SCIENTIFIC OASIS

Journal of Soft Computing and Decision Analytics

Journal homepage: www.jscda-journal.org
ISSN: 3009-3481

A Novel Approach for Developing Intrusion Detection Systems in Mobile Social Networks

Elaheh Rivandi[1,*] and Rozita Jamili Oskouei[1]

[1] Department of Computer Science and Information Technology, Mahdishahr Branch, Islamic Azad University, Mahdishahr, Iran

ARTICLE INFO

Article history:
Received 24 June 2025
Received in revised form 21 August 2025
Accepted 10 October 2025
Available online 20 October 2025

Keywords:
Intrusion Detection Systems; Mobile Social Networks; Ad Hoc Networks, Packet Drop Attack; Black Hole Attack; Sleep Deprivation Attack; TCP SYN Attack.

ABSTRACT

Intrusion Detection Systems (IDS) have been developed to improve security levels in networks. These systems can observe user activities and, upon detecting any abnormal behavior suggesting a potential hacker, issue an alert to network security managers while simultaneously limiting the hacker's activities. In this study, we propose an IDS based on communication analysis in social networks for ad hoc networks. The proposed IDS monitors the various relationships between ad hoc nodes and triggers an alert when one or more rules governing normal communications are violated. The performance of the proposed IDS is assessed under different system factors (such as mobility and traffic load). In this study, we tested the system against common attacks (such as packet dropping, black hole, sleep deprivation, and TCP SYN attacks). The results show that the proposed IDS can detect attacks with high speed, and the rate of false attack alerts is lower compared to previous work. Moreover, the results show that the proposed IDS offers significant advantages over IDSs based on the Apriori algorithm (in terms of computational complexity).

1. Introduction

Today, social networks are used by people across the globe for various purposes (business, advertising, education, family, friendship, etc.). Interestingly, even older people with little to no internet knowledge are utilizing social networks, particularly Telegram and WhatsApp (in Iran). Some of them start using these platforms for entertainment or to communicate with their children living in distant cities or countries. Social networks such as WhatsApp allow them to have voice or video conferences with their family at a much lower cost compared to traditional phone calls. Most of these users access mobile social networks using mobile devices (such as smartphones, tablets, etc.). Since these devices are mobile, the communication technologies used to connect them to the internet are wireless (such as Wi-Fi, RFID, GSM, etc.). As a result, these technologies are far more vulnerable in terms of security compared to LAN connections [1,2]. Intrusion Detection Systems, along with

* *Corresponding author.*
E-mail address: Elaheh.rivandi.research@gmail.com

https://doi.org/10.31181/jscda31202576

antivirus software and firewalls, act as defensive barriers in computer and internet networks, especially in mobile social networks. The effectiveness of such cybersecurity measures is deeply influenced by a complex interplay of individual, social, and organizational determinants [3], highlighting the need for robust technical systems that can operate effectively within these human-centric contexts. These systems can implement methods to detect intrusions and attacks carried out by hackers or internal and external intruders. The faster these IDSs can detect intrusions and attacks, the less damage will be caused by the intrusion [4]. Despite the broad research conducted over recent decades to improve the functioning of IDSs, challenges still exist (such as: the incapability to detect invasions and attacks immediately or in the early moments of an intrusion, and the inability to recognize recent or previously unseen intrusion methods), especially in social networks.

Machine learning and deep learning approaches have been widely used in different areas such as Cyber security, FinTech, Neuroscience and psychology [5-7]. For instance, advanced models like NARDL and dynamic parametric systems are critical for forecasting and analyzing uncertainty in stock markets [8-9]. Similarly, machine learning techniques are being leveraged to analyze complex economic factors in business, demonstrating the cross-domain applicability of these advanced analytical methods [10, 11]. Further efforts are also being made to standardize security and regulatory frameworks for emerging technologies, such as blockchain smart contracts [12]. Beyond security, the rapidly evolving field of Artificial Intelligence is driving innovation across sectors, including business and educational applications like automated assessment and decision-making for diversity and inclusivity [13-15]. In this study, we aim to propose an innovative approach for developing a new invasion detection system to identify intrusions and attacks in mobile social networks. In the proposed method, the various relationships between nodes (in this case, users connected to mobile social networks) are constantly monitored, and an alert is triggered when one or more communication rules are violated. Parameters such as (node mobility, network traffic load, etc.) are considered in rule creation and intrusion detection, and they play a very important role. Numerous matrices are created to simultaneously examine multiple relationships, and multi-relational analysis can reduce the number of false alerts and improve the precision of the intrusion detection system. This research is categorized into five sections. In the second section, we discuss intrusion detection systems and their use in detecting attacks in social networks. In the third section, we explain the proposed method, and in the fourth section, we discuss how the proposed method is implemented and evaluated. Finally, the fifth section contains the conclusion.

