Anomalous Link Detection in Dynamically Evolving Scale-Free-Like Networked Systems

Mehedi Hassan*, Mehmet Engin Tozal*, Vipin Swarup†, Steven Noel†, Raju Gottumukkala‡, Vijay Raghavan*
School of Computing and Informatics, University of Louisiana at Lafayette, Lafayette, LA 70504, USA*
The MITRE Corporation, McLean, VA 22102, USA†
Department of Mechanical Engineering, University of Louisiana at Lafayette, Lafayette, LA 70504, USA‡

***Abstract*—Networked systems are becoming increasingly complex and interconnected, making them more vulnerable to attacks. Identifying anomalous communication links is of paramount importance to ensure the integrity and security of a system. Existing methods based on static graphs only capture the interactions in networked systems using a single snapshot. Dynamically evolving graph-structured representations are gaining traction in modeling networked systems due to their ability to capture evolving relationships and complex interactions in time. In this study, we propose probabilistic approaches to predict future anomalous links in dynamically evolving scale-free-like networks. Specifically, we introduce three probabilistic models exploiting the scale-free property of networked systems to predict future link probabilities based on historical data. We conduct a grid search on the Receiver Operating Characteristic (ROC) curves of the models to determine the optimal decision boundaries for identifying anomalous links. We evaluate the performance of our proposed models on a synthetically generated dataset simulating a dynamically evolving real-world scale-free-like communication network. Our empirical results show that the proposed models' accuracy rates change between 63.20% to 96.50%, while the F1 score is between 66.25% and 96.52%. We also show that our approach can discern the direction of future links and estimate their probabilities based on link orientations.**

***Index Terms*—Anomalous link detection, link prediction, dynamic graphs, network security.**

I. Introduction

A networked system is a set of devices sharing resources using a communication medium [1]. The medium enabling communication among devices can be wired, wireless, or a combination of them. Besides, the devices in a networked system can range from personal wearables to IoT devices to smart home gadgets to computers and networking equipment. These networked systems can be found in the form of a Personal Area Network (PAN) or a Home Area Network (HAN) consisting of a set of personal or home gadgets operating together; a Local Area Network (LAN) consisting of a set of computers and equipment communicating in a small area such as a hospital building; a Metropolitan Area Network (MAN) consisting of a set of surveillance devices and networking equipment communicating within the boundaries of a city or a Wide Area Network (WAN) consisting of a set of equipment communicating across a region to enable the infrastructure for cellular telecommunications.

The hosts in these networks often have different roles, such as sensors that collect and forward data, networking equipment that routes data, computing devices that process data, database servers that store data, application servers that manage data and services, and client hosts that consume data and services [2]. While hosts may assume multiple roles, these roles often do not vary rapidly in time. Therefore, the hosts in such networked systems show predictable behavior from one-time interval to the next time interval [3].

Networked systems are often represented and studied as graphs, where each vertex (node) in the graph corresponds to a device in the network, and each edge (link) represents the communication taking place between two devices. Earlier studies found that many systems conform to the designation of scale-free networks such that the degree of a vertex is distributed by $P(k) \sim k^{-\gamma}$, where $2 < \gamma < 3$ [4]. More recent studies show that scale-free networks are not only rarer, but also their definitions are inconsistent among published works, *i.e.,* varying from "super-weak" scale-free to "strongest" scale-free [5]. Nevertheless, in this study, we focus on the type of networks that exhibit "scale-free-like" behavior such that (i) they consist of many vertices having lower degree and fewer vertices having higher degree; and (ii) higher degree vertices are not uncommon, *i.e.,* their degree distributions have heavier tails [6].

Many networked systems are designed with a client-server model in mind. Hence, they tend to exhibit scale-free-like behavior. That is, they consist of many service consumers (clients) and fewer service providers (servers) [7]. Note that some servers receive ancillary services from other servers while serving their clients. These systems often form a network of hub vertices and peripheral vertices, where the vertex roles do not change rapidly from one time interval to the next. Therefore, communication behaviors in such systems exhibit a predictable pattern from one time interval to the next.

In this study, we leverage the scale-free-like topology of networked systems along with the historical communication behaviors to identify anomalous communication patterns that may pose a threat to the system. In real-world systems, anomalous communication inceptions may correspond to (i) a disgruntled employee trying to access computers that he/she has never accessed before; (ii) a worm on a server trying to

spread to those client computers that have received a service in the past; and (iii) an internal perpetrator intermittently probing computers for known or unknown vulnerabilities. Identifying anomalous links in scale-free-like networks has applications beyond networked systems and security. Examples include, (i) connections between different proteins or genes that are not typically known to interact in protein-protein networks; (ii) an influx of links to the accounts of individuals or public figures in social networks; and (iii) a high number of links to seasonal products in recommendation networks. Nevertheless, our focus in this study is confined to the networked systems and their dynamically evolving graph representations in time.

There are many studies on anomaly detection in dynamic graphs [8]–[14]. However, a great portion of these studies either focus on detecting anomalous nodes and/or they are only applicable to undirected graphs. In this study, we propose three probabilistic models to predict future anomalous links in dynamically evolving scale-free-like networks using historical communications data. While lower link probabilities imply anomalous communication inceptions, higher link probabilities imply expected communications. Therefore, the proposed models can be employed in both anomalous link detection and expected link prediction tasks. Yet, the main objective of this study is anomalous link detection. These models are separately defined over undirected and directed graphs. We use a randomly selected training dataset consisting of different types of node pairs to learn and assign a probability to each node pair. Next, we conduct a grid search on the ROC curve of the training dataset to determine an optimal boundary to classify future links as anomalous or expected. Lastly, we present an algorithm that leverages the boundary and domain expertise to automate or semi-automate future link rejection and acceptance decisions in a network.

