

Deep Reinforcement Learning Based on Link Prediction Method in Social Network Analysis

T. Manjunath Kumar, R. Murugeswari

Abstract: Improving the performance of link prediction is a significant role in the evaluation of social network. Link prediction is known as one of the primary purposes for recommended systems, bio information, and web. Most machine learning methods that depend on SNA model's metrics use supervised learning to develop link prediction models. Supervised learning actually needed huge amount of data set to train the model of link prediction to obtain an optimal level of performance. In few years, Deep Reinforcement Learning (DRL) has achieved excellent success in various domain such as SNA. In this paper, we present the use of deep reinforcement learning (DRL) to improve the performance and accuracy of the model for the applied dataset. The experiment shows that the dataset created by the DRL model through self-play or auto-simulation can be utilized to improve the link prediction model. We have used three different datasets: JUNANES, MAMBO, JAKE. Experimental results show that the DRL proposed method provide accuracy of 85% for JUNANES, 87% for MAMABO, and 78% for JAKE dataset which outperforms the GBM next highest accuracy of 75% for JUNANES, 79% for MAMBO and 71% for JAKE dataset respectively trained with 2500 iteration and also in terms of AUC measures as well. The DRL model shows the better efficiency than a traditional machine learning strategy, such as, Random Forest and the gradient boosting machine (GBM).

Keywords: deep reinforcement learning; social network analysis; gradient boosting machine.

I. INTRODUCTION

Research identified with graph theory and network analysis in the field of sociology [1] spearheaded the improvement of social network analysis (SNA) approaches. Graph theory gives the applied develops, strategies, and procedures in analyzing the graph. The systems and strategies of criminal network analysis (CNA) [2] are basically received and adjusted from SNA scientific models and measurements [3]. Some social organizations, for example, Facebook and recommender frameworks, present potential companion's dependent on a SNA link prediction method [4].

SNA metrics used in the evaluation of links can be obtained from a network measure. Neighborhood-based metrics, Katz-based metrics, and random walk-based metrics are the approaches used. The measures based on the neighborhood include the common neighbor, the Jaccard index, and the

Revised Manuscript Received on December 16, 2019.

* Correspondence Author

T.ManjunathKumar*, Computer Science and Engineering, Kalasalingam Academy of Research and Education, KrishnanKoil, India. Email: manjunathkumar.t@klu.ac.in

Dr.R.Murugeswari, Computer Science and Engineering, Kalasalingam Academy of Research and Education, KrishnanKoil, India. Email: r.murugeswari@klu.ac.in

Adamic-Adar index [5].

Deep reinforcement learning (Fig. 1) method which includes deep learning and Reinforcement Learning (RL) learning approach for machine learning techniques [6-9]. Function approximation is a supervised learning technique used in the areas of machine learning, artificial neural networks pattern recognition, and statistical learning [10-12].

