Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2023.031522
Article

Tech Science Press

Combined Linear Multi-Model for Reliable Route Recommender in Next Generation Network

S. Kalavathi[1],[*] and R. Nedunchelian[2]

[1]Department of Information Technology, Sri Venkateswara College of Engineering, Sriperumbudur, Chennai, 602117, India
[2]Department of Computer Science & Engineering, Excel Engineering College, Komarapalayam, Namakkal, 638183, India
*Corresponding Author: S. Kalavathi. Email: kalavathi@svce.ac.in

Received: 20 April 2022; Accepted: 17 June 2022

Abstract: Network analysis is a promising field in the area of network applications as different types of traffic grow enormously and exponentially. Reliable route prediction is a challenging task in the Large Scale Networks (LSN). Various non-self-learning and self-learning approaches have been adopted to predict reliable routing. Routing protocols decide how to send all the packets from source to the destination addresses across the network through their IP. In the current era, dynamic protocols are preferred as they network self-learning internally using an algorithm and may not entail being updated physically more than the static protocols. A novel method named Reliable Route Prediction Model (RRPM) is proposed to find the best routes in the given hefty gage network to balance the load of the entire network to advance the network recital. The task is carried out in two phases. In the first phase, Network Embedding (NE) based node classification is carried out. The second phase involves the network analysis to predict the route of the LSN. The experiment is carried out for average data transmission and rerouting time is measured between RRPM and Routing Information Protocol (RIP) protocol models with before and after failure links. It was observed that average transmission time for RIP protocol has measured as 18.5 ms and RRPM protocol has measured as 18.2 ms. Hence the proposed RRPM model outperforms well than the traditional route finding protocols such as RIP and Open Shortest Path First (OSPF).

Keywords: Network embedding; node classification; link prediction; routing protocols; novel RRPM

1 Introduction

The mostly Distance and Link state protocols widely used to find static and dynamic paths. In the current extent of network based applications for text, image, audio and video data, Network science [1] is the most prominent and interesting field. Network science combines the concept of data science into the large scale of networks [2]. It should be required to efficiently process network data for future use. In that, the first and foremost challenging task is to treasure the effective Network Data Representation. Upon the completion of network representation concisely then the advanced analytical errands, such as outline unearthing, analysis of network [3], performance analysis [4], clustering of node [5], classification of node [6] and visualization of network, can be conducted efficiently in both time and space.

A novel NE has been developed to sustain the challenge and effort has been committed. Such a model is illustrated in Fig. 1. In this model, similar kind nodes labeled using ID and its closeness of each other node are coined as input with two dimensional spaces. It ensures and demonstrates that two dimensional embedding space and hence it is well modeled.

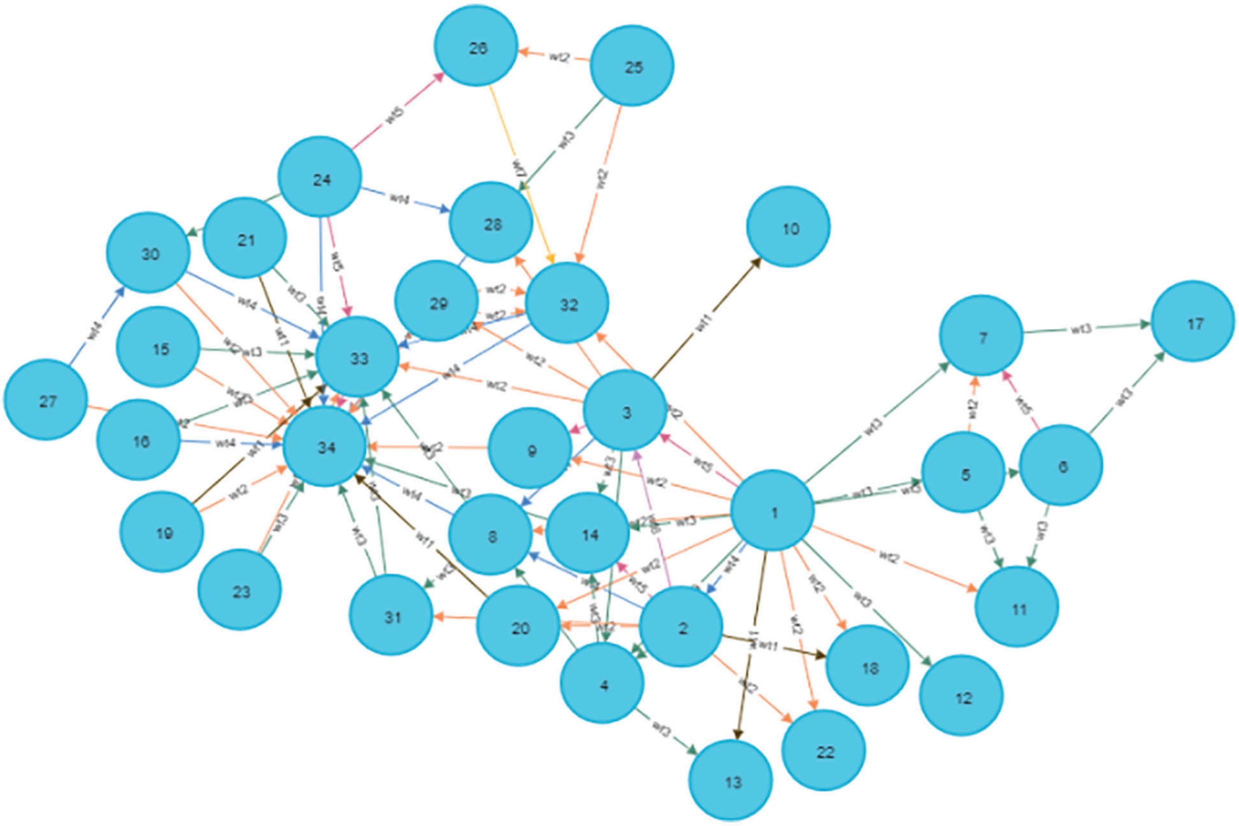

Figure 1: Input network graph

The networking processing tasks are as follows. Clustering and classification of nodes, prediction of links and visualization of networks. These tasks represent network depiction and it is called the NE. If NE was able to achieve the above processing steps, then it would be boycotting the traditional network process. Hence NE is deceptive and this strategy is the advantage of NE. The working steps of NE are illustrated in Fig. 2.