We experimentally demonstrate that our models achieve high accuracy, precision, and recall rates for both anomalous link detection and expected link prediction tasks. We conduct our experiments on a synthetic dataset that is provided by the MITRE Corporation. The dataset is generated using an AI model that processes a real-world network communication dataset and spins off its synthetic versions while adhering to its authentic characteristics. Our analysis in Section IV-C shows that the snapshots of the network conform to the "weak" and "weakest" designations of the scale-free networks [5]. Our empirical evaluations demonstrate that the proposed approaches reach up to an average accuracy of 96.50%, average precision of 95.87%, and average recall of 97.19% on directed graphs.

Note that the proposed models are developed for the systems that exhibit scale-free-like behavior as defined in [5]. Therefore, it is necessary to test scale-freeness of a networked system before applying the models.

The rest of the paper is organized as follows. Section II presents the related work. Section III explains the probabilistic models proposed for anomalous link prediction. Section IV describes the dataset and discusses the empirical results. Finally, Section V concludes our study.

II. Related Work

Existing anomaly detection approaches in scale-free networks employ node representations or embeddings. Feng et al. [15] propose a network embedding technique by implementing a "degree penalty" principle. CenGCN [16] captures the differentiation in information passing between vertices. Gu et al. [17] encodes graphs into low-dimensional vector representations for node centrality measurement and community detection. Lie et al. [18] introduce a Bayesian statistical model to enhance the identification of novel and unfamiliar anomalous nodes. Rahman et al. [19] develop a hybrid statistical approach to classify anomalous node behavior in social networks. These studies generate node embeddings using only static graphs.

Other studies use dynamic graphs for node anomaly detection. Bars et al. [8] develop a probabilistic framework for node-level anomaly detection. Ding et al. [9] propose a deep learning technique to detect anomalies in attributed networks. Tian et al. [10] introduce an encoder-based node anomaly detector. Heard et al. [11] develop a Bayesian model to detect anomalous nodes. While these methods are developed using dynamic graphs, they only detect anomalous nodes.

Fewer studies detect anomalous links in dynamic graphs. Xu et al. [12] propose sampling sub-graphs to detect suspicious adversarial edges. Cai et al. [13] leverage unusual sub-graph structures to detect anomalous links. Lo et al. [14] detect anomalous traffic in IoT networks. They utilize network traffic datasets where both normal and attack traffic are represented as labels in the network graph and do not consider the unobserved links. King and Huang [20] propose an intrusion detection framework to detect anomalous traffic involving network lateral movement. SEDANSPOT [21] is developed using random walk based edge anomaly scoring function to detect anomalous edge streams. Bhatia et al. [22] also develop an intrusion detection technique by detecting anomalous edge streams. Although these studies detect anomalous links, they employ node-neighborhood based or edge based sub-graphs, or classify attack traffic by employing edge labels.

Other studies use neural networks over directed graphs for link prediction. Gasteiger et al. [23] introduce directional message passing to leverage direction information. Zhang et al. [24] propose a neural network for directed graphs using magnetic Laplacian. Tong et al. [25] propose spectral based neural network for directed graphs. However, these studies do not detect anomalous links.

In this study, we propose a probabilistic approach to detect anomalous communication links in scale-free-like networked systems. Our study is different in two ways with respect to the existing approaches discussed above. First, our approaches only leverage node degrees to compute future link probabilities without requiring additional node features, edge features, or sub-graph sampling. Second, our developed model defined over directed graphs can predict the direction of the links.

III. Anomalous Link Prediction

Given that MITRE's network and other focused networks exhibit scale-free-like behavior, we take advantage of this

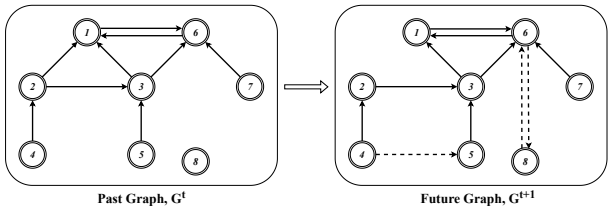

Fig. 1: An example of edges appearing and disappearing in a network between two time intervals (t and $t+1$).

property to estimate the link probabilities in the future based on past data. Employing these probabilities allows us to automate access control in a network domain by allowing the links with higher probabilities (expected) while rejecting the links with lower probabilities (anomalous).

Let G be a dynamic, simple graph representing a networked system changing in time. Let our current time be t such that the time interval between $t-1$ and t refers to the "past" and the time interval between t and $t+1$ refers to the "future". Let $G^t = (V^t, E^t)$ be the "past graph" representing the snapshot of G between time $t-1$ and t. The set of nodes or vertices, V^t, of graph G^t corresponds to the devices in the networked system between time $t-1$ and t. The set of links or edges, E^t, of graph G^t represents the communication taking place between pairs of devices in the networked system between time $t-1$ and t. Figure 1 illustrates an example of dynamically evolving nature of the networks. Solid edges represents repeated connections and dashed edges represent new connections forming in two consecutive time intervals. Also, connection between node pair (2,1) disappears from time t to $t+1$. *Our first goal is to estimate the likelihood of a link (communication) between any pairs of nodes (networked devices) in the future, i.e., from time t to $t+1$.* To achieve this goal, we develop probabilistic models that use the past graph $G^t = (V^t, E^t)$ to assign a probability to any future graph configuration $G^{t+1} = (V^{t+1}, E^{t+1})$. *Our second goal is to present an algorithm that consumes the estimated probabilities to decide whether the communication between two devices in the networked system is unexpected (anomalous), hence to be blocked or expected (ordinary), hence to be allowed.*