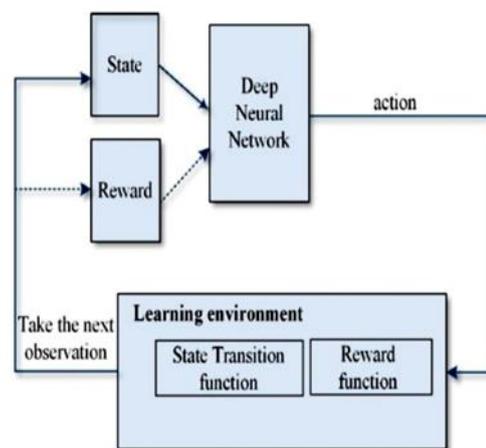


Fig. 1. Deep Reinforcement Learning

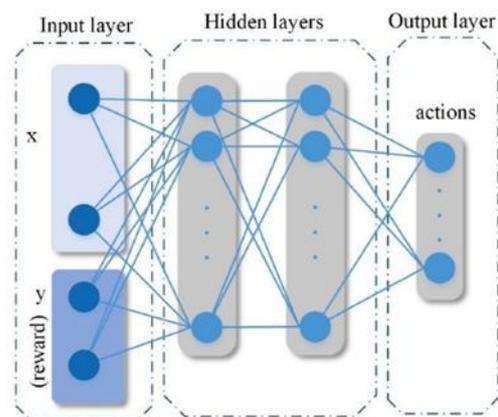


Fig. 2. Deep Neural Network

Deep Neural Network (fig. 2) have been implemented into RL and have obtained significant success in approximating value function. the first deep Q-network (DQN) algorithm that effectively combines a sophisticated nonlinear function approximation method (i.e., DNN) with the Q-learning algorithm. RL is a method in ML where programs, classified as agents, combine with the environments in which they adapt and learn on the basis of points given after completion of the job. positive if the points are labeled as rewards or negative if the points are labeled as

punishment. But supervised learning approach is different from the reinforcement learning approach of machine learning, which doesn't use training phase in the given dataset.

The paper structure is organized as follows: Related works in the field of link prediction in Social Network Analysis based machine learning techniques are reviewed in section 2. In section 3, the proposed DRL-based Link prediction model and algorithm are presented. In section 4, the validation of the work with dataset evaluation measure and Results are explained. Finally, Conclusion with future work is given in section 5

II. RELATED WORK

Emrah Budur et al. [13] directed research on connection expectation of criminal systems with the use of SNA models and measurements dependent on AI procedures. They analyzed the AI strategies utilizing the GBM managed learning technique to upgrade the exhibition of concealed connection forecast in a huge dataset. A test was directed with measurable models and likelihood connect expectation measurements got from the inspecting of the criminal system dataset to evaluate the presentation of the administered connection forecast model built by the GBM calculation. The example size utilized in the examination comprises of roughly 1.5 million hubs and four million undirected edges, which speak to an altogether bigger dataset size contrasted with test sizes utilized in other research papers. The bigger example size utilized was relied upon to all the more likely speak to the genuine attributes of criminal systems. The utilization of a huge size of test dataset additionally exhibited that the connected under-examining procedure improved the exactness of the inspecting technique received in this exploration, as teamed up by the superior of the prepared connection expectation model, shown by the registered AUC metric. The AUC metric of an AI model shows the prescient exactness of the model and has a worth that lies somewhere in the range of 0 and 1, where a higher worth demonstrates a superior prescient precision.

The essentially huge size of the criminal system test dataset utilized in the exploration, contrasted with the other research papers overviewed [14], was likewise examined, and logical measurements that are progressively precise in profiling the genuine qualities of criminal systems were given. Bellemare et al. [15] examined the advancement of DRL calculations utilizing current figuring equipment details, central processing units (CPUs) and Graphical Processing units (GPUs), explicitly NVIDIA DGX-1. They adjusted both arrangement slope and Q-learning calculations in AI models, utilizing on parallel, synchronous, and no concurrent processing procedures. In their paper, two profound RL calculation procedures are distinguished. Policy gradient technique trained on on-policy measurements and Q-value learning process involving off-policy training, including Deep Q-Networks [16], trained using GPU.

Thomas Anthony et al. [17] consolidated learning strategies with RL calculations with regards to the round of Hex to improve the exhibition of the RL calculations. The exploration was led in reference to the fundamental research deal with DRL by D. Silver et al. [18] in the improvement of a program fit for making the appearance of Go, named

AlphaGo, in the field of counterfeit general insight. David Silver et al. made a critical leap forward in their exploration on DRL [19], where the program they created, AlphaGo Zero, was, through DRL, prepared to do completely aching the antiquated key round of Go without anyone else's input play. The program figured out how to achieve a superhuman exhibition by overcoming the top grandmasters of the game from South Korea and China. The AlphaGo program was effectively fabricated utilizing a trailblazing DRL procedure of joining a neural system and RL. AlphaGo at first uses a neural net that is just coded with the fundamental principles of the round of Go. It at that point mimics a huge number of self-made diversions against different adaptations of itself.