Fig. 2 represents the traditional topology based network analysis along with NE based network analysis and it compares each step and features of both. It is compared for image extraction [7] with network analysis. The deep and detailed survey [7] made on Summary for various methods of NE and Structure Preserving NE techniques [8]. Also surveys carried out on NE with side information and Advanced information preserving based NE. Before finding the reliable routes of the network, the network is analyzed using various networking parameters. Network parameters include node similarity, eccentricity, distance etc. The network topology can also be analyzed by these parameters.

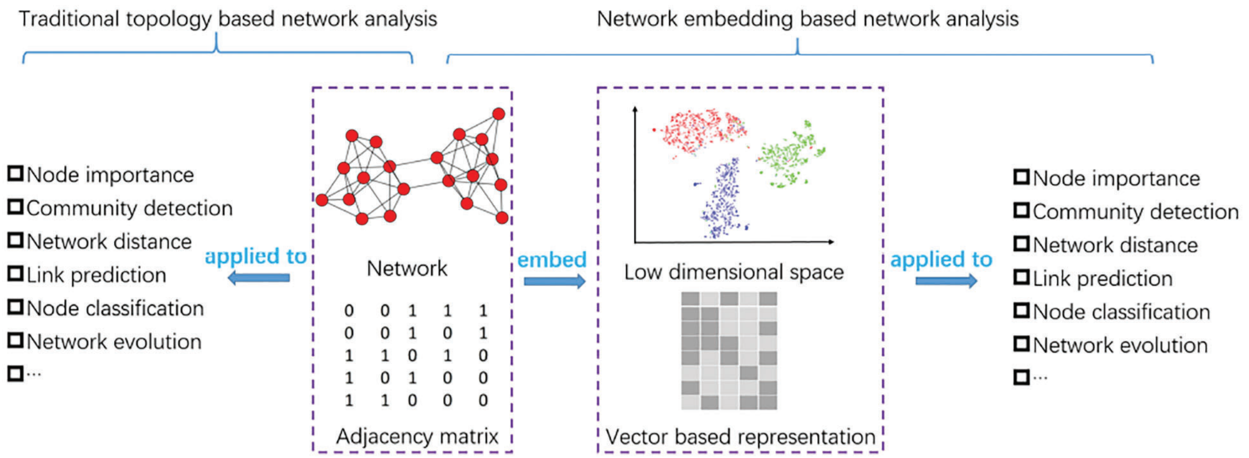

Figure 2: Comparison of tradition and NE based network analysis

The algorithms are applied on the network to perform various networking tasks such as network classification, link prediction, route analysis etc. Existing techniques focus on the static networks. The Field of Network Data science includes problems such as node classification and link prediction. Network node classification is a challenging domain in network science. The traditional algorithms are based on shortest path finding techniques. When the load in the static network increases, there is a necessity to find the alternate route to balance the network traffic. Any network with malicious nodes will pull down the performance of the network. Applying the traditional algorithms on such networks is too difficult.

The sampled nodes in the network or a network of interested nodes are called NE. NE so called as extreme learning where parameters need not be calculated and generated randomly. The output of NE is sent as input to node classification [4]. Logistic Regression (LR) and K–Nearest neighbor (KNN) classifier are used to classify the nodes as trusted and non-trusted nodes of the embedding network. LR with embedded graphs bounces improved results than both LR for node classification than label propagation on non-embedded graphs [9]. KNN outperforms efficiently than the label propagation algorithms such as Support Vector Machine (SVM) and back propagation algorithms.

The link prediction [6] algorithm deals with future predictions of the given network. Prediction of the missing links was carried out and it canister be used for building a Graph reconstruction, recommendation engine, and route prediction of a large scale network. RIP [10] with RRPM improves accuracy of the network and throughput of the network is measured in terms of routing parameters delay. The shortest path between source and destination nodes was found using RIP protocol. It finds the shortest path based on weights between nodes/hop counts. As RIP converges faster than OSPF, RIP with RRPM has to be measured as a research work as intelligence has to be established in network processing. In the next Gen networks, Dynamic route prediction is the most needed operation. To ensure dynamism in the next Gen Network route findings, a new method was proposed RRPM which obtains the network embedding Topology2vec [11] through a Machine Learning (ML) algorithm logistic regression using and works with RIP.

Further this research work is discussed as follows. Section 2 discusses the related work on NE and Routing information. Section 3 is discussing the proposed RRPM algorithm model and its related methods and algorithms. Section 4 is depicting the result and discussion of the experiment. Finally Section 5 is summarizing the conclusion and future work of this research work.

2 Related Works

In this section, it is essential to discuss previous work done in information embedding based on large networks and the technical background. A brief introduction about the issues of representation of network and also the related work in the representation learning of the network information has to be discussed. Secondly, discuss the ML tasks like prediction and classification in the learned network. Thirdly, it is necessary to investigate some related works for the routing problems and its application scenario for how and why it is essential.

In ML is learning, which is essential to study about the structure of topological networks as a typical method. NE or network representation learning techniques are attractive methods [12,13]. The original high dimensional spaces are the Learns latent features and then converted. The concept of a rather sample was used for NE [14]. In this work, the nonlinear characteristics of networks were taken as a challenge as it is difficult to handle due to nonlinearity. Also it was addressing, pair of nodes for multiple-order relationships.

Some exciting work has been carried out on methods such as deep walk [15] and work for the language processing area which is inspired by word2vec [16]. In this research work, statistical information on graph unbiased random walk was carried out and this walk treated information as a set of sentences. Initially the language modeling method is set to learn the network representation and it acts as a generalized language modeling method. The Node2vec [17] was introduced for performing Depth First Search (DFS) and Best First Search (BFS) as deep walk and it has been improvised with feature and weight of DFS and BFS. Other than DeepWalk, the method is called Node2vec enabling two hyper parameters and is carried out in DFS and BFS. Another work was discussed for second and first order proximity, both simultaneously, to reserve a simple objective function LANE [18]. In this work, Topology2Vec was coined for the DCN network environment to act as a well-designed representation learning method and it preserves global view of the topology between nodes using the local structures.

Topology2Vec [11] satisfies both efficiency and validity for Topology Zoo [19], in terms of average network latency when compared with k–means ML algorithm. Methods such as DeepWalk, LINE, and Node2vec time complexity was measured and it is actually NE methods. On the other hand, GraRep [20–22], M-NMF, and LANE, methods imposed for quadratic complexity, may cause the limit applied to the large networks and scalability issues too. Thirdly, Liblinear [23] classifier is used to train and learn from the training set.

