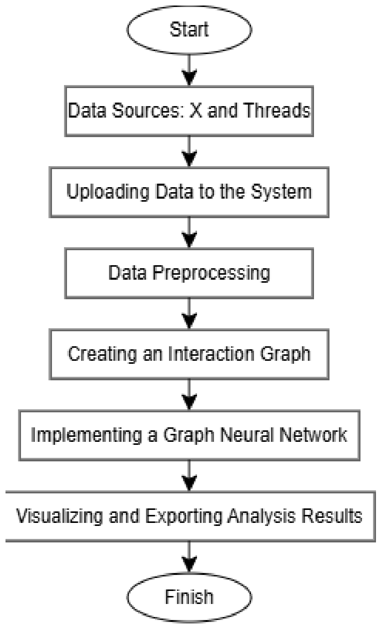


Figure 6. Architecture of the X and Threads User Response Analysis System

B. Dataset & Preprocessing

Data were collected through web scraping of public content on the X and Threads platforms using

DOI : 10.30872/jim.v21i2.29143 e-ISSN 2597-4963 dan p-ISSN 1858-4853

a single keyword, "radicalism," to ensure an equivalent search was conducted for cross-platform comparison. This single-keyword approach was deliberately chosen to maintain methodological consistency, so that any observed differences between the two platforms could be attributed to platform-specific interaction patterns rather than variations in data collection criteria. The number of raw posting data entries on both platforms was aligned at 5,000 entries each, while the number of successfully collected replies reached 1,327 on X and 1,273 on Threads. This relatively close number of replies between the two platforms, despite differences in user base and interaction mechanisms, provides a sufficiently fair basis for further comparative analysis. A summary of the dataset is presented in Table 1.

Table 1. Description of the Collected Dataset

No	Platform	Data Types	Number of Rows	Number of Columns
1	*X*	Raw Posting Data	5.000	23
2	*Threads*	Raw Posting Data	5.000	32
3	*X*	Response Interaction Data	1.327	11
4	*Threads*	Response Interaction Data	1.273	11

After the data were collected, preprocessing was carried out, including file validation, attribute adjustment, removal of empty data, data type adjustment, and merging of duplicate responses into a single weighted edge. File validation was performed to ensure the integrity and completeness of the collected data prior to further processing, while attribute adjustment and data type adjustment were carried out so that the datasets from both platforms could be aligned into a comparable structure despite their initially different column formats. Removal of empty data was intended to eliminate incomplete records that could introduce bias into the network construction, while merging duplicate responses into a single weighted edge served to reduce redundancy while still preserving the intensity of interaction between users through edge weighting. The results of the reply data preprocessing are presented in Table 2.

Table 2. Results of Reply Data Preprocessing

Platform	*X*	*Threads*
Initial Response Data	1.327	1.273
Unique Edge After Aggregation	1.223	1.081
User Node	1,000	1.009
Post Node	471	481

C. Create Graph & Analyse Network Metrics

Interaction graphs were constructed separately for each platform based on reply relationships, with users who replied as "user" nodes and posts as "post" nodes. This bipartite-style structure was chosen to explicitly capture the directional relationship between users and the posts they engage with, so that the resulting network could reflect actual reply behavior rather than an inferred user-to-user relationship. The results graph construction are presented in Table 3.

Table 3. Results of Interaction Graph Construction

No	Platform	User Node	Post Node	Total Node	Unique Edge
1	*X*	1.000	471	1.471	1.223
2	*Threads*	1.009	482	1.491	1.081

Although the final number of nodes and edges was relatively comparable between the two platforms, the formation process was not equivalent in terms of computational load, due to differences in the structure of the source raw data. This comparison is summarized in Table 4.

Table 4. Perbandingan Kompleksitas Proses Data

Aspect	*X*	*Threads*
Scraping tool	Apify	Apify + Nifty
Number of raw attributes	23 columns	32 columns
Normalization stages	1 source schema	2 source schemas
Preprocessing computational load	Light	Heavie

Thus, computational efficiency across platforms was determined not only by data volume (number of rows), but also by the heterogeneity of the source data structure. Threads required initial processing that was structurally more complex than X, even though both ultimately converged on similar final graph sizes. Network metric analysis was then carried out to determine the structural characteristics of the graph, covering density, average degree, number of weak components, and largest component ratio. These four metrics were selected because, together, they describe how tightly or loosely connected the nodes are, as well as how fragmented the overall network structure is. The analysis results are presented in Table 5.