2. Literature Review

Intrusion Detection Systems have been presented by various researchers as one of the key components for establishing security in computer networks. The main objective of these systems is to monitor the network and identify intrusions by analyzing user behavior. In general, the main objectives of IDSs are as follows [16]:

- Contribution in identifying intrusions or attacks implemented in the network
- Detect and identify potential attacks and notify the network or system administrator, as the purpose of IDS is not to prevent attacks.
- These systems are usually used together with other security systems such as firewalls, antivirus software, and anti-hacking tools to improve security.
- The responsibility of these systems is to identify and detect all not authorized use of the system and any type of misuse or harm which comes from inside users or outside ones.
- IDSs are implemented in both software and hardware forms, each with its own advantages and disadvantages.

The general components of IDSs are as follows [17]:

- Collector: Creates an interface for retrieving data that is used for the detection process. Usually, a data collector is employed in network operations to facilitate access to all raw packets in a network inside of a specific geographical location. Interface users can also access host-based data or external databases as collectors.
- Detector: Executes the actual detection process. This component works as the core of the system, accessing the gathered and stored data to decide what should be considered a threat and trigger an alarm.
- User Interface: Used to report results to the user and enables them to control the IDS. This interface provides simultaneous access to the storage source, detector, and responder.
- Storage Source: Stores the data continuously required by the detector or user interface. This data is either noticed by the detector or obtained from external devices. The storage source manages the system's database.
- Responder: Reacts to identified intrusions to prevent future security damage to the system. A responder can be activated either automatically or manually through the user interface.

Social networks are used to facilitate communication between people all around the world, but unfortunately, some individuals try to create security problems on these networks, which can endanger people's privacy and security. As a result, using methods (such as IDSs) to detect security issues early is essential. Below section provides a brief review of some research conducted by various scholars to enhance the performance of IDSs. Retnaswamy and Ponniah [18] presented a method for spotting attacks in a network using different agents (broker, IDS interface, and response agent). They developed an IDS which is capable of detecting destructive attacks and intrusions in a network in a very effective way. In this system, the broker agents, IDS interface, and response agent could detect intrusions and prevent the damage caused by those intrusions. Al-Badawy *et al.,* [19] proposed a framework to protect the geographic privacy of users and prevent users' friends on social networks from knowing their exact geographical location. The main goals of this framework are:

- Protect users' identities and geographic locations.
- Provide a strong encryption mechanism.
- Offer an essential agreement procedure to evade the restrictions of public key protocols before sharing or exchanging data directly.

Aleid *et al.,* [20] proposed a universal structure capable of being applied to all materials in social media to build a system which is intelligent. The proposed framework in this study contains three main factors: an interface, an analyzer, and a classifier. Huang and Lee [21] conducted a cross-analysis to discover the correlation among each feature with others using a decision-based classification algorithm to detect local abnormalities. They demonstrated that as a distributed IDS, creating extreme communication and computation among nodes lowers network performance. Similar optimization approaches have shown promise in related domains; for instance, Abapour *et al.,* [22] demonstrated the effectiveness of an Ant Colony Optimization algorithm combined with the RPL protocol for securing routing in IoT networks, highlighting the potential of bio-inspired algorithms in addressing security challenges across different network architectures. Al-Khanjari *et al.,* [23] proposed a method for implementing an IDS based on internal sensors in government websites. Internal sensors consist of an additional source code added to the e-government website, and intrusions that happen can be spotted in real-time using internal IDS systems. These systems, based on internal sensors, include mechanisms for organizing the data collected by IDSs. The performance evaluation of the proposed system showed that it is possible to develop an IDS which has the ability to identify different types of intrusions and detecting various attacks. In this study, the researchers

used small sensors to detect intrusions in the e-government website's source code. These sensors searched for specific intrusion signatures and monitored the website's behavior in real-time. The study concentrated primarily on protecting image data. This focus on analyzing complex, interactive systems is reflected in other domains, from understanding the determinants of user intention in immersive virtual reality environments [24] to developing frameworks for categorical reasoning in educational contexts [24,25] underscoring the universal challenge of modeling and securing dynamic interactions.

3. The Proposed Model

In this study, we propose an original method for creating an IDS in mobile social networks (such as Telegram, Viber, etc.). Figure 1 shows the overall structure of the proposed IDS based on mobile social networks.

Based on Figure 1, the proposed Intrusion Detection System features a distributed and collaborative architecture. This IDS can be installed at each node, letting a group of nodes to share diagnostic information for global alert decision-making. As illustrated in Figure 1, the overall structure of the proposed intrusion detection system consists of three modules:

Data Preprocessing Module: The data collection component collects all important inspection data based on the feature set outlined in Table 1. Next, the preprocessing component encodes the data in queues using a sliding window (sampling period). The data formats within networks are mainly numeric, as the analysis of social networks in this study is based on numerical data. This data is then classified to construct various ego networks (an ego network is formed around a single actor (ego) along with connected actors and all the relationships between them) for the subsequent module.