It was inferring the labels of the rest nodes that the trained classifiers were popularly applied to evaluate the metrics for multi-label classification problems. The classification includes Macro-F1 and Micro-F1 [24]. A high significance was achieved for a variety of applications with NE and has been proven, examples were node classification [4], graph visualization [25], recommendation tasks [26], and link prediction [27]. The four categories of data sets tested for multi-label classification applications have been successfully tested. Some names were social networks citation based applications. The BLOGCATALOG was used for evaluation of performance along with the methods like DeepWalk [15], node2vec [17], GraRep [20], SDNE [21], and LANE [22]. Finally it was concluded that NE algorithms have been widely used to catch performance on various networks. It has been well demonstrated their effectiveness using above methods on node classification. Some time, heterogeneous type of network's information can be used to measure personalized entities [28].

Manageable amount of attention is impacted on link prediction [27–30] for network analysis since it is one of the fundamental problems. In observed network structure [31], two nodes edge connection existence has been measured and also has to be estimated its likelihood and its observation. Liben-Nowell and Kleinberg [27] work stated the link prediction problem. It stated that the scale-free network was discovered and its feature time t and it is represented in Eq. (1).

$$\mathrm{tm} = \mathrm{t} + \mathrm{t} \tag{1}$$

The node centrality scores used for link prediction as the traditional graph-based features by different supervised and unsupervised approaches derive from the common neighborhood [27–29]. The work is that two nodes may gain higher propensity which has to be linked as similarity. In general, Mean Average Precision (MAP) and K mean precision measuring ML algorithms were used to evaluate the link prediction performance [14]. Link prediction on a social network such as Facebook and a biological network PPI were measured using Node2vec. The Database systems and Logic programming was demonstrating the effectiveness of citation networks using EOE [32]. Epinions and Slashdot, and SiNE [33] showed that the superior performance of signed NE for two social networks and it was measured as link prediction. It is expensive with respect to parameter measure computationally and spatially. Thus, it is essential to deploy a Random Forest (RF) classifier as an alternative. RF can carry out a new algebraic operation to obtain the symmetric pairwise feature representation. RF can use Topology2vec for a node pair. Yet another work was discussed about action recommenders for networks with ML concept [34]. This work was represented to solve the issue of Internet service Provider (ISP) to minimize the cost of network data transmission. Here an experiment was carried out to measure user quality and ISP cost. Another network related concept for optimizing an efficient network, a Drone based DOA beamforming antenna was discussed [35]. This work is essential to achieve high output for data transmission.

3 Proposed Work

A network G is a collection of nodes N and edges E and it is represented as Eq. (2).

$$G = \{N, \; E\} \tag{2}$$

A network can be viewed as a directed or undirected graph. LSN of nodes are created dynamically with various networking parameters. In General, all the networking parameters are not interesting in nature and also are time consuming if we process the entire large network. A process called NE is applied on the network to extract features of the network. A well-known random walk algorithm for a large network called Node2Vec/Topology2vec is applied on a dataset such as karate club, wiki sets and extracted the interesting nodes for down streaming tasks such as link prediction and node classification. Reliable or trusted nodes of the network are classified and extracted using linear regression. The KNN classifier is a better suited algorithm for non-embedded networks than the label propagation algorithms. Classifier accuracy, micro and macro F1 of linear regression is slightly higher than the label propagation such as SVM and back propagation. Traditional link prediction coefficients such as jaccard and adamic adar index. The newly learned reliable network is applied with link mining algorithms such as Node2vec and Topology2vec. Topology2vec with link prediction produces better results than traditional correlation of Jaccard's coefficient and state-of-the-art legacy algorithm Node2vec.

A Completely learned reliable trusted network is fed as input to the novel method called RRPM. Routing using RIP with and without RRPM is analyzed. The delay and throughput of the network on various trials were being measured. RIP with RRPM performs better. The Flow diagram of the proposed work is shown in Fig. 3.

3.1 Proposed Algorithms

The methods that were used for simulating a random walk to generate node contexts and it was for the statistical sampling-based approaches from a given network G (V, E), where V and E are the set of vertices and edges respectively. Afterward, the node classification function is invoked and it is set as the initialization phase. The procedure for node classification is coined as Algorithm 1.

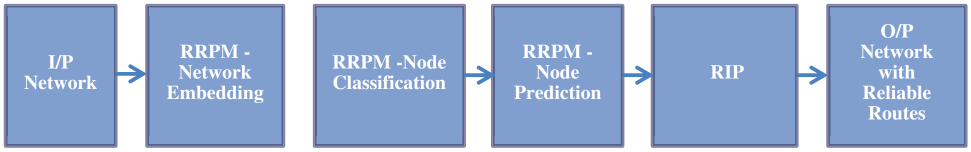

Figure 3: Flow diagram of the proposed work

Algorithm 1: Embedded, Predicted and Reliable route of Each Individual Node in Graph G

```
Input: G = {N, E} (Adjacency list data structure of the Graph.)
Output: G {Reliable Routes from source S to Destination D}
Procedure RRPM (Graph G, Adjacency list, Node Parameters)
NodeList←NodeClassification(G)
for ∀vi ∈ V do
    templist = neighbour(vi):
    ∃NAscore (neighbour(vi))> = τ, visited[vi] = 1
    for k ← 0 :Average_degree do
        Graphcontext = Context (templist, visited)
    end for
EVector(Individual_Score vi)← Graphcontext (skip-gram model)
FList ← EVector(SimkranHadamard(vi, vj) [])
Return list_of_feature_combination(s)
```

Further the node classification is expanded which is coined as Algorithm 2 and it is a classification procedure.

Algorithm 2: To Produce Classified Nodes of the Input Network

```
Input: Graph (G)
Output: F1 score and Accuracy Score
Procedure NodeClassification (G)
G ← Graph (Edge list)
Emodel ← Topology2Vec, Node2Vec (G)
Embedlist ← Emodel.getembeddings()
for Embed in Embedlist
    evaluate Embedding(Embed)
End for
Return Accuracy Score, F1 score
```

Then the context function is coined as Algorithm 3 is invoked from Algorithm 1. Input to the skip-gram model for learning the embedding vectors using the contextual nodes procedure are given as of individual nodes. The context function scores [33] is being calculated using Neighborhood Affinity (NA) and

Subgraph Affinity (SA) for producing contextual nodes. The output of RRPM is given as input to RIP. RIP with RRPM produces the reliable routes from the source to destination node. Hence RRPM has become a Novel RRPM.