Table 5. Results of Network Metric Analysis

No	Network Metrics	*X*	*Threads*
1	Total Edge	1.471	1.491
2	Unique Edge	1.223	1.081
3	Density	0,000565	0,000487
4	Average Degree	1,663	1,450
5	Weak Components	299	412
6	Largest Component Ratio	0,078	0,015

The very low density values on both platforms (X = 0.000565; Threads = 0.000487) indicate a sparse network structure, in which most of the connections that could possibly occur between nodes do not actually occur a common characteristic of large-scale social media interaction networks. Nevertheless, X has a higher density, average degree, and largest component ratio (0.078), while Threads has a far greater number of weak components (412 compared to 299 on X) with a much smaller largest component ratio (0.015). This difference stems from the interaction mechanisms of each platform: X facilitates reply chains that are public and easily accessible across users within a single conversation thread, so that a single post tends to attract replies

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i2.29143

from overlapping groups of users and forms larger, more concentrated network components, as seen in Figure 7. The conversation structure on Threads more closely resembles personal threads that are less cross-connected between posts, so that interactions tend to be isolated within small, separate groups.

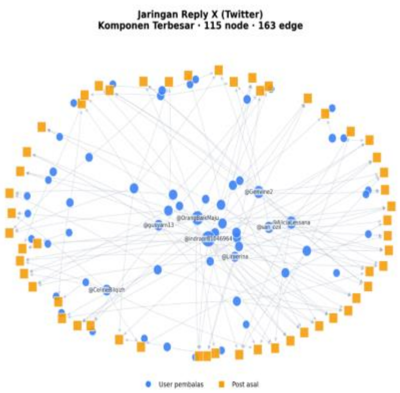


Figure 7. Graphical Visualisation of Interactions (X)

In other words, this numerical difference is not simply a matter of "X being denser," but reflects a fundamental difference in the pattern of discourse dissemination surrounding the keyword "radicalism": on X it is concentrated within relatively large interaction groups, while on Threads it is dispersed across many small, standalone conversations. This pattern has implications for how discourse on radicalism could potentially spread on each platform: the smaller number of weak components on X, combined with a higher largest component ratio, indicates that most reply interactions are concentrated within a small number of large conversation threads, so that a single topic tends to draw the simultaneous involvement of many different users, which could potentially accelerate the spread of information within the same network. Conversely, the much larger number of weak components on Threads indicates that user interactions occur more within small, standalone groups that are not interconnected, as seen in Figure 8 a pattern that may indicate a tendency toward small-scale polarization, where each group discusses the same issue separately without any cross-group point of interaction.

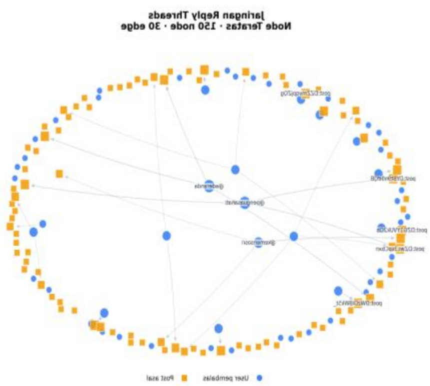


Figure 8. Graphical Visualisation of Interactions (Threads)

D. Graph Neural Network Implementation

The representation of relationships between nodes was learned using a Graph Autoencoder model with a Graph Convolutional Network (GCN) encoder, chosen for its ability to capture the topological structure of the network in an unsupervised manner without requiring labeled data. This model aims to learn structural relationship patterns and produce node embeddings, rather than to classify or label accounts. The input at this stage is the graph produced by the SNA construction process, consisting of user nodes, post nodes, and reply edges, which the model processes to learn the connectivity patterns among nodes. Model performance was evaluated through a link prediction task: a subset of edges was held out as test data (positive edges), while an equal number of unconnected node pairs were sampled as negative edges. Evaluation used Area Under the ROC Curve (AUC) and Average Precision (AP), as both metrics specifically measure the quality of graph structure reconstruction by the GCN encoder, rather than simply node classification accuracy. A summary of the results is presented in Table 6.