The learning picked up from this self-play is then nourished back to the neural system to recalibrate the hyper-parameters. The recalibrated neural system is then used to improve the productivity of the Monte Carlo tree search (MCTS) calculation by filling in as a function approximator. The presentation of AlphaGo Zero is accordingly expanded negligibly in every cycle. Nonetheless, through a great many self-play involvement in multi day, the program could reproduce the amassing of many years of human learning of Go and performed like a top-level grandmaster in just three days.

While information in the fields of both the neural system and RL was at that point known before the advancement of AlphaGo Zero, the way to the jump in the exploration by David Silver et al. was because of the novel structure of the calculation and huge improvement in the GPU parallel registering power. The advancement of the DRL system initially includes the design of a neural system to foresee the best move in a given position and after that to tweak this neural system with RL through self-play with various forms of the program. This neural system, which establishes the arrangement system of AlphaGo, was observed to be fit for beating the most progressive unadulterated play MCTS Go programs 85% of the time. This was an exceptionally critical leap forward contrasted with the 10% that was recently accomplished.

The exploration was then kept on improving the exhibition by joining MCTS into the approach organize. Notwithstanding, the strategy system performed too gradually in the assessment of each move expanding from the great many potential results in the recreation of the game move outs figured with MCTS. Along these lines, a moment worth system was prepared utilizing the information created through self-play; this could assess the results quicker, i.e., the take-off system. The worth system could assess a position quicker dependent on the evaluated likelihood of the game being won from a particular position. The consolidated arrangement system is just utilized once to acquire a gauge on an underlying move with the most astounding winning likelihood. The quicker take off strategy system is then executed to cross rapidly utilizing MCTS, through the numerous potential parts of moves expected to achieve a definitive result of the game.

This method improves the determination of the underlying move for a hunt calculation in a recreation as opposed to by means of arbitrary choice and afterward influences on the quick MCTS calculation to finish the take-off of every reenactment. The key reason MCTS works proficiently for the round of Go is that it expels the troublesome need of coding human-characterized

assessment calculations for explicit examples of positions in the game. Therefore, the AlphaGo calculation includes a worth system that is equipped for anticipating moves with the most noteworthy winning likelihood, which is then assessed by means of recreation move outs utilizing the MCTS arrangement organize. The blend of the neural systems brought about a colossal jump in execution contrasted with the individual execution of each neural system. To streamline the AlphaGo execution, the program was then conveyed utilizing eight GPUs for the neural system parallel figuring. The hunt strings of MCTS were executed on 48 CPUs.

Table – I: Link prediction metrics [13].

Metrics	Definition
Common Neighbour	$S_{xy} = \varphi(x) \cap \varphi(y) $
Jaccard Index	$S_{xy} = \frac{ \varphi(x) \cap \varphi(y) }{ \varphi(x) \cup \varphi(y) }$
Hub Index	$S_{xy} = \frac{ \varphi(x) \cap \varphi(y) }{\min(K_x, K_y)}$
Preferential attached Index	$S_{xy} = (K_x \times K_y)$
Adamic-Adar Index	$S_{xy} = \sum_{z \in \varphi(x) \cap \varphi(y)} \frac{1}{\log K_z}$
Katz	$S_{xy} = \sum_{t=1}^{\infty} \beta^t \cdot n_{xy}^{(t)}$

III. PROPOSED METHOD

Deep reinforcement learning method includes both deep learning and RL learning approach for machine learning techniques. Hence, the efficiency of DRL for link prediction of the model relies on a few variables and is assessed on the basis of SNA metrics (Table- I) and DRL techniques formulation.

Predicting the chance of the occurrence or rejection of links within a model is formulated as a binary classification issue using a supervised learning method. In this work, the criminal dataset has been used which will be converted into the Graph database as a data preprocessing. In the graph network, edge between the two nodes is represented as the links. E denotes the set of all possible edges. E+ denotes positive edges and E- denotes negative edges. Positive edges are defined as the edge between two nodes in the graph network. Negative edges are defined as the no edge between the particular pair of nodes in

the graph network. It means absence of edge is a negative edge. In the feature Metrix selection, SNA metrics is used and using this metrics Link prediction metrics produces the value to the Deep reinforcement Learning Layer. The development of the link prediction model involves mapping a network structure into a matrix of attributes for each pair of nodes derived from the link. The metrics chosen for the feature selection matrix are neighborhood-based, Katz-based, and random walk-based SNA metrics [13].