Algorithm 3: Produce the Graph Context of a Source and Destination Node in a Network

```
procedure Context(templist, visited)
    Gclist = templist
    for$\forall$vj, vj$\in$templist and $\neg$visited[v j]: do
        Taugmentlist = neighbour (v j):
        $\exists$NAscore(neighbour(v j))> = $\tau$, visited[v j] = 1
        Augmentlist = $\exists$Taugmentlist:
        SAscore(Taugmentlist) > = $\tau$
        append(Gclist, Augmentlist)
    end for
    return Gclist
end procedure
```

3.2 Proposed RRPM Model

The approach of learning latent low-dimensional features is referred as NE representations for the links or the nodes in a network for the downstream ML tasks. For any graph given, the Node2vec algorithm can learn for the nodes as continuous feature representations. It is further used for various up and downstream ML tasks. Again Node2Vec was used to calculate node embedding. To ensure that nodes are embedded to learn in such a way and which are close in the graph. Remaining close is the embedding space with Node2Vec.

The concept of second order random walks, however, tries to model the transition probability based on the currently visited node v, the node t visited before the current one, and the node x which is the target of a candidate relationship. Node2Vec random walks are thus influenced by two parameters: the returnFactor and the inOutFactor.

A network with 34 nodes of Zachary's karate club network data set is taken for algorithm implementation. Topology2vec is applied on the network and obtained 128 dimensions for the down streaming process. Only interested nodes are computed based on the node embedding outputs. The python packages networkx and numpy for the simulation gensim are required. This approach is coined as Algorithm 4.

Algorithm 4: Topology2Vec arguments

```
Input: Node1, Node2, Weight (two columns of data, weights in the third column)
Output: num_of_nodesdim_of_representation, node_id, Embedded Vector
Graph G = {Directed|Undirected Network)
Initialize p, q(Default to 1), Noofwalks, walk_length, Edim
```

where dim1, … , dimd is the d-dimensional representation learned by node2vec.Node2vec produces embedding vectors. The vector is then visualized in two dimensional spaces. karate club network and weighted Input Network is represented as Graph after extraction from NetworkX package for network embedding process.

3.3 Node Classification

An important applied task in the area of Network Data Science with Node Classification. It could be viewed as a Graph in a Network as nature. Networkx or Nodex and Neojs are useful in representing a network as the Graphs. The Produced Network Graph is then can be trained by a model to learn in which class a node belongs. Pre-processing algorithms include Node Embedding such as Node2Vec, Centrality algorithms, and auxiliary algorithms. These preprocessing algorithms convert the Graph to Vector Representation which is the required format for down streaming functions.

Fig. 4 depicts the Route from Source S to Destination D is S->4->5->D. If traffic passes through node 5 leads to packet drop due to non-reliability of the node 5. Before recommending the route, RRPS finds the reliable routes. For finding the reliable route, Node classification has to be carried out using combined linear models.

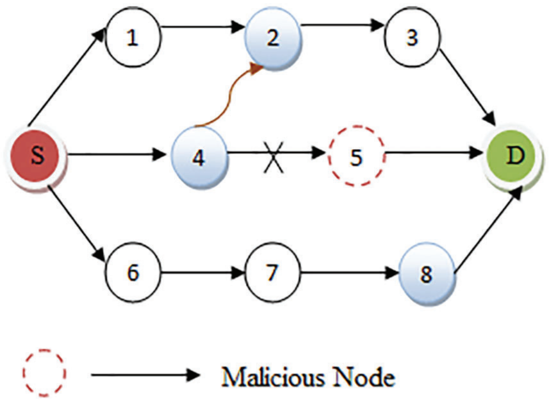

Figure 4: Rerouting due to identification of malicious/non reliable nodes

Both Node Classification and Link Prediction have training parameters that can be tuned automatically given a set of allowed values. The parameters maxEpochs, tolerance and patience control for how long the training will run until termination. These parameters give ways to limit a computational budget. In general, higher maxEpochs and patience and lower tolerance lead to longer training but higher quality models. It is however well-known that restricting the computational budget can serve the purpose of regularization and mitigate overfitting.coefficient = 0.0. Coefficient is calculated with the help of Eq. (3). Here α and δ are constant.

$$\text{coefficient} = \text{coefficient} - (\alpha * \delta) \tag{3}$$

Fig. 5 shows the 10 nodes of 3 Class labels. 3 unclassified nodes classified into 3 class labels.

The combined Linear Models such as Logistic Regression and SVM algorithm is applied to a pre-processed network. Node classification can be done as a downstream task from node representation learning or NE, by training a supervised or semi-supervised classifier against the embedding vectors. The Embedding vector is obtained using the Node2vec algorithm. Fig. 6 shows the classification of reliable and non-reliable nodes of the network. A Network is classified with class 0 non reliable and class 1 reliable nodes. The Classification algorithm applied on the network and classification accuracy of the network is obtained.