Table 6. Results of Graph Neural Network Implementation

Platform	**Total Node**	***Unique Edge***	**AUC**	***Average Precision***	***Model Output***
X	1.471	1.223	0,5397	0,5879	Node embedding
Threads	1.491	1.081	0,4987	0,5507	Node embedding

The model was successfully applied to the interaction graphs of both platforms and produced node embeddings for 1,471 nodes on X and 1,491 nodes on Threads. The AUC and AP values for X (0.5397 and 0.5879) were slightly higher than those for Threads (0.4987 and 0.5507). For comparison, an AUC value of 0.5 is equivalent to random guessing, so X's position slightly above this threshold indicates that the model still captures a genuine structural signal, albeit of moderate magnitude, while the Threads value close to 0.5 suggests that its structure is more difficult to reconstruct local patterns alone.

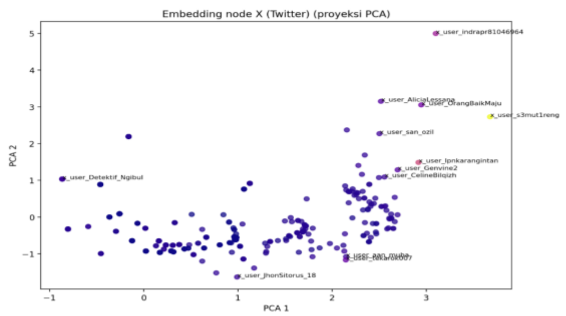


Figure 9. Visualization of X Platform Node Embeddings (PCA Projection)

The results of the two-dimensional PCA projection of node embeddings for the X platform

DOI : 10.30872/jim.v21i2.29143 e-ISSN 2597-4963 dan p-ISSN 1858-4853

are shown in Figure 9. The embeddings on X form a relatively dense cluster, indicating that its nodes tend to share similar structural roles, although a few accounts such as x_user_indrapr81046964 and x_user_s3mut1reng are positioned noticeably far from the main cluster. This density is a direct consequence of the message-passing mechanism in GCN: the large component that dominates the X graph (ratio 0.078) allows aggregation to reach more nodes simultaneously, making the contextual information absorbed by each node during training more homogeneous and resulting in denser embeddings.

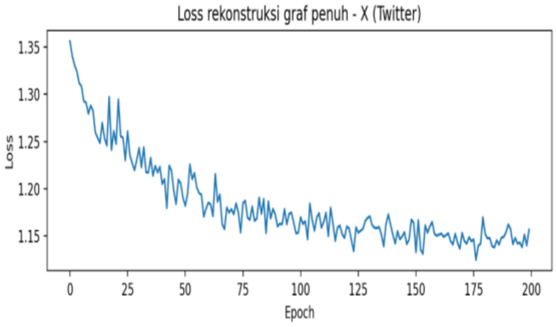


Figure 10. Full-Graph Reconstruction Loss Curve for the X Platform

This embedding density is also consistent with the reconstruction loss trend shown in Figure 10, which decreases sharply from around 1.36 to approximately 1.15 over 200 epochs and begins to flatten toward convergence after epoch 100, despite minor fluctuations early in training. This steady decline indicates that the model genuinely learned the relational structure of the X graph, rather than merely producing evaluation figures without a valid learning basis, and is consistent with the AUC and AP values in Table 6, which sit slightly above the random-guessing threshold.

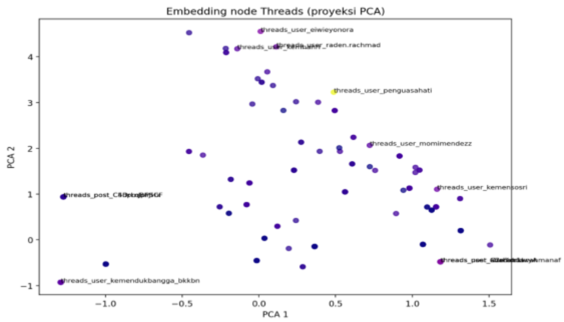


Figure 11. Visualization of Threads Platform Node Embeddings (PCA Projection)

A different pattern emerges for the Threads platform. Its embeddings are far more scattered across separate regions, reflecting a wider variation in structural position among nodes compared to X as seen in accounts such as threads_user_eiwieyonora, threads_user_kemensosri, and threads_user_kemendukbangga_bkkbn, which are positioned far apart from one another. This dispersion is a direct consequence of the large number of isolated weak components on Threads (412 components), which limits the reach of GCN aggregation, since nodes in different components never exchange information during training, resulting in embeddings scattered across separate areas. This pattern aligns with the earlier network metric analysis and confirms that the dispersion of embeddings is not coincidental, but rather a direct structural consequence of GCN's dependence on local graph connectivity as its learning signal source.