Social Network Creation Module: This module provides more flexibility for designing the IDS for various purposes. A significant advantage of using social networks is the inherent flexibility in defining "actors" and the "relationships" between them. For ego networks with diverse actors and relationships, identical analytical methods can be employed, meaning that the system does not need to convert its analytical tools. Instead, different objects apply analytical criteria for various design objectives.

Intrusion Response Module: This module enables supervisory nodes to combine local intrusion alerts with global intrusion alerts. Outputs from the social network analysis module generate local alerts when violations of the governing rules of social relationships (i.e., centrality criteria) exceed typical ranges. In this research, we use data traffic and control mechanisms to create social networks. An adjacency matrix indicates each social media. For the control matrix (which includes management messages), the "degree" metric is calculated, as "betweenness" is less significant for control messages. For the data matrix, both "betweenness" and "degree" metrics are measured. Although we do not use other matrices, such as MAC ACK, Request Route, Route Reply, or Route Error for monitoring, all ego networks can be effortlessly created when necessary. Each ego network can display a relationship and consolidate results from various metric analyses, aiding in monitoring multiple relationships. For example, a malicious node introducing infected packets into the network may involve both the RTS and data traffic matrices. Different relationships can generally be recorded by separate matrices (multidimensional matrices), and relationships can also be specified as multiple matrices with different qualitative values for different relationships.

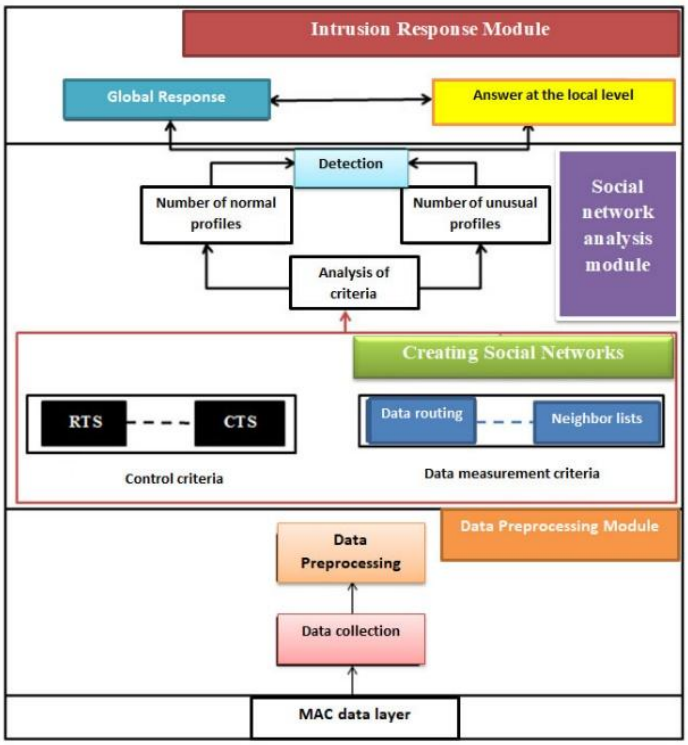

Fig. 1. General structure of intrusion detection system based on mobile social networks

The profile creation process is divided into two phases:

- Training Phase: During this phase, training data is gathered over a long period (3000 seconds) compared to the testing data (2000 seconds). We select a relatively long duration for data collection to make sure we do not overlook unexpected variations in the data we test. If the training period lacks enough length to capture these statistical changes, the normal profile will not be accurate. Through multiple simulation repetitions, averaging the results of the metrics analysis component builds a normal profile.
- Testing Phase: In this phase, the social metrics calculated from the testing data are compared with the normal profile. If the statistics from the testing data exceed the predefined thresholds of the normal profile, an anomaly will be reported.

4. Simulation and Performance Evaluation of the Proposed Method

In this study, we focus our analysis on ego local networks and using NS2 software for simulation. Table 1 outlines the specifications of our simulation environment.

In this study, we utilize the Random Waypoint (RWP) model to simulate node mobility. RWP is commonly employed to model user movement in the context of ad hoc routing protocols and transport layer protocols. In the simulation, each node starts its movement from a randomly chosen location with a speed selected from a range of [zero to maximum speed]. Once a node reaches its target, it remains in a pause state for a specified duration before randomly selecting its next destination. The maximum speed can vary based on various methods, and the pause time, network topology, and dynamics can be modified as needed. The overall distribution of network traffic is influenced by the number of traffic connections, the total number of nodes, and their movement patterns. With the parameters set in this thesis (Table 1), our ad hoc network is approaching saturation, with all nodes engaged in sending and receiving data packets. In the simulations conducted in this research, time 0 signifies ths50, and during the first 50 seconds, data from all training and testing simulations is discarded based on the initialization process in the simulator. The

AODV-based ad hoc networks in this study use link layer feedback or HELLO messages for definitive communication detection. Due to challenges in obtaining IEEE 802.11 feedback in real network connections, many ad hoc networks use HELLO messages to control connectivity status in local networks. If a node does not receive broadcast messages over a defined distance, a HELLO message is locally broadcasted. Consequently, in each time period, there is at least one Hello message transmission. Failure to receive a HELLO message from a neighbor over several time intervals specifies that the neighbor is out of the range of transmission, resulting in a lost connection.