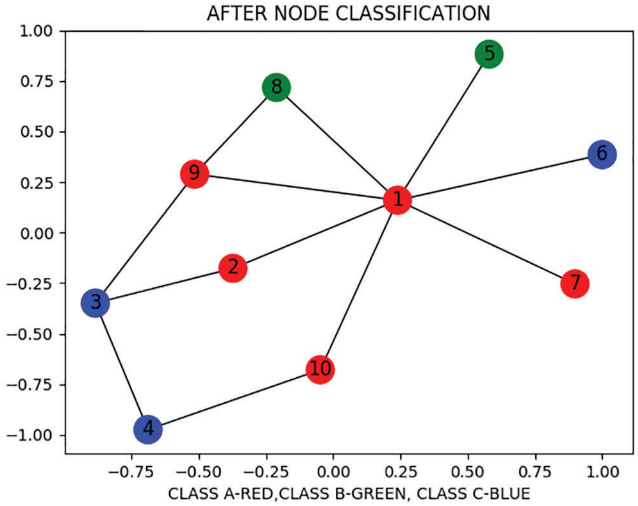

Figure 5: After node classification

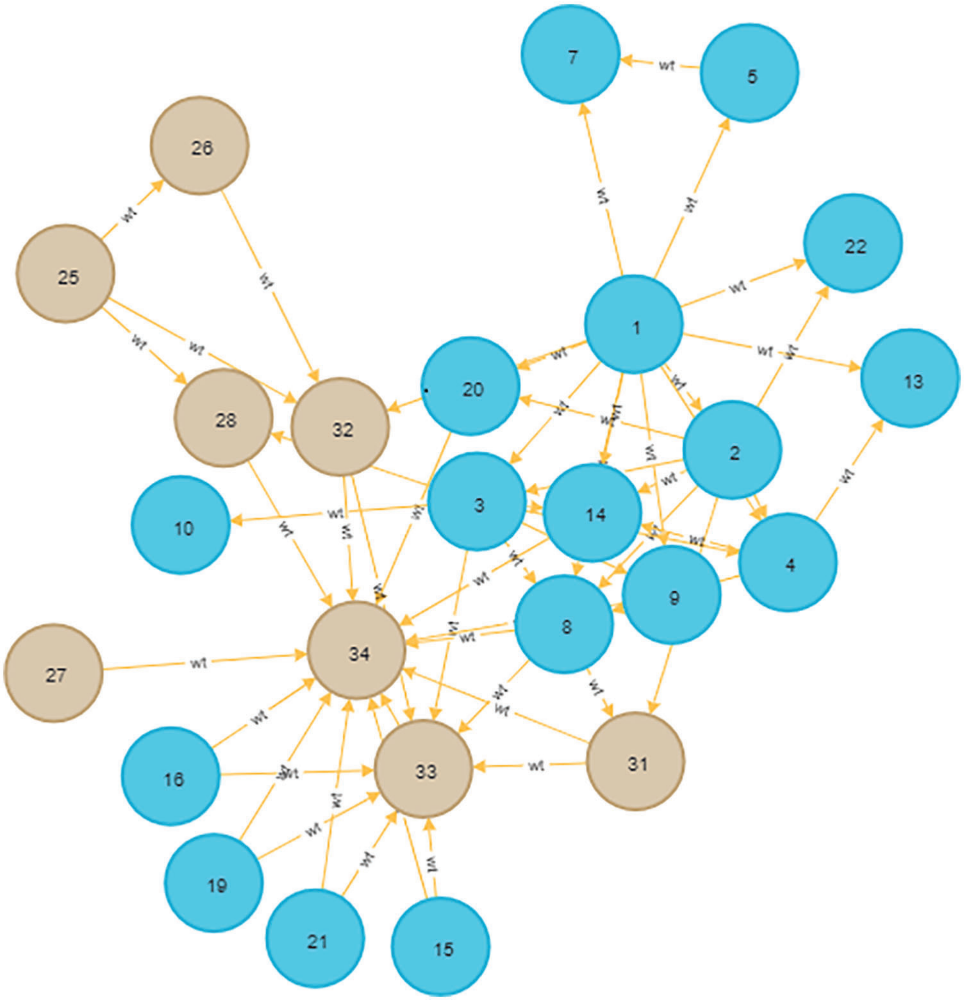

Figure 6: Training network with reliable and non-reliable nodes (class1 and class0 network)

Out of 34 nodes, 24 nodes are considered for the training phase and 10 for test nodes. The algorithmic set up is mentioned below. The algorithm is set with various penalty values. With the penalty of 0.05, the model performed well. Time taken to train the LR model is considered, Fig. 6 shows the classification model outcome of training graph.randomSeed: 2, holdoutFraction: 0.2, validationFolds: 5, params: {penalty: 0.05}, metrics: ['F1_WEIGHTED'].

The measures of the classification model are mentioned in Tab. 1. It shows the measures like nodeProjection, nodeCount, relationshipCount and createMillis for training model and Model time measurement.

Table 1: Train the model time measure

NodeProjection	NodeCount	RelationshipCount	CreateMillis
{"id":{"properties":{},"label":id}}	34	79	19

It is observed that penalty 0.0625 performed for the model candidate with the best in the training phase. Further it is measured with a score of almost 100% over the train graph as shown in Tab. 2. The model scores a bit lower at about 64% if used in the test graph. This signposts that to the train graph, the model reacted very well. It was able to unseen data well when generalized fairly.

Table 2: The winning model with varying penalty

Winning model	Train graph score	Test graph score
{"penalty":0.05}	0.98921428569	0.4674999997

Tab. 3 is depicting the predicted class, probability confidence of tested graph and it is run till 30 nodes.

Table 3: Predicted class, probability confidence of test graph

Classified node ID	Predicted class	Predicted probabilities	Confidence
25	0	[0.36358798726248853, 0.6364120127375109]	63
26	1	[0.15490441396376228, 0.8450955860362375]	84
27	0	[0.36358798726248853, 0.6364120127375109]	63
28	1	[0.15490441396376228, 0.8450955860362375]	84
29	0	[0.36358798726248853, 0.6364120127375109]	63
30	1	[0.15490441396376228, 0.8450955860362375]	84
31	0	[0.36358798726248853, 0.6364120127375109]	63
32	1	[0.15490441396376228, 0.8450955860362375]	84
33	0	[0.36358798726248853, 0.6364120127375109]	63
34	1	[0.15490441396376228, 0.8450955860362375]	84

3.4 Link Prediction

Firstly the karate club Network is integrated with the Neo4j and NetworkX packages. The java based Neo4j Class Query Language (CQL) was useful in creating a new network dynamically or importing and integrating with the existing datasets and represented as a graph. Neo4j, data is represented by nodes and relationships between those nodes. The network analysis of various parameters was carried out. The advantage is that it can store many interconnecting relationships as data in neo. It is a simple, powerful and flexible and highly scalable data model for Graph Representations. Also the model ensures vertical scaling, concurrency and tuning of various parameters. A well-known Neo schema graph database created with integrated karate club data. Link prediction algorithms based on node with multiple attributes and it is represented in Tab. 4.

Table 4: Multiple attributes node for link prediction algorithm

Jaccard coefficient	(19, 15)- > 1
	(8, 21)- > 0.6666666666666666
	(32, 28)- > 0.5
Adamic-adar index	(19, 15)- > 0.7553857282466059
	(8, 21)- > 0.7553857282466059
	(32, 28)- > 0.35295612386476116

The link prediction algorithm finds the average score and number of relationships written. Also the model is computed with a train and test graph score of 95% with the average penalty of 0.5 as shown in Tab. 5.

Table 5: Link prediction model

Score	Relationships written	Winning model	Train graph score	Test graph score
0.352956	162	{"maxEpochs":1000,"penalty":0.5}	0.641199	0.605384

4 Evaluation and Results

In the given network, Initially RIP and RRPM are configured in each trial. The size of test packets is fixed as 1000 byte. Both the models produced the shortest path of the routes as shown in Fig. 7.

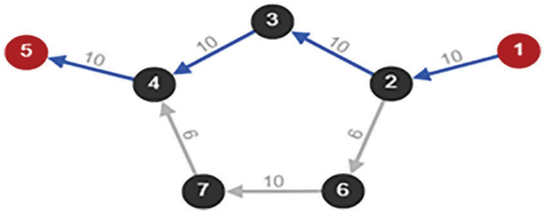

Figure 7: Before a link fails of transmission path *x*

The transmission path and time of both the models is measured when there is no link failure in the network on multiple trials. Single link failure is considered and transmission path is recorded for the link failure as shown in Figs. 8 and 9.