The node within the graph network that has neighbor node x is represented as $\varphi(x)$. The degree of relation in the node x is represented as k_x . The node between x and y, which has the number of walk length t is represented as n_{xy}^t . The discovery factor of longer length walk is denoted as β . Therefore, an array of feature selection metrics is developed as an information record for each pair of nodes. These records of data are used as a binary classification of positive-negative node pairs for training purposes in supervised machine learning. The pair of nodes where there is an edge is labeled as positive, and the pair where there is no edge is labeled as negative.

The link prediction algorithm will produce a dataset during the training stage where each multi-function information record represents either a couple of nodes with a link between them or a couple of nodes without a link between them. The various characteristics consist of the predictive SNA link metrics that apply to the node pair. The class label indicates the presence or lack of a link between the nodes as an outcome of creating the training model from the records. Each pair of nodes used in the testing stage is transformed into a record with various characteristics from which the model will predict binary classification of the presence of an edge.

In supervised Machine learning the Function approximator is a method, to identify the node pair which has the highest probability of missing link in the MCTS search. As a function approximator, a deep neural network will be used to take the initially corrupted network structure, S_0 as an input and output vector of edge probabilities. These edge probabilities are obtained for each node based on the SNA metrics, and the probability of missing links from the original network structure, S_0 is estimated. The value network was built with two neural network hidden layers to assess the probability distribution of nodes and edges from which the tree search function starts its search, use the preferential attachment index and the Hub index. The Value network will provide the weight to the edges of the dataset used in the criminal network.

Monte Carlo Tree search (MCTS) is a heuristic search algorithm for the process of decision making. It is used to analysis the most promising moves, expand the search tree based on random samples. It has four major steps 1. Selection, 2. expansion 3. Simulation and 4. Backpropagation. The proposed model utilizes an MCTS algorithm for policy network. Reinforcement Learning (RL) includes both the function of policy and value learning. Policy learning involves studying the field standards into the agent module that optimizes the benefits of each action, whereas value learning involves analyzing the agent model that



maximizes the projected future results of an action by generalizing from the aggregated outcomes of all previous state iterations and action combinations. The function of both value learning and policy learning will be trained using the NN (Neural Network) after that RL agent will form the policy and value network. The RL policy network will calculate the SNA matrix score based on neighborhood link prediction metrics, the Jaccard index, and the Adamic-Adar index to simulate link prediction in each network state.

Dataset D has total $2E$ edges; $E+$ positive edges and $E-$ negative edges. 75% of the dataset is used for the training data of the proposed model and represented as D_{train} . Remaining 25% of the Dataset to be used for testing phase and represented as D_{test} . Both training and testing data has equal number of positive and negative edges. Then the proposed DRL model will be trained and it will be evaluated using the test data and calculate the resulting Area Under Curve (AUC).

Random forests (RF) are an ensemble learning method for classification that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode based on a feature of the SNA metrics of the individual trees. State obtained from links with the greatest chance of presence based on the random selection of attributes will be optimized based on predictive accuracy processed against the test data.

Gradient Boosting Method (GBM) is a supervised machine learning model using a decision tree algorithm and improving learning efficiency by slowly decreasing the classification mistake resulting from the decision tree's previous iteration. When comparing the state of the network tree search with the greatest chance and the test data, learning rate parameter will be measured based on the validation feature.

Fig. 3. Proposed DRL model.