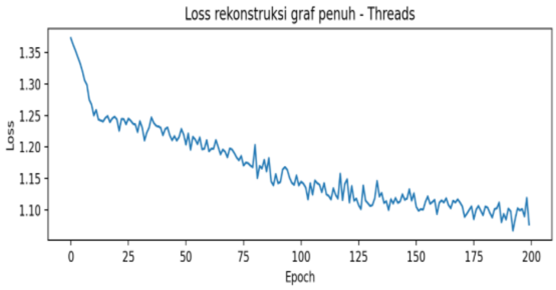


Figure 12. Full-Graph Reconstruction Loss Curve for the Threads Platform

The Threads reconstruction loss curve shows a downward trend from around 1.37 to approximately 1.10 over 200 epochs, with a decline that appears smoother and more consistent than the X curve, despite minor fluctuations, such as a brief spike around epoch 80. This convergence further supports the evidence that the Graph Autoencoder model was able to learn the relational structure of the Threads graph, even though its network is far more fragmented than that of X.

E. Comparison of User Responses

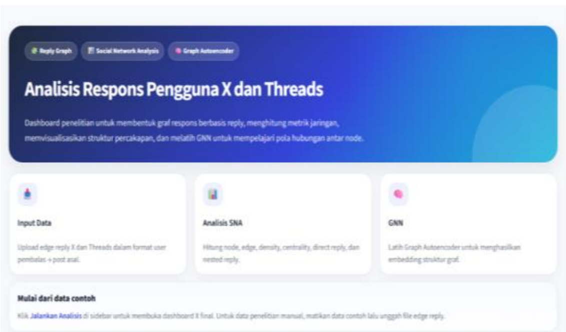
Although the amount of raw data and search keywords were identical across both platforms, the resulting network characteristics differed: X produced a more connected network (higher unique edges, density, average degree, and largest-component ratio), consistent with its nature as a fast, open, public-conversation platform, while Threads was more fragmented into many small clusters. The GNN embedding patterns reinforce this finding, as the distribution of node representations aligns with these structural characteristics. This difference carries practical implications as a basis for a decision-support system for monitoring radicalism-related discourse. The higher largest-component ratio on X (0.078 versus 0.015) indicates that a single interaction cluster can reach a far larger proportion of users, meaning that radicalism-related discourse has the potential to spread more quickly once it enters the core of that network, making X a priority candidate for early monitoring based on large-component detection, since intervention on one dominant cluster is more effective at cutting off its spread.

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i2.29143

In contrast, the much larger number of weak components on Threads (412 versus 299) indicates a more decentralized diffusion pattern across many small, independent groups; this characteristic actually makes detection more difficult, since there is no single central point, meaning that monitoring on Threads needs to be directed toward parallel, cross-cluster detection. Accordingly, both metrics have the potential to serve as quantitative indicators for prioritizing platforms and tailoring different monitoring strategies. This study several limitations: the data coverage is limited to a single keyword; the analysis focuses only on reply-interaction structure without text content or sentiment; the GNN is limited to graph-structure representation rather than account classification; the decision-support system potential described above remains conceptual and has not been empirically validated; and the AUC values, which are close to the random-guessing threshold (0.5), indicate that the embeddings are more appropriately interpreted as moderate-level structural representations. Future research could incorporate keyword variation, an extended time range, sentiment analysis, alternative GNN architectures such as GraphSAGE or GAT, and empirical validation of the proposed decision-support system.

F. System Display

The system is equipped with an interactive dashboard that supports all stages of the research, starting from data upload and interaction graph construction to the implementation of the Graph Neural Network. The system interface, which includes pages for data upload and analysis, the results of interaction graph construction in the form of node and edge structures obtained from processing reply data, as well as the results of the Graph Neural Network implementation in the form of node embeddings and model evaluation, is shown in Figure 11. Each page is designed to present the data processing workflow in a clear and structured manner, allowing users to easily follow the process from raw data input to the final model output. This integrated dashboard design aims to facilitate a more efficient and comprehensive understanding of the interaction patterns being analyzed.