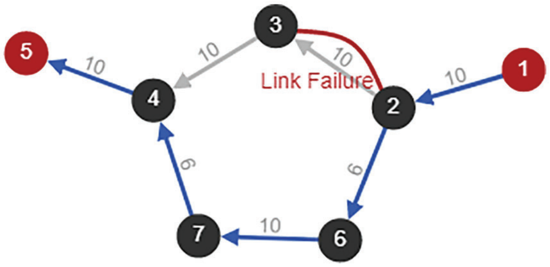

Figure 8: After a link fails of transmission path finding

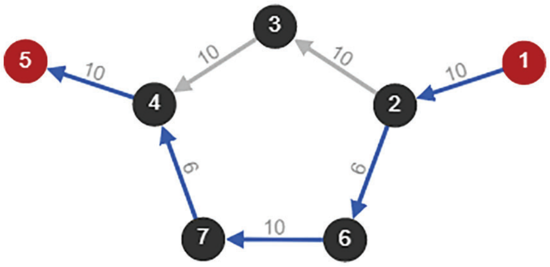

Figure 9: After a link fails of transmission path found

In the experimental setup, the two routing protocols such as RIP and RRPM use the same path in data transmission for the single link failure as shown in Tab. 6.

Table 6: The route of transmission

Directions of nodes before link failure	Directions of nodes use after link fail
1–2	1–2
2–3	2–6
3–4	6–7
4–5	7–4

When the node is identified as a malicious node in the network, the transmission path is chosen as shown in the following Fig. 10 and complete the path in Fig. 11.

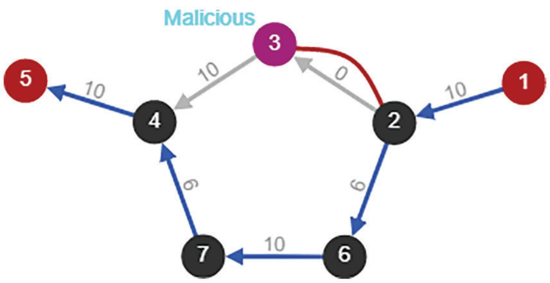

Figure 10: Transmission paths due to malicious node

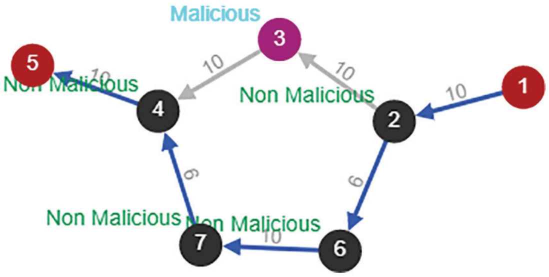

Figure 11: Complete transmission paths

RIP, RRPM models experimented on the weighted karate datasets of 34 nodes and the transmission time of the network is measured on each trial as shown in Fig. 12. Graph depicted in Fig. 12 compares link fail and no link fail.

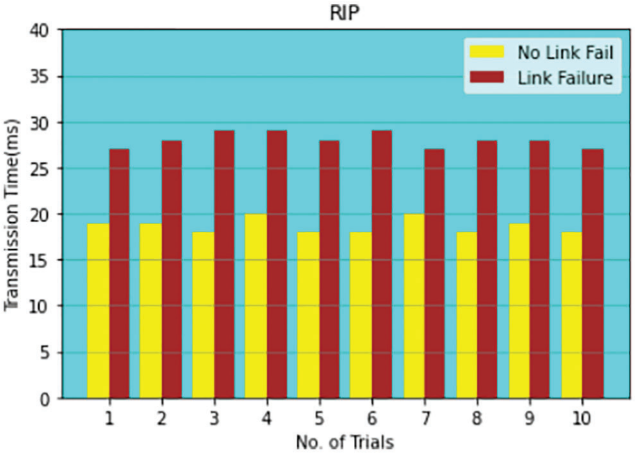

Figure 12: Comparison of link failure and no link failure for 30 nodes

The experiment is carried out for both RIP and RRPM algorithms. The results are illustrated with graph as Fig. 13. The Inferences are measured with the average data transmission time. Before the link fails, RIP protocol's average transmission time is 18.5 ms whereas 18.2 ms for RRPM protocol and hence RRPM is considerably faster than RIP. Similarly after the link fails, transmission time is calculated for both algorithms. The RRPM protocol's packet average data transmission time is measured as 26.5 ms. Whereas in RIP protocol's packet average data transmission time is measured as 28 ms. The inference was noted when the transmit data path is based on the same route and it is for both before and after the link fails. Standard deviation is measured with a sample and noted that before the link fails the RIP is 0.813 ms and RRPM is 0.799 ms. like that after the link fails to RIP is 0.806 ms RRPM is 0.704 ms. If it is compared with other existing algorithms like OSPF and RIP [14–30], RRPM out performed as this work coined Algorithm 1 to Algorithm 4. The transmission time is comparatively less for this proposed RRPM method and hence this work has introduced a novel method.

Also important note pointed out that the average data transmission time before and after link fails. Observation claimed that there is a slight difference between RRPM protocol model and RIP protocol model. Next, the rerouting time is calculated for finding a new transmission path using RRPM protocol, which is shorter than RIP protocol for reliability. From the experimented results, it is noted that the RRPM protocol has produced considerably better performance than RIP protocol. The convergence time of dynamic routing protocols with single/multiple link failure and the activities single/multiple link recovery and with RRPM represented in Figs. 14 and 15 respectively.