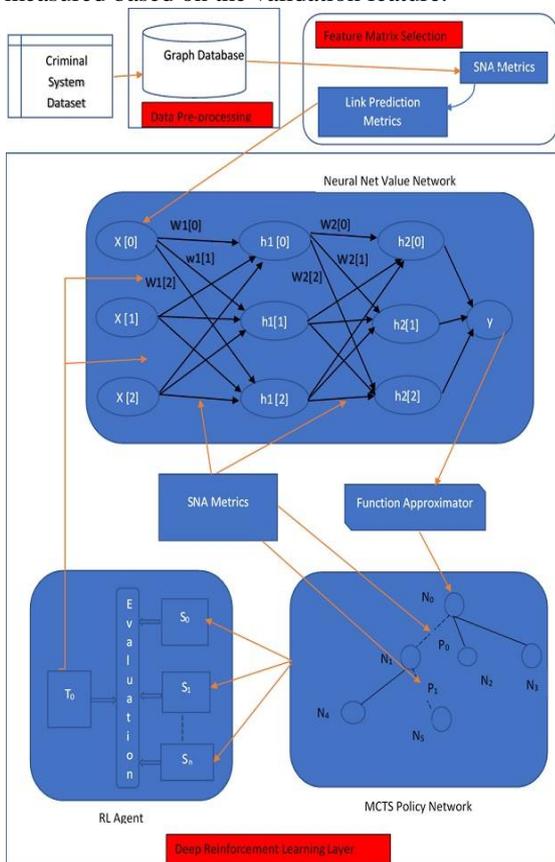
A. Algorithm for DRL:

1. Criminal network dataset stored in graph database is mapped to a feature matrix for each pair of nodes.
2. The SNA feature matrix of the common neighbor, Jaccard, Adamic-Adar Metrics, serve as the input layer $x [0], x [1], x [2]$ of the value network.
3. The SNA metrics of Hub index and preferential attachment index functions as weights $(w1[0], w1[1], w1[2]$ and $w2[0], w2[1], w2[2])$ for the hidden layer 1 ($h1[0], h1[1], h1[2]$ and hidden layer 2 $h2[0], h2[1], h2[2])$ respectively of the value network.
4. The Monte-Carlo tree search (MCTS) Module perform the network path traversal initiated on the random node pairs (N_0, N_1) with the highest probability of hidden links identified by the function approximate of the value network.
5. The rollout of the tree building will be based on the node pair with the highest probability (P_0, P_1) of hidden link.
6. The state, S_0 , to S_n denote networks reconstructed with the identified hidden links at the end of each simulated link prediction rollout. The Stage generated are evaluated against the initial training dataset (T_0) to measure the degree of success in the link prediction.
7. Performance metrics is feedback to value network to calibrate the weights parameters.

IV. EVALUATION METRICS AND RESULT

The criminal system dataset utilized for this examination depends on the UCINET cocaine pirating dataset, which contains a dichotomous contiguousness framework of medication clients in Hartford [5]. The cocaine pirating dataset from UCINET [5] contains four criminal system gatherings, i.e., JUANES, MAMBO, JAKE, and ACERO, which have 51, 62, 22, and 11 hubs, individually. The AUC is utilized as the assessment metric, as it is not interested in class irregularity and is a regularly utilized method to assess the classifier execution. The ACERO Network graph is shown in Fig. 4. Every criminal gathering dataset is partitioned into the preparation and test sets following a proportion of 75:25. The preparation set was utilized to construct both the GBM and DRL connect expectation model. The test set was connected in the assessment of the last model as far as the presentation of the parallel grouping capacity.

A portion of the hubs in the criminal system (Fig. 4) were then evacuated to reproduce an undermined criminal system dataset with missing connections (Fig. 5) for the consequent testing of the prepared connection forecast models. The tainted criminal system dataset was then changed over into a component framework and modified through both the GBM and DRL criminal system reproduction models (Fig. 6).