One important feature of the models developed in this study is that any communication attempt involving a device that has never been observed in the past is flagged as anomalous. To put in other words, links (edges) involving unseen nodes (vertices) should assume zero probability. We believe that such communication attempts should be blocked by default until the device and behavior is vetted by a security officer or systems administrator. Our belief is also coherent with the security principle of "Fail-Safe Permission Default" meaning that the default access control configuration should conservatively protect a system. In that vein, our models support the addition of new devices into the system, but require the intervention of a security officer or a systems administrator.

A. Models for Anomalous Link Prediction

Preferential attachment index [4] of a link between two nodes is defined as the multiplication of the degrees of the two nodes. The index simply implies that the well-connected nodes in a graph are expected to have more connections. In the following, we first develop a probabilistic model based on the preferential attachment index. Then, we introduce modifications of the model to improve probability estimations in undirected and directed graphs. Finally, we present an algorithm that integrates one of the probabilistic models to detect anomalous links and automate or semi-automate access control decisions in a network domain.

1) Model-A: The first model assumes that the probability of an edge between two vertices is higher, when the degrees of the induced vertices are higher. The probability of an edge $e_{i,j}^{t+1} \in E^{t+1}$ between v_i^{t+1} and v_j^{t+1} is estimated by Equation 1.

$$P(e_{i,j}^{t+1} \in E^{t+1}) = \frac{d_i^t d_j^t}{\sum_{k=1}^{|V^t|-1} \sum_{l=k+1}^{|V^t|} d_k^t d_l^t} \tag{1}$$

where d_i^t and d_j^t are the degrees of the vertices v_i^t and v_j^t in G^t. Note that the denominator of Equation 1 is computed only once for all probability estimations.

The vertices which have never been seen in the past take degree zero. Therefore, any of their edges assume zero probability in Equation 1. Coherent with the principle of "Fail-Safe Permission Default", such edges are flagged as anomalous until vetted by a security officer or systems administrator and their vertices introduced into the system.

2) Model-B: The first model assigns higher probabilities to the links between two higher-degree nodes compared to the links between a higher-degree and a lower-degree node. While this model is suitable for some scale-free networks, it does not necessarily fit the computer network domains. In a typical computer network, client nodes connect to various service nodes resulting in high-degree service nodes. However, often the service nodes are independent or partially dependent. Therefore, it is more suitable to treat the links between two higher-degree nodes as well as a higher-degree and a lower-degree node similarly. The probability of an edge $e_{i,j}^{t+1} \in E^{t+1}$ between v_i^{t+1} and v_j^{t+1} is estimated by Equation 2.

$$P(e_{i,j}^{t+1} \in E^{t+1}) = \frac{\max\{d_i^t, d_j^t\} \mathbb{1}_{V^{1..t}}(v_i^{t+1}) \mathbb{1}_{V^{1..t}}(v_j^{t+1})}{\sum_{k=1}^{|V^t|-1} (|V^t| - k)\, \pi_k^t} \tag{2}$$

where d_i^t, d_j^t are the degrees of the vertices v_i^t and v_j^t from the past graph, G^t, and π_k^t is the k^{th} highest degree node in the past graph G^t. Indicator functions $\mathbb{1}_{V^{1..t}}(v_i^{t+1})$ and $\mathbb{1}_{V^{1..t}}(v_j^{t+1})$ ensure that both v_i^{t+1} and v_j^{t+1} have been seen in the past, starting from the first day. The indicator functions ensure that Model-B is coherent with the security principle of "Fail-Safe Permission Default", *i.e.,* Equation 2 is zero for the edges involving unseen vertices. Note that the denominator of Equation 2 is computed only once for all probability estimations.

3) Model-C: While Model-A and Model-B are defined over undirected graphs, the availability of direction information distinguishes between a link from one node to the other and

its reverse. For example, communication data gathered at the transport layer of the TCP/IP protocol has a direction from client nodes, which "initiate" a connection, to service nodes. Note that a service node may also behave as a client node to receive some service from another node. Hence, Model-A and Model-B fail to capture the missions of the nodes in link prediction over data with direction information. When the direction information is available, the probability of an edge $e_{i,j}^{t+1} \in E^{t+1}$ from v_i^{t+1} to v_j^{t+1} is estimated by Equation 3.

$$P(e_{i,j}^{t+1} \in E^{t+1}) = \frac{d_j^{in^t} \mathbb{1}_{V^{1..t}}(v_i^{t+1}) \mathbb{1}_{V^{1..t}}(v_j^{t+1})}{|E^t|(|V^t|-1)} \quad (3)$$

where $d_j^{in^t}$ is the in-degree of the vertex v_j^t from the past graph, G^t and $|E^t|$ is the edge counts and $|V^t|$ is the vertex counts of the past graph, G^t. Indicator functions $\mathbb{1}_{V^{1..t}}(v_i^{t+1})$ and $\mathbb{1}_{V^{1..t}}(v_j^{t+1})$ ensure that both v_i^{t+1} and v_j^{t+1} have been seen in the past, starting from the first day. The indicator functions ensures that Model-C is coherent with the security principle of "Fail-Safe Permission Default", *i.e.,* Equation 3 is zero for edges involving unseen vertices. Note that the denominator of Equation 3 is computed only once for all probability estimations.