Table 1
Specifications of the simulation environment

Amount/Value	Parameter
500 meters by 500 meters	Topology
Random Waypoint Model (RWP)	Node mobility
250 meters	Radio range
AODV	Routing protocol
11 Mbps	Link capacity
10 packets per second or 512 bytes per packet	Average transfer rate in data stream
25	Number of connections
30	The number of nodes in the set
10/100/300/1000 seconds	Duration of pause
1.5 meters per second	Maximum movement speed

Thus, HELLO messages serve two purposes:

- To confirm the establishment or loss of connectivity between various nodes in the network.
- To collect general information about all nodes that are interconnected within the network.

In ego networks, all 'alter' (or non-adjacent) nodes can extract routing table entries from HELLO messages or send out control messages to the 'ego' node, which can then make informed judgments about whether 'alters' have direct connections to other nodes. This can potentially prevent increasing network overhead, although memory space is required to store the table. In this thesis, we collect information about each node's neighbors every 5 seconds to update the values which are necessary for establishing "relationships" between nodes in the network. Consequently, the time intervals applied in this research ("x-axis time") are set to 5 seconds.

4.1 Validation of the Proposed Intrusion Detection System

To validate the proposed intrusion detection system in this research, we simulated four common attacks (including packet dropping, black hole, denial of sleep, and TCP SYN flooding) that frequently occur in ad hoc networks. To assess the performance of the intrusion detection system which is proposed, we employ three key metrics:

- Successful Detection Rate of a Node: It is described as the probability of a node successfully identifying attacks within a specific time frame.
- Successful Detection Rate of Multiple Nodes: It is described as the ratio of the number of nodes that correctly identified attacks within a given time frame to the total number of nodes in the network.

False Positive Rate of Multiple Nodes: It is described as the ratio of the total number of nodes incorrectly identifying all non-attack incidents as anomalies within a specific time frame to the total number of nodes in the network.

The simulation results show that local betweenness rises to a certain amount in the routing table after the primary network is established, fluctuating within a specific local range that reflects changes in routing information during node mobility or failure. Thus, throughout both operations on training data (without attacks) and testing data (with attacks), we use a sliding window to measure the decrease in betweenness fluctuations. Using the training data, we can conduct a set of local measurements, which might include the mean and standard deviation (Bi and Si) for each window i. Upon comprehensive analysis of the training data sets, it becomes evident that the mean value of {Si/Bi} remains close across all nodes over different time intervals, specifically for stop durations of 10, 100, and 1000 seconds. We assume that the random variable follows a Gaussian distribution with mean μ and standard deviation σ. We utilize this random variable to build a normal profile from the training data. With this profiling criterion, the proposed method in this research demonstrates high efficacy in detecting the anomalies even at circumstances which are very unstable (such as rapid topology changes and continuous connections or disconnections of nodes).

Betweenness to put the data to the test, we use two standard deviations (σ) from μ as the confidence interval (CI) to ensure 95% confidence (level of certainty) and apply {Si / Bi} local metrics that fall within the CI. "Betweenness" indicates the importance of a node in the network. The shortest line drawn between two points on a plane (GP) is defined as the least distant rout between two nodes in the network. The betweenness of a node K is calculated using (1) [26]:

$$C_B(k) = \Sigma_{\forall i,j \in G,\, i\#j\#k\#i} \frac{gikj}{gij} \quad (1)$$

In equation (1), g_{ikj} represents the number of paths drawn from node *i* to *j* using intermediary node K, while g_{ikj} indicates the overall number of paths drawn with or without the involvement of node K. We designate node number 15 as the spy (or attacker) conducting the black hole attack and study the effects of node 15's attack on its neighboring node (node 27). It is important to note that because node identifiers are randomly assigned in each simulation, node number 15 is clearly identified as the attacker. As shown in Figure 2, during the attack, the betweenness of node number 15 (the attacker) increases in an effort to attract other nodes to connect to its ego network, with all new nodes joining the network directly connected only to the hacker.

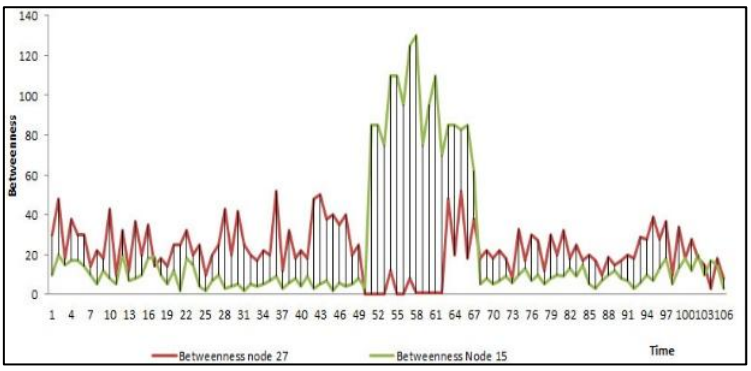

Fig. 2. Changes in Betweenness Under Black Hole Attack (a)Betweenness of the Attacker Node 15 (b)Betweenness of Neighboring Node 27