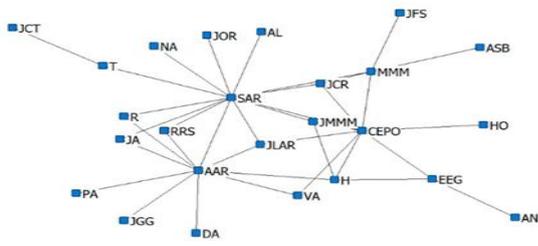
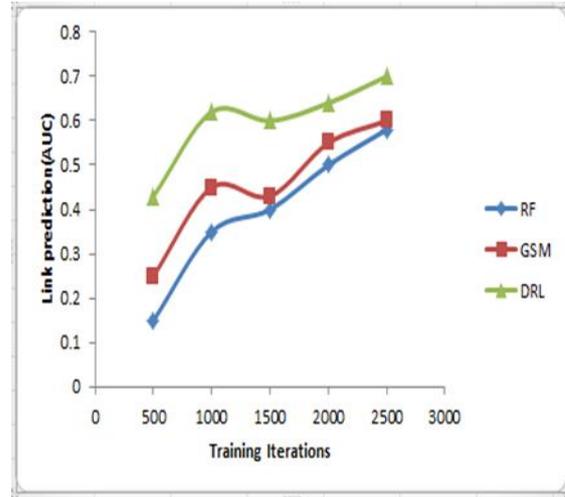


Fig. 4. Network Graph



(a)

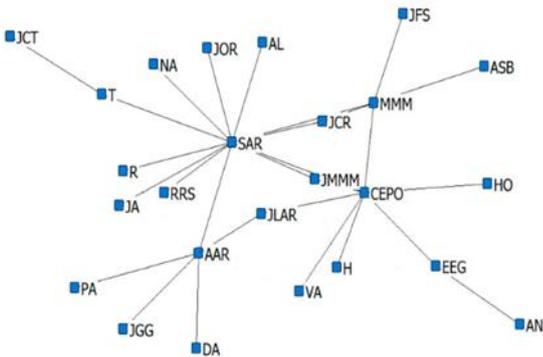
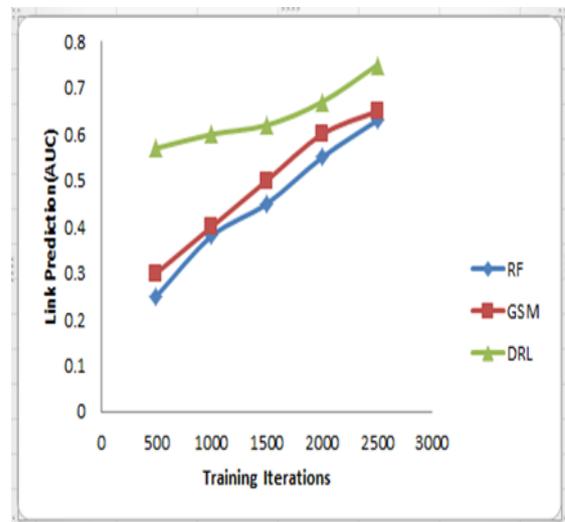


Fig. 5. corrupted Network Graph.



(b)

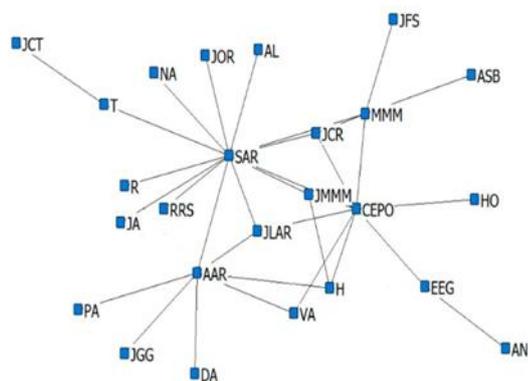
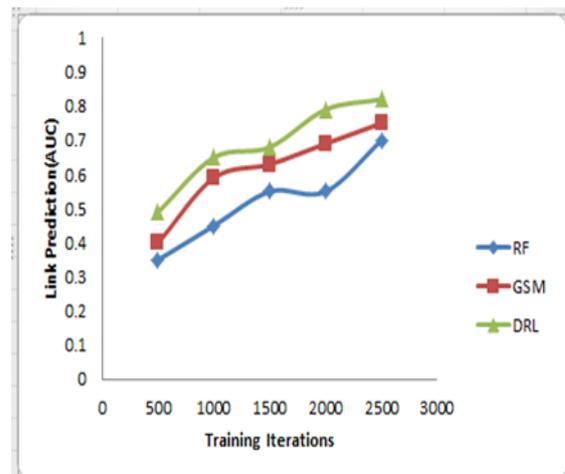


Fig. 6. constructed Network Graph with DRL.