B. An Algorithm for Anomalous Link Decision

Since the computed probabilities are not calibrated, using 0.5 probability threshold as a decision boundary to automatically separate anomalous links from the expected ones does not work under any model. In this study, we use the Receiver Operating Characteristic (ROC) curve [28] of different threshold values to find the best decision boundary among many candidates. ROC curve plots TPR (True Positive Rate) against the FPR (False Positive Rate) at various boundary values. The optimal decision boundary is typically the one on the upper-left corner of the ROC curve. We use the optimization presented in Equation 4 to find the best decision boundary, b^* among a sequence of candidate values, b.

$$b^* = \arg\min_b |TPR(b) + FPR(b) - 1| \quad (4)$$

The best decision threshold requires a grid search between 0 and 1. In practice though, the search is conducted over an interval spanning from the minimum probability to the maximum probability reported by each model. The practical minimum and maximum probabilities are obtained while computing the denominators of Equations 1, 2, and 3.

Algorithm 1 requires a reference to the object representing one of the probabilistic models, M, that was run on the past graph G^t. In addition to the model, the algorithm requires the link to be tested, $e_{i,j}^{t+1}$, the maximum threshold, b^r for an automatic reject and the minimum threshold, b^a for an automatic accept. Algorithm 1 estimates the probability of the input link $e_{i,j}^{t+1}$ and fetches the optimal boundary at lines 1 and 2 respectively. As illustratively depicted in Figure 2, the conditions at lines 3 and 4, and lines 5 and 6 in Algorithm 1 automatically rejects or accepts the tested communication link. Lines 7 and 8 on the other hand, requires human intervention to investigate the test link.

Algorithm 1 Anomalous Link Decision

Require: M ▷ the object reference for one of the probabilistic models
Require: $e_{i,j}^{t+1}$ ▷ the communication link to be tested
Require: b^r ▷ link rejection threshold
Require: b^a ▷ link acceptance threshold
1: $p \leftarrow M.Estimate.Probability(e_{i,j}^{t+1})$
2: $b^* \leftarrow M.Optimal.Boundary()$
3: **if** $p < b^r$ **then**
4: Automatically reject link $e_{i,j}^{t+1}$
5: **else if** $p \geq b^a$ **then**
6: Automatically accept link $e_{i,j}^{t+1}$
7: **else**
8: Investigate link $e_{i,j}^{t+1}$ further
9: **end if**

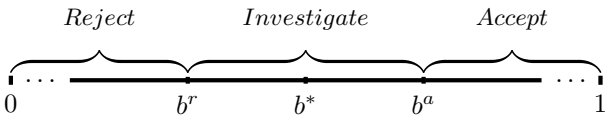

Fig. 2: An illustrative example of link decision boundaries.

In Algorithm 1, by default (i) a link is rejected if its probability is below b^r; (ii) a link is accepted if its probability is above or equal to b^a; and (iii) a link requires further investigation by a security officer or a system administrator if its probability is between b^r and b^a. Algorithm 1 looks like a simple algorithm, but one can enrich its behavior by controlling the threshold probabilities b^r and b^a. For example, when $b^r = b^a = b^*$ the system will be fully automated allowing any link with probability above or equal to b^* to be accepted and any link with probability below b^* to be rejected. The system will still be fully automated when $b^r = b^a < b^*$, but it will be more forgiving, *i.e.,* lenient towards accepting links. Similarly, the system will still be fully automated when $b^r = b^a > b^*$, but it will be stricter this time, *i.e.,* lenient towards rejecting links.

IV. Empirical Evaluations

In this section, we first describe our dataset and provide preprocessing details. Next, we present the designations of the scale-free property of our dataset. Then, we explain the experimental design of our study. Finally, we present the empirical evaluations the proposed models.

A. Description

Commercial enterprises are naturally reluctant to share their network topology or traffic flow information due to security concerns. On the other hand, working with realistic datasets is essential to understand the behavior of enterprise networks. In this study, we use a communication network dataset provided by the MITRE Corporation. The dataset is synthetically generated using an AI model that processes a real-world communication network and produces synthetic versions, while retaining the authentic characteristics of the original network flow data. The dataset spans over eight days and contains network flow information, including source IP

TABLE I: Day-wise statistics of the MITRE dataset.

Days	No. of Nodes	No. of Edges	No. of Unique Edges	Density	No. of Isolated Nodes
Day_1	1822	32197	1810	0.01808	487
Day_2	1822	31738	1811	0.01801	494
Day_3	1822	29614	1817	0.01675	492
Day_4	1822	28265	1818	0.01541	467
Day_5	1822	31618	1815	0.01813	501
Day_6	1822	29102	1816	0.01608	476
Day_7	1822	30038	1811	0.01710	496

address, destination IP address, starting and ending timestamps of the flows, destination port, and the protocol.

B. Dataset Preprocessing

Networked systems exhibit predictable behavior from one-time interval to the next. The duration of time intervals depends on the application domain. Using 24-hour or one-day time intervals is more suitable in our case. Hence, we divided the dataset into eight days based on the starting timestamps. We observed only four edges and eight nodes on Day_8. However, these connections appear right after midnight of Day_7. So, we ignore Day_8 and utilize the data from the first seven days. We present the day-wise statistics of the dataset in Table I. The number of edges represents multiple connections between two devices, while the number of unique edges represents distinct connections between two devices. Moreover, we found that roughly 500 nodes appear or disappear over time. Since these nodes are part of the original network, we add these nodes to our daily graphs as isolated nodes. Consequently, our network representation in each graph snapshot includes all nodes in the network over seven days. Hence, our models are capable to make predictions involving isolated nodes.