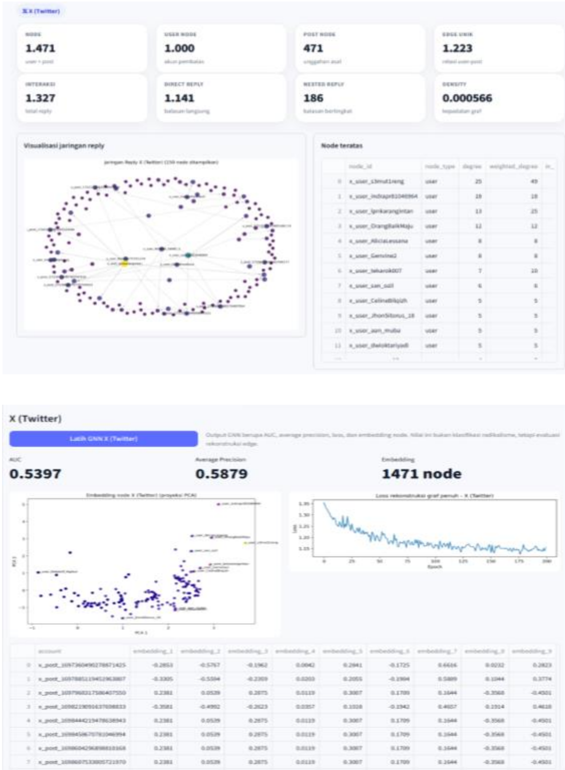


Figure 13. System View

5. CONCLUSION

This study successfully analyzed patterns of user interaction on the X and Threads platforms related to radicalism-related issues through a Social Network Analysis (SNA) and Graph Neural Network (GNN) approach. From 5,000 posts on each platform, 1,327 replies were obtained on X and 1,273 replies on Threads, which were then represented as user-post graphs consisting of 1,471 nodes and 1,223 unique edges on X, and 1,491 nodes and 1,081 unique edges on Threads. The network metric results show that X is more connected than Threads (density 0.000565 versus 0.000487; average degree 1.663 versus 1.450; largest-component ratio 0.078 versus 0.015), while Threads is more fragmented (412 weak components versus 299 on X). The implementation of a GCN-based Graph Autoencoder substantially reinforces this finding: the AUC and AP values for X (0.5397 and 0.5879) are slightly above the random-guessing threshold, indicating that the model is able to capture a genuine structural signal, albeit of moderate strength, while the Threads values, which are close to 0.5 (0.4987 and 0.5507), indicate that its network structure is more difficult to reconstruct from local patterns due to its high degree of fragmentation.

These findings carry practical implications as a basis for a decision-support system for monitoring radicalism-related discourse on social media. The high largest-component ratio on X indicates that a single interaction cluster is able to reach a far larger proportion of users, meaning that discourse has the potential to spread quickly once it enters the core of that network; this makes X a priority candidate for an early-monitoring strategy based on dominant-component detection, since intervention on a single large cluster has the potential to be more effective at cutting off the spread of information. In contrast, the large number of weak components on Threads reflects a decentralized diffusion pattern across various small, independent groups without a single central point, meaning that a suitable monitoring strategy for Threads needs to be directed toward parallel, cross-cluster detection rather than a single focal point.

Given that this study is still limited to a single keyword, an interaction-structure analysis that does not account for text content or sentiment, and a GNN representation that is structural in nature and has not yet been empirically validated as a decision-support system, further development is recommended to include the addition of sentiment analysis and content classification modules to the dashboard in order to enrich the context of the structural findings, an expansion of keyword coverage and the data-collection time range so that the temporal dynamics of discourse can be captured, and exploration of alternative GNN architectures such as GraphSAGE or GAT to improve the quality of node representation, particularly on highly fragmented graphs such as Threads.