(c)

Fig.7 demonstrates the AUC score from the criminal dataset JUANES, MAMBO and JAKE Network (a) For JUANES dataset which has 51 nodes , the proposed method of DRL has the higher value than GBM and RF , but at the 1500 iteration both supervised learning produces almost same result.(b) For MAMBO dataset which has 62 nodes the biggest size among the three dataset the proposed method of DRL has the higher value than GBM and RF , but at the 1000 and 2500 iteration both supervised learning produces almost the same result (c) For JAKE dataset which has 22 nodes the smallest size among the three dataset the proposed method of DRL has the higher value than GBM and RF , but at the 1500 iteration both DRL and GBM almost has the same result .

Fig. 7. AUC metrics of DRL, GBM and RF link prediction models for (a) JUANES network; (b) MAMBO network; (c) JAKE network.

Table II demonstrates that the DRL model outperforms the RF and GBM model in the three criminal system datasets.

Table- II. Accuracy of DRL, GBM, and RF link prediction model.

Dataset	RF	GBM	DRL	Iterations
Juanes	0.72	0.75	0.85	2500
Mambo	0.74	0.79	0.87	2500
Jake	0.67	0.71	0.78	2500

The classification efficiency outcomes were also measured using the confusion matrix rate for the number of accurate and inaccurate link predictions between network nodes of the criminal dataset.

Performance evaluation is used to measure the link prediction model in terms of correctly predicted decision of the models. Class variables can be assumed to be Positive (P) and Negative (N). Actual positive (P) which the classifier properly labels as positive are called true positive (TP). Actual positives wrongly labelled by the classifier as negative are known to be false negatives (FN). Actual negatives (N) which the classifier properly labels as negatives are called true negatives (TN). Actual negatives wrongly labelled by the classifier as positives are known to be false positives (FP). Confusion Matrix uses these terms.

The equations are

$$AUC = \frac{(\hat{n} + 0.5\hat{n})}{n}$$

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FN)}$$

$$Precision = \frac{TP}{(TP + FP)}$$

$$Recall = \frac{TP}{(TP + FN)}$$

$$TPR = \frac{TP}{(TP + FN)}$$

$$FPR = \frac{FP}{(TN + FP)}$$

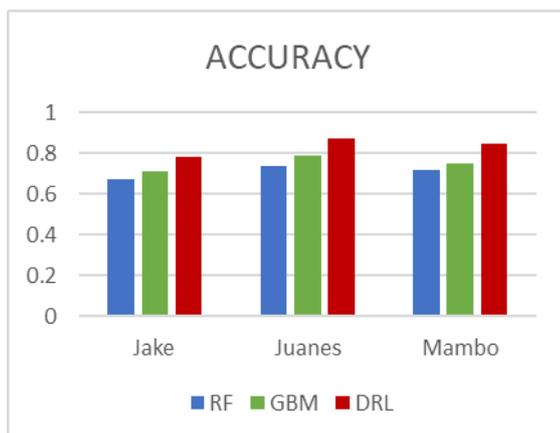


Fig. 8. Accuracy measure of Various model.

V. CONCLUSION AND FUTURE WORK

Before In this paper, an experiment was performed to show that a DRL-based link prediction model is capable of outperforming the RF, GBM model with a given dataset. This is shown by the DRL link prediction model's Accuracy results of 0.85, 0.87, and 0.78 in figure7 compared to the other model for JUANES, MAMBO, and JAKE, respectively. The accuracy comparison is made for the dataset among the DRL, GBM, Random Forest and in which the DRL suggests the excellent accuracy and also the precision, recall, is measured. Future work can be conducted with the complex dataset to better performance of the model and to develop the recommendation system for Crime detection using machine learning techniques.