Next, we convert the daily network dataset into undirected and directed simple graphs for each day separately to obtain training and test datasets. We extract the edge list based on the observed or existing links from the graphs. Note that the number of possible node pairs is in quadratic order with respect to the number of nodes in the graph. However, only a small portion of node pairs have edges between them. The density of daily graphs indicates that only 1.5% - 1.8% out of all possible node pairs have links between them, while the remaining $\sim$98% of the pairs do not have links. Existing links are considered expected links, and non-existing links are considered anomalous links in training and test datasets. Furthermore, we include two types of anomalous links in our datasets, namely, two-way-missing links and one-way-missing links. A non-existing link between a node pair in both directions is considered as a "two-way-missing link". On the other hand, a non-existing link in the opposite direction of an existing link is considered as a "one-way-missing link". As the number of expected and anomalous links is significantly imbalanced, we use random sampling on the anomalous links (including both two-way and one-way-missing links) to balance the datasets. Hence, each dataset contains about 50% expected, 25% two-way-missing anomalous, and 25% one-way-missing anomalous links. We set the label of expected links to 0 and anomalous links to 1. Finally, the dataset generated from Day_N is used as the training dataset for the test dataset generated from Day_{N+1}, where $N = \{1, 2, 3, 4, 5, 6\}$.

TABLE II: Designations and criteria of scale-free property.

Designation	Criteria
Super-Weak	When no alternative distribution is favored over the power law in at least 50% of the nodes of a graph
Weakest	When a power-law distribution cannot be rejected ($p \geq 0.1$) for at least 50% of the nodes of a graph
Weak	When it satisfies the requirement of "Weakest" and the power-law region contains at least 50 nodes ($n_{tail} \geq 50$)
Strong	When it satisfies the requirements of the "Weakest" and "Weak", and the exponent γ falls within the range $2 < \gamma < 3$ for at least 50% of the nodes of a graph
Strongest	When all the requirements of "Strong" are met for at least 90% of the nodes of a graph, and "Super-Weak" for at least 95% of the nodes of a graph
Not Scale-Free	Neither "Super-Weak" nor "Weakest"

C. Scale-Free Property Test

We perform a statistical test developed in [5] to ensure that our dataset exhibits the scale-free property. There are six designations based on the scale-freeness of the networks: super-weak, weakest, weak, strong, strongest, and not scale-free [5]. Table II demonstrates the criteria for each designation. Note that these designations are nested. We test whether the daily graphs satisfy one of the designations of scale-free property. We found that Day_2, Day_3, and Day_6 conform to the "weak" designation at best, and the remaining four days conform to the "weakest" designation. Therefore, we conclude that our daily datasets exhibit scale-free-like behavior.

D. Experimental Design

First, we compute the link probabilities on the training (Day_1) dataset using Equations 1, 2, and 3 for Model-A, Model-B, and Model-C, respectively. Then, we perform a 10-fold cross-validation to obtain the optimal decision threshold using Equation 4. The training (Day_1) dataset is divided into train and validation sets. We use stratified k-fold cross validation [29] to balance expected and anomalous links in each fold. The best-performing threshold across 10-folds is selected as the optimal decision boundary. Next, we compute the link probabilities and apply the selected decision boundary to classify the anomalous links and expected links on the test (Day_2) dataset. Then, we present and discuss the classification results of the test (Day_2) dataset. We repeat the same process for all daily consecutive datasets That is, we compute the link probabilities and obtain the optimal decision boundary from the training (Day_N) datasets. Then, we apply the decision boundary and classify the links from the test (Day_{N+1}) datasets. Next, we present the evaluation metrics of our models across all test datasets. Finally, we summarize the performance of our models based on anomalous link detection and expected link prediction tasks. In the following, we present and analyze the empirical results of each model individually.

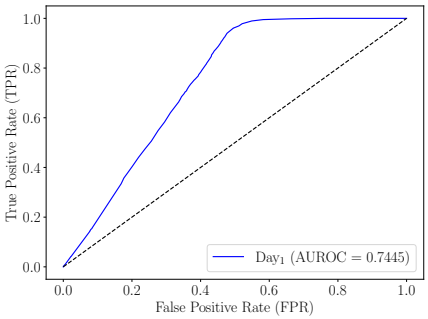

Fig. 3: AUROC scores of Model-A on training (Day$_1$) dataset.

TABLE III: Performance of Model-A on test (Day$_2$) dataset.

Accuracy	Precision	Recall	F1 Score
63.46%	61.50%	71.93%	66.31%

TABLE IV: Confusion matrix of Model-A on test (Day$_2$) dataset.

Actual	Predicted	
	Expected	Anomalous
Expected	996	815
Anomalous	508	1302

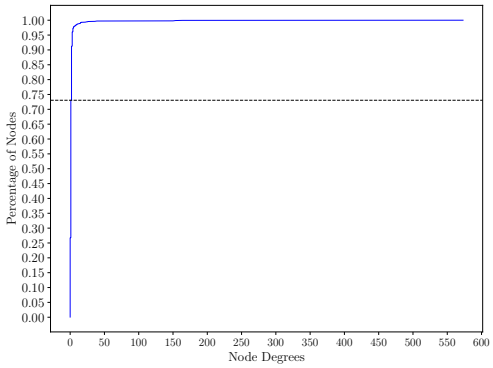

Fig. 4: ECDF for node degrees of (Day$_2$) undirected graph.