During the black hole attack, the betweenness of node 27 decreases as many altered nodes sever their connections with it. Consequently, the ego network shrinks, comprising fewer altered nodes. Moreover, the impact of betweenness on neighboring nodes is not limited to those directly adjacent to the attacker. Without direct contact with attacker node 15, non-adjacent nodes in the ego network also experience changes during the attack. Table 2 shows Betweenness Values for Ego Node and

Altered Nodes. As shown in Table 2, if node 2 has neighboring nodes (such as nodes 6, 21, 25, and 27) that are victims and close neighbors of the attacker node 15, communications from these victim nodes to all other altered nodes will drop to zero immediately at the onset of the attack. Consequently, the betweenness of node 2 significantly increases.

Therefore, even with unchanged neighboring nodes, node 2—which is not in the victim list (within one hop)—can detect the attack within several hops. This illustrates how altered node dynamics directly influence the detection capabilities of non-victimized nodes in the network during an attack.

Table 2
Changes in Betweenness for Node 2 Before and After the Black Hole Attack

	Betweenness	
ID	Before Attack (Time=545.181s)	After Attack (Time=550.376s)
2	5.046	49.61
0	1.032	2.976
6	0.794	0.0
7	0.1	0.1666
8	0.969	0.7
13	1.594	1.693
17	4.192	1.493
18	2.399	2.11
21	1.288	0.0
24	0.8996	0.1429
25	1.5496	0.0
27	2.2131	0.0
28	2.6329	2.11

Figure 3 illustrates the betweenness of a non-adjacent node that includes some neighbors (such as nodes 6, 21, 25, and 27) that are victims and close neighbors of the attacker node 15. As the attack commences, communications from these victim nodes to all altered nodes will diminish to zero. Consequently, the betweenness of node 2 will gradually increase as a numerical counter in (1). Therefore, with unchanged neighboring nodes, node 2, which is not on the victim list (within one hop), can detect the attack within several hops.

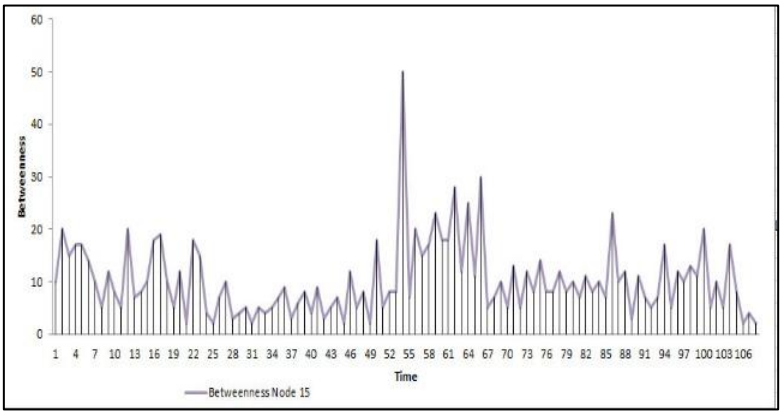

Fig. 3. Betweenness of Non-Adjacent Node During a Black Hole Attack

This highlights the importance of non-adjacent nodes in recognizing anomalies and maintaining network integrity, even when directly affected nodes are being compromised. Betweenness in the Ego Network We calculated the betweenness for all 30 nodes in the ego network, and the results

indicate that significant statistical changes occur in between the attack period (Table 3). The proportions of successful and erroneous recognitions depend on various features.

Table 3
Average Ratios of Successful and False Positive Detections

Multiple nodes false positive rate	Multi-node detection rate (%)	Detection rate (%) of a node	Stop time (seconds)
black hole attack			
2.22	43.33	100	1000
1.48	54.81	100	300
0.37	60.37	100	100
0.37	84.81	100	10
Sleep restriction			
2.96	45.56	100	1000
1.57	57.41	100	300
0.37	64.07	100	100
0.37	66.67	100	10