REFERENCES

- Cheliotis, G. *Social Network Analysis*. 2014. Availableonline:<https://wiki.nus.edu.sg/download/attachments/57742900/Social+Networks+-+Social+Capital.pdf> (accessed on 19 October 2018).
- McAndrew, D. *The structural analysis of criminal networks. In the Social Psychology of Crime: Groups, Teams, and Networks; Offender Profiling Series*; ACM: Aldershot, UK, 1999; Volume III.
- Xu, J.; Chen, H. *Criminal Network Analysis and Visualization*. Commun. ACM 2005, 48, 100–107.
- Campana, M.G.; Delmastro, F. *Recommender Systems for Online and Mobile Social Networks: A survey*. Online Soc. Netw. Media 2017, 3, 75–97. [CrossRef]
- Borgatti, S.P.; Everett, M.G.; Freeman, L.C. *Ucinet for Windows: Software for Social Network Analysis*; Analytic Technologies: Harvard, MA, USA, 2002; Volume 1.
- Li, H.; Kumar, N.; Chen, R.; Georgiou, P. *A Deep Reinforcement Learning Framework for Identifying Funny Scenes in Movies*. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 3116–3120.
- Dzmitry, B.; Chorowski, J.; Serdyuk, D.; Brakel, P.; Bengio, Y. *End-to-end attention-based large vocabulary speech recognition*. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016.
- Zeiler, M.D.; Fergus, R. *Visualizing and Understanding Convolutional Networks*. In Proceedings of the European Conference on Computer Vision (ECCV), Sweden, UK, 6–12 September 2014; Volume 1.
- Yao, K.; Zweig, G.; Hwang, M.Y.; Shi, Y.; Yu, D. *Recurrent neural networks for language understanding*. Interspeech 2013, 2013, 2524–2528.
- Li, X.; Du, N.; Li, H.; Li, K.; Gao, J.; Zhang, A. *A deep learning approach to link prediction in dynamic networks*. In Proceedings of the 2014 SIAM International Conference on Data Mining, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 22–29 January 2014; pp. 289–297.
- Williams, R.J. *Simple statistical gradient-following algorithms for connectionist reinforcement learning*. Mach. Learn. 1992, 8, 229–256. [CrossRef]
- James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *Statistical learning. In An Introduction to Statistical Learning*; Springer: New York, NY, USA, 2013; pp. 15–57.
- Budur, E.; Lee, S.; Kong, V.S. *Structural Analysis of Criminal Network and Predicting Hidden Links using*. Mach. Learn. 2015.
- Hasan, M.A.; Zaki, M.J. *A Survey of Link Prediction in Social Networks*. Soc. Netw. Data Anal. 2011, 1, 243–275.
- Bellemare, M.G.; Dabney, W.; Munos, R. *A Distributional Perspective on Reinforcement Learning*. Mach. Learn. 2017.
- Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. *Human-level control through deep reinforcement learning*. Nature 2015, 518, 529–533.

17. Anthony, T.; Tian, Z.; Barber, D. *Thinking fast and slow with deep learning and tree search*. Adv. Neural Inf. Process. Syst. 2017, 5360–5370.
18. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Chen, Y. *Mastering the game of Go without human knowledge*. Nature 2017, 550, 354.
19. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, L.; Panneershelvam, V.; Lanctot, M.; et al. *Mastering the game of Go with deep neural networks and tree search*. Nature 2016, 529, 484–489

AUTHORS PROFILE



T.Manjunathkumar currently pursuing Ph.D in the department of Computer Science and Engineering at Kalasalingam Academy of Research and Education, Krishnankoil. He Completed M.Tech Dergree in Kalaslingam university and Received B.Tech degree in Anna university.



Dr.R.Murugeswari completed her Ph.D(CSE) from Kalasalingam University, Krishnankoil and M.E (CSE) from Arulmigu Kalasalingam College of Engineering, Krishnankoil under Anna University, Chennai. She is currently working as a Associate Professor in the Department of Computer Science and Engineering at Kalasalingam University, Krishnankoil. She has more than 18 years of Teaching Experience. Her research interests are wireless network, Evolutionary Algorithm, Big Data Analytics and Social Network Analysis. She has published more than 15 papers in various International journals and International Conferences.