E. Performance of Model-A

AUROC scores on the training (Day$_1$) dataset reach up to 74% as depicted in Figure 3. We found that the optimal decision threshold is 2.27×10^{-5}. Then, we apply the decision threshold on the test (Day$_2$) dataset. Table III demonstrates that Model-A achieves an accuracy of 63.46%, precision of 61.50%, recall of 71.93%, and F1 Score of 66.31%. Next, we present the confusion matrix for the test (Day$_2$) dataset in Table IV to analyze the results. Model-A misclassifies almost half of the expected links as anomalous. We investigate the Empirical Cumulative Distribution Function (ECDF) of node degrees to gain more insights. Figure 4 depicts that more than 73% of the nodes have very low degrees (less than 3). Consequently, Model-A estimates a lower probability to the majority of expected links connecting to two lower-degree nodes and misclassifies them as anomalous. Next, we present day-wise evaluation metrics in Figure 5. Model-A maintains consistent performance across all remaining test datasets, where the difference is between 2 to 4 points.

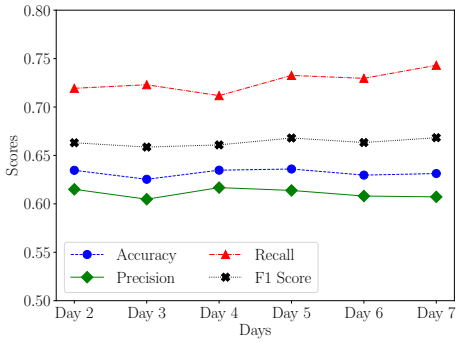

Fig. 5: Performance of Model-A across all test datasets.

TABLE V: Performance summary of Model-A on average across all test datasets.

Accuracy	Anomalous Link Detection			Expected Link Prediction		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
63.20%	61.10%	72.66%	66.37%	66.29%	53.74%	59.35%

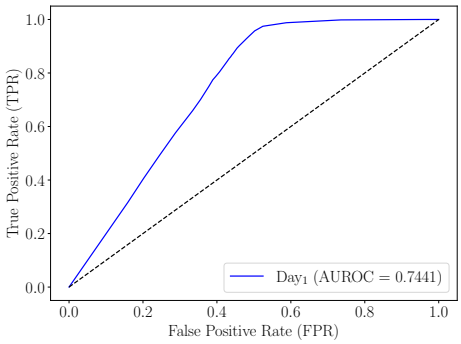

Fig. 6: AUROC scores of Model-B on training (Day$_1$) dataset.

Finally, we summarize the performance of Model-A across all test datasets in Table V. Model-A obtains an accuracy of 63.20% on average across all test datasets. In addition to detecting anomalous links, our proposed model demonstrates the capability to predict future expected links. Hence, we also present the results of the expected link prediction in Table V. Model-A achieves an F1 Score of 66.37% for anomalous link detection and 59.35% for expected link prediction tasks. Note that anomalous link detection is our primary goal, while the expected link prediction is a secondary outcome of our approach.

F. Performance of Model-B

Figure 6 depicts that AUROC scores of Model-B also reaches up to 74% on the training (Day$_1$) dataset. We found that the optimal decision threshold is 3.32×10^{-5}. Table VI demonstrates the results of the test (Day$_2$) dataset after applying the decision threshold. Model-B achieves an accuracy of 66.5%, precision of 66.95%, recall of 65.20%, and F1 Score of 66.06%. Next, we present the confusion matrix in Table VII. We observe that Model-B improves over Model-A in predicting expected links. This can be attributed to Model-B's ability to treat the links between two higher-degree nodes

TABLE VI: Performance of Model-B on test (Day_2) dataset.

Accuracy	Precision	Recall	F1 Score
66.50%	66.95%	65.20%	66.06%

TABLE VII: Confusion matrix of Model-B on test (Day_2) dataset.

Actual	Predicted	
	Expected	Anomalous
Expected	1238	573
Anomalous	620	1190

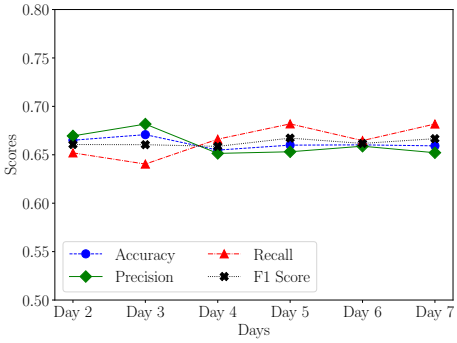

Fig. 7: Performance of Model-B across all test datasets.

TABLE VIII: Performance summary of Model-B on average across all test datasets.

Accuracy	Anomalous Link Detection			Expected Link Prediction		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
66.17%	66.11%	66.45%	66.25%	66.27%	65.89%	66.15

similarly to a higher-degree and a lower-degree node. Next, we present day-wise evaluation metrics in Figure 7. Model-B maintains consistent performance across all remaining test datasets, varying between 2 to 5 points only. Lastly, we summarize the performance of Model-B across all test datasets in Table VIII. Model-B achieves accuracy scores of 66.17% on average across all test datasets. Moreover, the F1 scores for anomalous link detection and expected link prediction suggest that Model-B performs similarly for both tasks. Furthermore, Model-B outperforms Model-A with a 3 points improvement in accuracy scores.