In other words, it can be observed that more than 50% of the nodes can notice these attacks, and both neighboring and non-adjacent nodes can respond during an attack. This can result in achieving 100% success in detecting attacks on a single node. At the same time, the rate of successful detections decreases as the pause time increases. This can be understood through the "small-world phenomenon" in social networks. The "six degrees of separation" principle says that on average, every person passes through 6 stages in their life on Earth. In our small ad-hoc network, which consists of 30 nodes, the effect of attacks on node relationships spreads across the network in several hops. However, only the nodes that significantly affect the victim nodes can easily detect the attacks carried out by those nodes. Therefore, the detection rate of attacks when an attack happens depends on the number of nodes that fall inside the attack range. With less mobility, the number of nodes affected by the attacks is limited, whereas with more mobility, more information is disseminated across the network. The range of radio transmission also affects attack detection performance. In the NS2 simulator, the default long-distance transmission range is 250 meters (radius), while the preconfigured detection range for the 802.11 standard is 550 meters. To explore the outcomes of a shorter transmission range on the proposed system in this thesis, we changed the range of radio transmission in NS2 to 100 meters and re-ran the simulations for black hole and sleep deprivation attacks. With a transmission range which is shorter, the size of ego networks becomes smaller. In Table 4, we can see that the performance of the proposed IDS in this thesis slightly decreases but remains acceptable. Based on Table 4, it can be argued that the correct detection rate decreases for a few nodes because less number of nodes are directly or indirectly targeted as victims. Also, the false positive rate increases because the ego's vitality decreases with a reduction in the number of changes. The "betweenness" parameters used for detection are more responsive to the movements and dynamics of the nodes, which can lead to some false positive alerts.

4.2 Comparison of the Execution Complexity of the Proposed Method with Other Methods

In this section, we compare the runtime complexity and effectiveness of the proposed intrusion detection system in this research with the complexity of intrusion detection using the association rule mining technique based on the Apriori algorithm. Association rules in data mining have been successfully implemented in IDSs in networks and can identify normal behaviors from abnormal ones [26]. The major task of this algorithm is to reduce a large amount of information into a small,

understandable, and statistically supported set. For example, in the test performed in this research, one of the discovered rules is: (routingDataPkt) ⇒ (DATA)(0.1021, 1)

This rule states that a MAC `routingDataPkt` is observed in 10.21% of transactions, and all of these packets are 100% data packets. This rule is insignificant because it provides no information about whether behaviors are normal or abnormal, yet the machine is incapable of eliminating this rule. Therefore, the complexity of detection in IDS increases as the number of rules grows. Given that the size of the information collected from MAC layer traffic is usually enormous, even for a short period of time, and since the amount of rules created by the Apriori algorithm is typically substantial, the number of false positive alarms generated will also be very high. Hence, in this research, we used the MFI (Maximum Frequent Item sets) criterion. MFI is defined as a frequent itemset where no adjacent item sets are frequent. The process of pruning significantly drops the size of the normal rule-based profile while still capturing related recurring patterns from the dataset. In the simulation that we run, MFI pruning could reduce the number of association rules by 20 to 40%. The implementation of IDS using the Apriori algorithm and MFI criteria has been conducted in this thesis. Tables 4 and 5 shows examples of anomalous rules obtained from the Apriori algorithm for misbehaving node 15.

Table 4
The Average Successful Detection and False Positive Rates with a Radio Transmission Range of 100 Meters

False positive rate (%) of several nodes	Detection rate (%) of multi-module	Detection rate (%) of single node	Stop time (seconds)
Black hole attack			
8.33	45.0	100	1000
4.76	42.0	100	300
4.0	46.67	100	100
2.67	48.47	100	10
Sleep restriction			
8.33	42.78	100	1000
7.5	35.83	100	300
5.0	46.76	100	100
5.33	30.67	100	10

For instance, in the Flooding attack, all three rules show that node 15 sends a massive amount of data packets to the monitoring node. In the Blackhole attack, the first two rules show a large amount of data packets destined for DA15, and the third rule shows that data packets from several neighboring nodes are sent to the source of attack with destination DA15. All these rules can be addressed using social criteria. Association rules organize large amounts of traffic data similarly to a "shopping basket," while the social network method organizes the same data based on social relationships between the network's nodes, which conceptually provides a more visual approach. Unlike association rules, which can be used to detect anomalies in the network, social networks can cover a wide range of interactions between nodes. However, association rules are still used for analyzing social networks. In this research, three ego networks have been created to monitor the MAC layer (including direct traffic and physical data traffic), and the neighbors at the network layer are listed. Each node moves with a maximum speed of 5 meters per second, with a pause time at each position of 10 seconds, and the simulation runtime is 30 minutes. The average detection delay for the "Social Analysis Module" in Figure 3 (for creating ego networks to alert an attack) is 0.424 seconds. As the number of ego networks raises, the processing time increases linearly. Although the adjacency matrices are sized N × N, only the physical nodes within range for each ego are calculated, so the complexity does not increase as the total network size grows. In this research, the optimized

code developed for implementing the IDS serves as the basis for the complexity performance evaluation. For comparison, under similar conditions using the Apriori-based IDS, we used an average of 930,000 transaction data over a total of 30 minutes for each node, with each transaction having seven features. Before aggregation was applied to the data, 264 rules were generated, of which only 21 rules remained after aggregation, and 11 rules could be used for conclusive discovery. The average detection delay for the association rule discovery algorithm was 1.051 seconds, which is deducted from the time spent obtaining the rules. Thus, the proposed IDS in this thesis is simpler in terms of implementation and computations compared to both the social criteria (Socio) and the association rules based on anomaly detection. Table 5 shows the results of the studies conducted on the association rule-based IDS and the proposed IDS in this research.