6. DAFTAR PUSTAKA

Adek, R. T., Bustami, & Ula, M. (2021). Systematics Review on the Application of *Social media analytics* for Detecting Radical and E*X*tremist Group. *IOP Conference Series: Materials Science and Engineering*, 1071(1), 012029. https://doi.org/10.1088/1757-899*X*/1071/1/012029

AKRAM, M., & NASAR, A. (2023). A Bibliometric Analysis of Radicalization through Social Media. *Ege Akademik Bakis (Ege Academic Review)* . https://doi.org/10.21121/eab.1166627

Efendi, A. L., Fadilla, A., Khoirunnisa, A. C., Bakry, G. N., & Aristi, N. (2023). Analisis Jaringan Komunikasi #Pilpres2024 Pada *Platform Twitter*. *WACANA: Jurnal Ilmiah Ilmu Komunikasi*, *22*(2), 219–232. https://journal.moestopo.ac.id/inde*X*.php/wacana/article/view/2976

Fadlan, S., & Ramdani, D. (2022). PENERAPAN *SOCIAL NETWORK ANALYSIS* PADA JARINGAN GSM UNTUK ANALISA JARINGAN KRIMINAL. *Jurnal Teknologi Informasi,* *2*(1). http://jurnal.lpkia.ac.id/ inde*X*.php/jti/article/view/385

Jain, L., Katarya, R., & Sachdeva, S. (2023). *Opinion leaders* for Information Diffusion Using *Graph Neural Network* in Online Social Networks. *ACM Transactions on the Web*, *17*(2), 1–37. https://doi.org/ 10.1145/3580516

Jayaningsih, A. A. R., Sudiatmika, I. P. G. A., & Artana, W. W. (2024). *Twitter* vs. *Threads*: Bagaimana Media Sosial Mempengaruhi Pandangan Politik di Kalangan Pengguna Aktif. *Innovative: Journal of Social Science Research*. https://j-innovative.org/inde*X*.php/Innovative/article/view/13794

Khemani, B., Patil, S., Kotecha, K., & Tanwar, S. (2024). A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions. *Journal of Big Data*, *11*(1), 1–43. https://doi.org/10.1186/s40537-023-00876-4

Lindawati, R., Anisa, N., Latif, A., & Sabrila, T. S. (2026). *Analisis Perbandingan Algoritma Random Forest, Support Vector Machine, dan Naive Bayes dalam Klasifikasi Tingkat Pemahaman Radikalisme Pengguna Media Sosial*. *10*(1), 430–436. https://doi.org/10.46880/jmika.Vol10No1.pp430-436

Mukti, D. S., Adek, R. T., & Agusniar, C. (2026). *Topic Classification on Twitter Using a Multi - View Graph Neural Network (GNN) Model*. *6*(3), 383–391. https://itscience-indexing.com/jurnal/index.php/brilliance/article/view/9061

Patel, A., & Sutrakar, V. K. (2025). *Advanced TeXt Analytics — Graph Neural Network for Fake news Detection in Social Media*. https://doi.org/10.48550/ar*X*iv.2502.16157

Phan, H. T., Nguyen, N. T., & Hwang, D. (2023). *Fake news* detection: A survey of *Graph Neural Network* methods. *Applied Soft Computing*, *139*, 110235. https://doi.org/10.1016/j.asoc.2023.110235

Putra, F. A. R., & Sibaroni, Y. (2022). Detection of Radicalism Speech on Indonesian Tweet Using *Convolutional Neural Network*. *Building of Informatics, Technology and Science (BITS)*, *4*(2), 441–447. https://doi.org/10.47065/bits.v4i2.1907

Roffi, M. S. (2023). *Peran Intelijen Dalam Deteksi Dini Potensi Ancaman Radikalisme Pada Badan Usaha Milik Negara*. *3*(2), 14340–14353. https://j-innovative.org/index.php/Innovative/article/view/2059

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i2.29143

Surono1, A., Thamrin, D., & Ali, H. (2026). *Strategi Penguatan Deradikalisasi Melalui Media Sosial oleh BNPT: Analisis SWOT*. *2*(2), 1862–1880. https://ojs.indopublishing.or.id/index.php/iej/article/view/1251

Thayyibi, A. D., & Mansur, J. F. (2021). Implementation of *Social Network Analysis* in the Spread of Natuna Issues on *Twitter*. *JISA: Jurnal Informatika Dan Sains*, *4*(1). https://doi.org/10.31326/jisa.v4i1.899

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2021). A Comprehensive Survey on *Graph Neural Network*s. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(1), 4–24. https://doi.org/10.1109/TNNLS.2020.2978386

Zhafira, A. (2024). Inisiatif Pemerintah Indonesia Melawan Ancaman Ideologi Radikal di Sosial Media. *Jurnal Ilmiah Wahana Pendidikan*, *10*(22), 128–135. https://doi.org/10.5281/zenodo.14522905