G. Performance of Model-C

Model-A and Model-B cannot capture the direction of the links as they are developed over undirected graphs. Model-C is developed to take advantage of the direction information. As depicted in Figure 8, Model-C achieves above 99% AUROC score. We found that the optimal decision threshold is 9.1×10^{-7}. Then, we present the results from the test (Day_2) dataset in Table IX. Model-C obtains a remarkable accuracy of 96.85%, precision of 96.54%, recall of 97.18%, and F1 Score of 96.86%. The confusion matrix presented in Table X shows that Model-C adeptly identifies both expected and anomalous links. Moreover, the number of misclassified samples has reduced significantly compared to other models. Figure 9

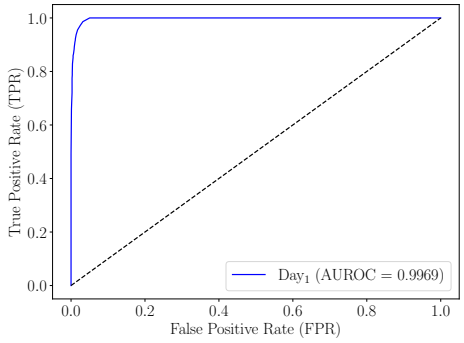

Fig. 8: AUROC scores of Model-C on training (Day_1) dataset.

TABLE IX: Performance of Model-C on test (Day_2) dataset.

Accuracy	Precision	Recall	F1 Score
96.85%	96.54%	97.18%	96.86%

TABLE X: Confusion matrix of Model-C on test (Day_2) dataset.

Actual	Predicted	
	Expected	Anomalous
Expected	1748	63
Anomalous	51	1759

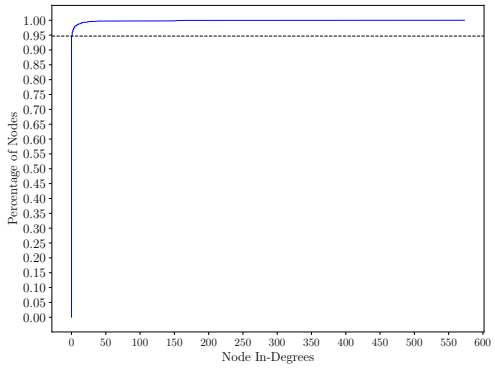

Fig. 9: ECDF for node degrees of (Day_2) directed graph.

shows that more than 94% of nodes have zero in-degrees based on the test (Day_2) dataset. Only 97 out of 1822 nodes receive connections from other nodes. As a consequence, only a few nodes have very high in-degrees, while the majority of the nodes have very low or zero in-degrees. Model-C takes advantage of this fact to accurately classify anomalous and expected links. Next, we present day-wise evaluation metrics in Figure 10. Accuracy, recall, and F1 scores are consistently more than 96%, while precision scores are over 95%. Lastly, we summarize the performance of Model-C across all datasets in Table XI. Model-C obtains accuracy scores of 96.50% on average across all test datasets, while the other two models achieve 63.20% and 66.17%, respectively. F1 scores for anomalous link detection and expected link prediction are also similar. Furthermore, Model-C outperforms other models with at least a 30 points improvement in accuracy scores.

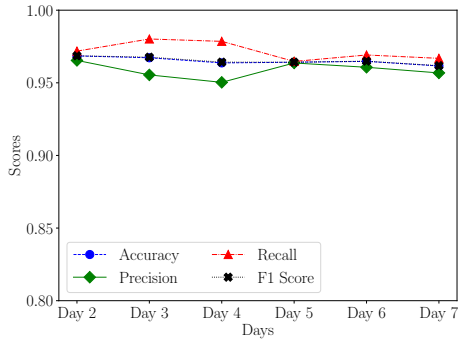

Fig. 10: Performance of Model-C across all test datasets.

TABLE XI: Performance summary of Model-C on average across all test datasets.

Accuracy	Anomalous Link Detection			Expected Link Prediction		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
96.50%	95.87%	97.19%	96.52%	97.16%	95.81%	96.48%

V. Conclusions

In this study, we proposed probabilistic approaches to detect anomalous links in dynamically evolving scale-free-like networks. We evaluated our models based on a synthetically generated dataset simulating a real-world communication network provided by the MITRE corporation. We divided the datasets into seven days with respect to the starting timestamps of the network flows. Next, we converted daily graphs into undirected and directed simple graphs. We incorporated one-way-missing links into our training and test datasets as anomalous links. These one-way missing links are a unique category of non-existing links in graphs, denoting connections that appear in the opposite direction of the existing links. Then, we developed three separate models, namely Model-A, Model-B and Model-C, defined over undirected and directed graphs. The empirical evaluations show that Model-C achieves an average accuracy of 96.50% across all test datasets. Furthermore, we showed that Model-C can successfully discerns the directions of the future links.

References

[1] M. S. M. Gismalla, A. I. Azmi, M. R. B. Salim, M. F. L. Abdullah, F. Iqbal, W. A. Mabrouk, M. B. Othman, A. Y. Ashyap, and A. S. M. Supa'at, "Survey on Device to Device (D2D) Communication for 5GB/6G Networks: Concept, Applications, Challenges, and Future Directions," *IEEE Access*, vol. 10, pp. 30 792–30 821, 2022.