Table 5
Comparison of The Effectiveness of Apriori-Based Ids with The Proposed Ids in This Research

% False Positive Rate of Multiple Nodes	Multiple Node Detection % Rate	Single Node Detection % Rate	Type of Attack
IIDS based on Apriori Algorithm			
0.5	38.33	93.0	Dropping
0.3	46.46	99.3	Black Hole
0.7	36.67	90.0	Sleep Restriction
IDS Proposed in This Research			
0.08	40.0	99.9	Dropping
1.11	60.83	99.8	Black Hole
1.32	58.43	99.9	Sleep Restriction

The false detection rate is examined on average across four different pause times: 1000/300/100/10 seconds. Mobility describes the "dynamics" of the nodes within the network, and we use various average pause times to discover some usage patterns in real-world networks, as we cannot assume that all nodes move at the identical speed in the ad hoc network. As observed, both systems include an acceptable false positive rate, with the IDS based on the Apriori algorithm performing slightly better than our proposed method. On the other hand, the IDS proposed in this thesis shows a higher detection rate for single and multiple attacking nodes compared to the Apriori-based IDS for various attacks. This is because, in the IDS proposed in this thesis, non-neighboring nodes can also assist in detection through the analysis of social relationships, while in the Apriori-based IDS, such detections are more difficult to achieve.

5. Conclusion

In this research, a social network-based intrusion detection system was proposed. The proposed system was then evaluated using the NS2 software in an environment consisting of 30 nodes. Three types of attacks (including packet dropping, sleep deprivation, and blackhole attacks), which are the most well-known attacks in Ad-Hoc social networks, were considered, and the performance of the proposed IDS in detecting these attacks was tested. Finally, the outcomes achieved from the evaluation of the proposed IDS's performance in this research demonstrated that the system could detect the attacks under study with high accuracy (a high rate of correct detections and a low number of false positives). The performance of the proposed IDS was also compared with an intrusion detection system based on association rule mining algorithms (specifically the Apriori algorithm). It was observed that the proposed IDS in this research has lower computational and system complexity compared to the Apriori-based system. Moreover, the attack detection time in the proposed IDS is shorter, and the number of rules generated for use in intrusion detection is also fewer compared to the Apriori-based IDS. In the simulation conducted in this study, pruning the MFI can reduce the

number of association rules by 20 to 40 percent. The implementation of an intrusion detection system using the Apriori algorithm and MFI metrics was carried out in this research. There is a linear increase in processing time, as the number of ego networks increases. Although the adjacency matrices are of size N × N, only the physical nodes within the range of each ego are calculated, so the complexity does not increase with the expansion of the entire network. The assessment of time complexity performance relies on the execution of the intrusion detection system developed using optimized code in this study. On the other hand, the proposed IDS in this research achieves a higher detection rate for single and multiple attacking nodes compared to the Apriori-based IDS for various types of attacks. This is because, in the proposed IDS, even non-neighboring nodes can assist in detection through social relationship analysis, whereas such detections are more challenging in the Apriori-based IDS.

Acknowledgement
This research was not funded by any grant.

Conflicts of Interest
The authors declare no conflicts of interest.

References
[1] Kavitha, D. (2017). Survey of data mining techniques for social networking websites. *International Journal of Computer Science and Mobile Computing, 6*(4), 418-426.
[2] Adedoyin-Olowe, M., Gaber, M. M., & Stahl, F. (2013). A Survey of Data Mining Techniques for Social Media Analysis. *arXiv preprint arXiv:1312.4617*. https://doi.org/10.48550/arXiv.1312.4617
[3] Nasiri, S., Shahabi, S., Shafiesabet, A., Talebbeidokhti, M., & Behineh, E. A. (2026). Cybersecurity in Action: Unraveling the Effects of Individual, Social, and Organizational Determinants. *Tehnički glasnik, 20*(2), 1-10. https://doi.org/10.31803/tg-20240627004731
[4] Provan, K. G., Fish, A., & Sydow, J. (2007). Interorganizational networks at the network level: A review of the empirical literature on whole networks. *Journal of management, 33*(3), 479-516. https://doi.org/10.1177/0149206307302554
[5] Mirfathollahi, A., Ghodrati, M. T., Shalchyan, V., Zarrindast, M. R., & Daliri, M. R. (2023). Decoding hand kinetics and kinematics using somatosensory cortex activity in active and passive movement. *iScience, 26*(10), 107808. https://doi.org/10.1016/j.isci.2023.107808
[6] Baheti, R., & Gill, H. (2011). Cyber-physical systems. *The impact of control technology, 12*(1), 161-166.
[7] Dwyer, D. B., Falkai, P., & Koutsouleris, N. (2018). Machine learning approaches for clinical psychology and psychiatry. *Annual review of clinical psychology, 14*, 91-118. https://doi.org/10.1146/annurev-clinpsy-032816-045037
[8] Roshdieh, N. (2024). The Effect of Monetary Policy Uncertainty on Stock Market Uncertainty with NARDL Approach. *Research Journal of Finance and Accounting, 15*(10), 1-9.
[9] Hosseinidoust, S. E., Fotros, M. H., & Massahi, S. (2016). Application of Dynamic Parametric and Non Parametric Systems in Stock Market Return Forecasting: Case Study of Tehran Stock Market. *Quarterly Journal of Fiscal and Economic Policies, 3*(12), 125-148. http://qjfep.ir/article-1-289-en.html
[10] Bagherabad, M. B., Rivandi, E., & Mehr, M. J. (2025). Machine Learning for Analyzing Effects of Various Factors on Business Economic. *Authorea Preprints*. https://doi.org/10.36227/techrxiv.174429010.09842200/v1
[11] Rivandi, E. (2024). FinTech and the Level of Its Adoption in Different Countries Around the World. *Available at SSRN 5049827*. https://dx.doi.org/10.2139/ssrn.5049827
[12] Heidari, S., Hashemi, S., Khorsand, M. S., Daneshfar, A., & Jazayerifar, S. (2024). Towards standardized regulations for block chain smart contracts: Insights from Delphi and SWARA analysis. *Amity Journal of Management, XI*(II), 1-15. https://doi.org/10.31620/AJM.1121
[13] Karizaki, M. S., Gnesdilow, D., Puntambekar, S., & Passonneau, R. J. (2024, July). How Well Can You Articulate that Idea? Insights from Automated Formative Assessment. In *International Conference on Artificial Intelligence in Education* (pp. 225-233). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-64299-9_16