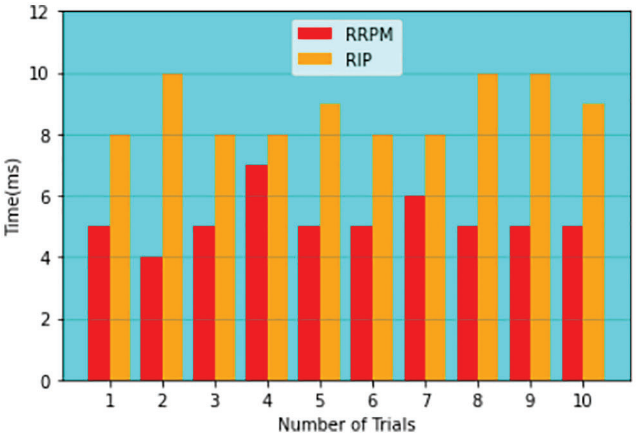

Figure 13: Comparison of RRPM and RIP model for average data transmission

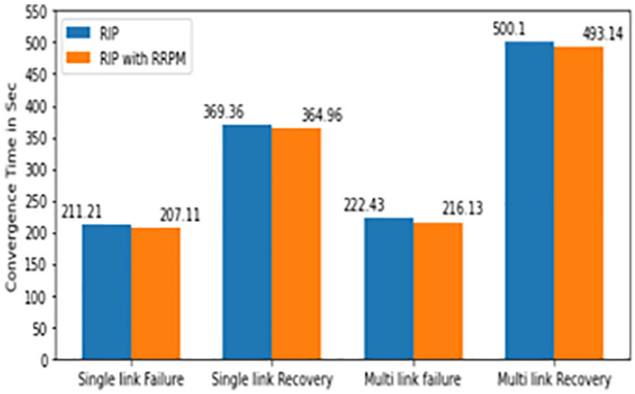

Figure 14: RIP protocol with RRPM

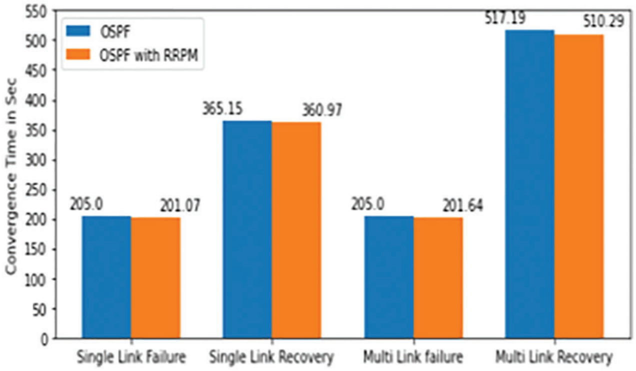

Figure 15: OSPF protocol with RRPM

The summary of the notable events in the multiple links failure/recovery scenario is has given below. The exported values for discussion are also presented in Tabs. 7 and 8. This proposed work is only compared with RIP and OSPF, but as technology of routing protocol has enhance for WSN, so this proposed work is also extended with some WSN based protocol and its applications.

Table 7: Routing protocols convergence time for RIP and OSPF

Protocol	Type of link(s) failure	Convergence time (in Sec)		
		No link failure	Link failure	Link recovery
RIP		10.78		
RIP	Single		211.21	369.36
RIP	Double		222.43	500.1
OSPF		53.06		
OSPF	Single		205.1	365.15
OSPF	Double		205.1	517.19

Table 8: Routing protocols RIP and OSPF with RRPM model

Protocol	Type of link(s) failure	Convergence time (in Sec)		
		No link failure	Link failure	Link recovery
RIP with RRPM		10.78		
RIP with RRPM	Single		207.11	364.96
RIP with RRPM	Double		216.13	493.14
OSPF with RRPM		53.06		
OSPF with RRPM	Single		201.07	360.97
OSPF with RRPM	Double		201.64	510.29

5 Conclusion and Future Work

RRPM performed well when compared with traditional single source routing algorithms. The operations in RRPM can be increased to approach various issues of the network. It was observed that convergence time of RIP is faster (11.21 s) than OSPF (53.06 s) at the initial stage of the network as there is no link failure or recovery RIP would be preferred if the network is not prone to frequent network failure as the convergence time is higher than the OSPF. But the convergence time of RIP would be improved if the hybrid model was implemented. The average convergence time on multiple trials is improved by a considerable rate. The performance of hybrid model RRPM with various types of protocols such as measured RIP and OSPF was measured. The applications of this proposed work may be introduced in duplicate video comparison video surveillance system [36], authenticated audio video transmission system [37], automated network traffic based stock price prediction system [38] and highband based network for vehicle identification system [39,40].

In future, this work could be extended to automate the node, link and network failure detection using DL algorithm. Also this proposed work would be extended as Failure Detection System (FDS) and Failure Prevention System (FPS).

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflict of interest to report regarding the present study.

References

[1] Z. M. Fadlullah, F. Tang, B. Mao, N. Kato, O. Akashi *et al.,* "State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2432–2455, 2017.

[2] N. Bui, M. Cesana, S. A. Hosseini, Q. Liao, L. Malanchini *et al.,* "A survey of anticipatory mobile networking: Context-based classification, prediction methodologies, and optimization techniques," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1790–1821, 2017.

[3] X. Kong, Y. Shi, S. Yu, J. Liu and F. Xia, "Academic social networks: Modeling, analysis, mining and applications," *Journal of Network and Computer Applications*, vol. 132, pp. 86–103, 2019.

[4] N. M. Balamurugan and S. Appavu, "Performance analysis of ad-hoc on demand distance vector routing protocol," *Indian Journal of Science and Technology*, vol. 9, no. 4, pp. 1–8, 2016.

[5] M. Sowmiya and M. Adimoolam, "Secure cloud storage model with hidden policy attribute based access control," in *Proc. of Fourth IEEE Int. Conf. on Recent Trends in Information Technology, Madras Institute of Technology*, Anna University, Chennai, pp. 1–6, 2014.

[6] L. Wang, J. Ren, B. Xu, J. Li, W. Luo *et al.,* "MODEL: Motif-based deep feature learning for link prediction," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 2, pp. 503–516, 2020.

[7] C. Peng, X. Wang, J. Pei and W. Zhu, "A survey on network embedding," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 5, pp. 833–852, 2018.

[8] X. Huang, J. Li and X. Hu, "Label informed attributed network embedding," in *Proc. 10th ACM Int. Conf. Web Search Data Mining*, New York, NY, USA, pp. 731–739, 2017.

[9] S. Kumar, S. Chaudhary, S. Kumar and R. K. Yadav, "Node classification in complex networks using network embedding techniques," in *Proc. 5th Int. Conf. on Communication and Electronics Systems (ICCES)*, Coimbatore, India, pp. 369–374, 2020.

[10] R. Rajyashree and N. M. Balamurugan, "Mobility management in power controlled and rate adaptive ad hoc networks," *International Journal of Engineering Research & Technology*, vol. 7, no. 3, pp. 311–314, 2018.

[11] X. Zhenzhen, L. Hu, K. Zhao, F. Wang and J. Pang, "Topology2vec: Topology representation learning for data center networking," *IEEE Access*, vol. 6, pp. 33840–33848, 2018.

[12] N. M. Balamurugan and S. Balamurugan, "Graph and sinr based interference modeling and routing in manet," *International Journal of Advanced Engineering Technology*, vol. 7, no. 2, pp. 290–294, 2016.