[2] D. J. I. Z. Chen and L.-T. Yeh, "Data Forwarding in Wireless Body Area Networks," *Journal of Electronics and Informatics*, vol. 2, no. 2, pp. 80–87, 2020.

[3] J. Wang, S. Hao, R. Wen, B. Zhang, L. Zhang, H. Hu, and R. Lu, "IoT-Praetor: Undesired Behaviors Detection for IoT Devices," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 927–940, 2020.

[4] A.-L. Barabási and R. Albert, "Emergence of Scaling in Random Networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[5] A. D. Broido and A. Clauset, "Scale-free networks are rare," *Nature Communications*, vol. 10, no. 1, p. 1017, 2019.

[6] P. Holme, "Rare and everywhere: Perspectiveson scale-free networks," *Nature Communications*, vol. 10, no. 1, p. 1016, 2019.

[7] H. Fereidooni, A. Dmitrienko, P. Rieger, M. Miettinen, A.-R. Sadeghi, and F. Madlener, "FedCRI: Federated Mobile Cyber-Risk Intelligence," in *Network and Distributed Systems Security (NDSS) Symposium*, 2022.

[8] B. Le Bars and A. Kalogeratos, "A Probabilistic Framework to Node-level Anomaly Detection in Communication Networks," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 2188–2196.

[9] K. Ding, J. Li, R. Bhanushali, and H. Liu, "Deep Anomaly Detection on Attributed Networks," in *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 2019, pp. 594–602.

[10] S. Tian, J. Dong, J. Li, W. Zhao, X. Xu, B. Song, C. Meng, T. Zhang, L. Chen *et al.*, "SAD: Semi-Supervised Anomaly Detection on Dynamic Graphs," *arXiv preprint arXiv:2305.13573*, 2023.

[11] N. A. Heard, D. J. Weston, K. Platanioti, and D. J. Hand, "Bayesian anomaly detection methods for social networks," *The Annals of Applied Statistics*, 2010.

[12] X. Xu, H. Wang, A. Lal, C. A. Gunter, and B. Li, "EDoG: Adversarial Edge Detection For Graph Neural Networks," in *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 2023, pp. 291–305.

[13] L. Cai, Z. Chen, C. Luo, J. Gui, J. Ni, D. Li, and H. Chen, "Structural Temporal Graph Neural Networks for Anomaly Detection in Dynamic Graphs," in *Proceedings of the 30th ACM international conference on Information & Knowledge Management*, 2021, pp. 3747–3756.

[14] W. W. Lo, S. Layeghy, M. Sarhan, M. Gallagher, and M. Portmann, "E-GraphSAGE: A Graph Neural Network based Intrusion Detection System for IoT," in *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2022, pp. 1–9.

[15] R. Feng, Y. Yang, W. Hu, F. Wu, and Y. Zhang, "Representation Learning for Scale-free Networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[16] F. Xia, L. Wang, T. Tang, X. Chen, X. Kong, G. Oatley, and I. King, "CenGCN: Centralized Convolutional Networks with Vertex Imbalance for Scale-free Graphs," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[17] W. Gu, F. Gao, R. Li, and J. Zhang, "Learning Universal Network Representation via Link Prediction by Graph Convolutional Neural Network," *Journal of Social Computing*, vol. 2, no. 1, pp. 43–51, 2021.

[18] T. Liu, A. Qi, Y. Hou, and X. Chang, "Method for Network Anomaly Detection Based on Bayesian Statistical Model with Time Slicing," in *2008 7th World Congress on Intelligent Control and Automation*. IEEE, 2008, pp. 3359–3362.

[19] M. S. Rahman, S. Halder, M. A. Uddin, and U. K. Acharjee, "An efficient hybrid system for anomaly detection in social networks," *Cybersecurity*, vol. 4, no. 1, pp. 1–11, 2021.

[20] I. J. King and H. H. Huang, "Euler: Detecting Network Lateral Movement via Scalable Temporal Link Prediction," *ACM Transactions on Privacy and Security*, 2022.

[21] D. Eswaran and C. Faloutsos, "SedanSpot: Detecting Anomalies in Edge Streams," in *2018 IEEE International conference on data mining (ICDM)*. IEEE, 2018, pp. 953–958.

[22] S. Bhatia, R. Liu, B. Hooi, M. Yoon, K. Shin, and C. Faloutsos, "Real-Time Anomaly Detection in Edge Streams," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 16, no. 4, pp. 1–22, 2022.

[23] J. Gasteiger, J. Groß, and S. Günnemann, "Directional Message Passing for Molecular Graphs," *arXiv preprint arXiv:2003.03123*, 2020.

[24] X. Zhang, Y. He, N. Brugnone, M. Perlmutter, and M. Hirn, "Magnet: A Neural Network for Directed Graphs," *Advances in neural information processing systems*, vol. 34, pp. 27 003–27 015, 2021.

[25] Z. Tong, Y. Liang, C. Sun, D. S. Rosenblum, and A. Lim, "Directed Graph Convolutional Network," *arXiv preprint arXiv:2004.13970*, 2020.

[26] P. Erdős and A. Rènyi, "On Random Graphs," *Publ. Math*, vol. 6, pp. 290–297, 1959.

[27] E. N. Gilbert, "Random Graphs," *The Annals of Mathematical Statistics*, vol. 30, no. 4, pp. 1141–1144, 1959.

[28] J. A. Hanley and B. J. McNeil, "The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve." *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.

[29] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*. Springer, 2013, vol. 112.