[14] Sadeghi, S., & Niu, C. (2024). Augmenting Human Decision-Making in K-12 Education: The Role of Artificial Intelligence in Assisting the Recruitment and Retention of Teachers of Color for Enhanced Diversity and Inclusivity. *Leadership and Policy in Schools*, 1-21. https://doi.org/10.1080/15700763.2024.2358303
[15] Nikzat, P., & Noorymotlagh, M. (2025). Artificial Intelligence in Business: Driving Innovation and Competitive Advantage. *International journal of industrial engineering and operational research, 7*(3), 50-62. https://doi.org/10.22034/ijieor.v7i3.180
[16] Fire, M., Katz, G., & Elovici, Y. (2012). Strangers intrusion detection-detecting spammers and fake profiles in social networks based on topology anomalies. *Human journal, 1*(1), 26-39.
[17] Sayar, A. A., Pawar, S. N., & Mane, V. (2014). A review of intrusion detection system in computer network. *International Journal of Computer Science and Mobile Computing, 3*(2), 700-703.
[18] Retnaswamy, B., & Ponniah, K. K. (2016). A new ontology-based multi agent framework for intrusion detection. *International Journal of Communication Systems, 29*(17), 2490-2502. https://doi.org/10.1002/dac.3189
[19] Al-Badawy, A. M., Abbas, H. M., & Belal, M. (2018). A TTP-Free Location Privacy Framework for Mobile Social Networks with Key Agreement Protocol. *International Journal of Applied Engineering Research, 13*(14), 11540-11547.
[20] Aleid, H., Alkhalaf, A. A., Taemees, A. H., Almurished, H. S., Alharbi, K. A., & Alanazi, N. H. (2018). Framework to classify and analyze social media content. *Social Networking, 7*(02), 79.
[21] Huang, Y. A., & Lee, W. (2003, October). A cooperative intrusion detection system for ad hoc networks. In *Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks* (pp. 135-147). https://doi.org/10.1145/986858.986877
[22] Abapour, N., Shafiesabet, A., & Mahboub, R. (2021). A novel security based routing method using ant colony optimization algorithms and RPL protocol in the IoT networks. *Mapta Journal of Electrical and Computer Engineering (MJECE), 3*(1), 1-9.
[23] Al-Khanjari, Z., Alanee, A., Kraiem, N., & Jamoussi, Y. (2013). Proposing a real time internal intrusion detection system towards a secured development of e-government web site. *European Scientific Journal*, 27-37.
[24] Latifi, T., Li, J., Blum, S. C., & Fowler, D. (2024). Determinants of Users' Intention to Visit a Destination: A Virtual Reality Quality Framework. *Journal of Quality Assurance in Hospitality & Tourism*, 1-25. https://doi.org/10.1080/1528008X.2024.2440010
[25] Azimi Asmaroud, S. (2022). *Preservice Elementary Teachers' Categorical Reasoning and Knowledge Transfer on Definition Tasks With Two Dimensional Figures* (Publication No. 1588) [Doctoral dissertation, Illinois State University]. Theses and Dissertations. https://ir.library.illinoisstate.edu/etd/1588
[26] Zhang, Q., Deng, R., Ding, K., & Li, M. (2024). Structural analysis and the sum of nodes' betweenness centrality in complex networks. *Chaos, Solitons & Fractals, 185*, 115158. https://doi.org/10.1016/j.chaos.2024.115158