[13] W. L. Hamilton, R. Ying and J. Leskovec, "Representation learning on graphs: Methods and applications," *IEEE Data Engineering Bulletin*, vol. 3, pp. 1–24, 2017.

[14] D. Wang, P. Cui and W. Zhu, "Structural deep network embedding," in *Proc. KDD*, New York, NY, USA, pp. 1225–1234, 2016.

[15] R. M. Bose and N. M. Balamurugan, "Improving the invulnerability of wireless sensor networks against cascading failure," *Lecture Notes on Data Engineering and Communications Technologies*, vol. 35, pp. 870–876, 2020.

[16] M. Adimoolam, M. Sugumaran and R. S. Rajesh, "A novel efficient clustering and secure data transmission model for spatiotemporal data in WSN," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 8, pp. 117–125, 2018.

[17] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining (KDD)*, New York, NY, USA, pp. 855–864, 2016.

[18] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan *et al.,* "Line: Large-scale information network embedding," in *Proc. 24th Int. Conf. World Wide Web*, Republic and Canton of Geneva, CHE, pp. 1067–1077, 2015.

[19] S. Knight, H. X. Nguyen, N. Falkner, R. Bowden and M. Roughan, “The internet topology zoo,” *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 9, pp. 1765–1775, 2011.

[20] S. Cao, W. Lu and Q. Xu, “Grarep: Learning graph representations with global structural information,” in *Proc. of the 24th ACM Int. on Conf. on Information and Knowledge Management*, New York, NY, USA, pp. 891–900, 2015.

[21] X. Shen and F. Chung, “Deep network embedding for graph representation learning in signed networks,” *IEEE Transactions on Cybernetics*, vol. 50, no. 4, pp. 1556–1568, 2020.

[22] X. Huang, J. Li and X. Hu, “Label informed attributed network embedding,” in *Proc. 10th ACM Int. Conf. on Web Search Data Mining*, New York, NY, USA, pp. 731–739, 2017.

[23] M. Adimoolam, M. Sugumaran and R. S. Rajesh, “A novel efficient redundancy free data communication model for intelligent surveillance system in WSN,” *Journal of Advanced Research in Dynamical and Control Systems*, vol. 10, no. 3, pp. 743–754, 2018.

[24] M. Adimoolam, G. Karthi, A. John, M. Senthilkumar, A. Ali *et al.,* “A hybrid learning approach for the stage-wise classification and prediction of covid-19 x-ray images,” *Expert Systems*, vol. 39, no. 4, pp. 1–15, 2021.

[25] M. Adimoolam, K. Hemachandran, R. Prasanth, N. Prabavathi and M. T. Mathew, “Protected multi-tiered sensor network uses spatiotemporal feature,” *International Journal of Advanced Research in Computer Science*, vol. 4, no. 2, pp. 124–127, 2013.

[26] M. Adimoolam, A. John and N. M. Balamurugan, “Green ICT communication, networking and data processing,” *Green Energy and Technology*, Cham: Springer, pp. 95–124, 2021.

[27] M. Adimoolam, M. Sugumaran and R. S. Rajesh, “The security challenges, issues and countermeasures in spatiotemporal data: A survey,” *Lecture Notes on Data Engineering and Communications Technologies*, Cham: Springer Publications, vol. 26, pp. 1216–1224, 2018.

[28] X. Yu, X. Ren, Y. Sun, Q. Gu, B. Sturt *et al.,* “Personalized entity recommendation: A heterogeneous information network approach,” in *Proc. 7th ACM Int. Conf. on Web Search Data Mining*, New York, NY, USA, pp. 283–292, 2014.

[29] B. Gallagher and T. Eliassi-Rad, “Leveraging label-independent features for classification in sparsely labeled networks: An empirical study,” in *Proc. Int. Workshop on Social Network Mining and Analysis*, Berlin, Germany, Springer, pp. 1–19, 2008.

[30] J. Wang, Y. Ma, M. Liu and W. Shen, “Link prediction based on community information and its parallelization,” *IEEE Access*, vol. 7, pp. 62633–62645, 2019.

[31] L. Getoor and C. P. Diehl, “Link mining: A survey,” *ACM SIGKDD Explorations Newsletter*, vol. 7, no. 2, pp. 3–12, 2005.

[32] L. Xu, X. Wei, J. Cao and P. S. Yu, “Embedding of embedding (eoe): Joint embedding for coupled heterogeneous networks,” in *Proc. 10th ACM Int. Conf. on Web Search Data Mining*, New York, NY, USA, pp. 741–749, 2017.

[33] S. Wang, J. Tang, C. Aggarwal, Y. Chang and H. Liu, “Signed network embedding in social media,” in *Proc. SIAM Int. Conf. on Data Mining*, Houston, United States, pp. 327–335, 2017.

[34] S. A. Mohammed, A. R. Mohammed, D. Côté and S. Shirmohammadi, “A Machine-learning-based action recommender for network operation centers,” *IEEE Transactions on Network and Service Management*, vol. 18, no. 3, pp. 2702–2713, 2021.

[35] N. M. Balamurugan, M. Senthilkumar, M. Adimoolam, A. John, G. T. Reddy *et al.,* “DOA tracking for seamless connectivity in beamformed IoT-based drones,” *Computer Standards & Interfaces*, vol. 79, pp. 103564, 2022.

[36] N. M. Balamurugan, T. K. S. Rathish babu, M. Adimoolam and A. John, “A novel efficient algorithm for duplicate video comparison in surveillance video storage systems,” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–15, 2021. https://doi.org/10.1007/s12652-021-03119-7.

[37] M. Adimoolam, A. John and M. Gunashanthi, “Anti-piracy for movies using forensic watermarking,” *International Journal of Computer Application*, vol. 63, no. 4, pp. 26–32, 2013.

[38] A. John, D. Praveen, M. Adimoolam and N. M. Balamurugan, “Prediction strategies of stock market data using deep learning algorithm,” *Recent Advances in Computer Science and Communications*, vol. 14, no. 6, pp. 1852–1859, 2021.

[39] W. Sun, G. C. Zhang, X. R. Zhang, X. Zhang and N. N. Ge, “Fine-grained vehicle type classification using lightweight convolutional neural network with feature optimization and joint learning strategy,” *Multimedia Tools and Applications*, vol. 80, no. 20, pp. 30803–30816, 2021.

[40] W. Sun, X. Chen, X. R. Zhang, G. Z. Dai, P. S. Chang *et al.,* “A Multi-feature learning model with enhanced local attention for vehicle re-identification,” *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3549–3561, 2021.