2024 6th International Conference on Frontier Technologies of Information and Computer (ICFTIC) | 979-8-3315-4175-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICFTIC64248.2024.10913048

BGP Anomaly Events Detection Method Based on Deep Reinforcement Learning

Rui Wang
Infomation Engineering Institute
Zhengzhou, China
598251783@qq.com

Han Qiu*
Infomation Engineering Institute
Zhengzhou, China
*qiuhan410@aliyun.com

Shun Wang
Infomation Engineering Institute
Zhengzhou, China
1445362237@qq.com

Junyu Ren
Infomation Engineering Institute
Zhengzhou, China
3022443707@qq.com

***Abstract*—BGP is the most important protocol for inter-domain routing network connectivity, and plays a key role in the global network interconnection. However, the design flaws of BGP make it vulnerable to attacks or failures, leading to network fluctuations or even outages. Rapid detection of BGP anomalies and timely implementation of effective countermeasures are essential for maintaining internet security and stability. Current research on BGP anomaly detection has many problems such as poor generalization ability and high training cost, which makes it difficult to apply in real-world scenarios. To address theses problems, a semi-supervised BGP anomaly event detection method based on deep reinforcement learning, BAD-DRL, is proposed in this paper. This method trains an agent for anomaly detection scenarios, which autonomously learns the latent features of unlabeled data through an internally designed reward function and optimizes the agent's behavioral strategies through an externally designed reward function. Experiments show that the proposed method achieves an improvement in average accuracy of at least 1.35% on datasets from the same source and at least 7.57% on datasets from different sources.**

Keywords-BGP; anomaly event detection; deep reinforcement; key nodes

I. INTRODUCTION

Modern Internet consists of numerous Autonomous Systems (AS) that primarily exchange domain routing information through the Border Gateway Protocol (BGP), thereby establishing inter-domain routing networks and the global Internet. Consequently, the security and stability of BGP are paramount for ensuring the reliable and stable operation of the entire Internet. However, due to a lack of adequate security mechanisms in BGP, it is susceptible to various attacks or failures, leading to frequent BGP anomalies that significantly impact modern society's production and daily life.

In 2018, Amazon AWS was hijacked by BGP, resulting in a large number of users' cryptocurrency wallets being emptied [1]. In 2019, Safe Host, a Swiss data center hosting provider, made erroneous updates to its ASes routing protocols. This misconfiguration affected over 368 million IP addresses and disrupted more than 70,000 internet routes [2]. In 2021, Vodafone, a UK telecommunications company, experienced a route leak that involved the incorrect announcement of over 30,000 routes. This incident impacted more than 20,000 ASes globally [3]. Following the onset of the Russia-Ukraine conflict in 2022, there has been a notable increase in BGP anomalies between Russia and Ukraine. Three major ASes in Ukrainian have encountered routing failures. Several US operators interrupted their announcements to ASes in Russia, which have resulted in frequent disruptions to Russia's international internet routing capabilities.

BGP security has gradually become a crucial research direction in Internet security. By perceiving BGP anomalies and timely grasping the network security status, targeted defense response decisions can be made to ensure the stability and security of the Internet. With the research of machine learning algorithms, researchers have proposed various methods for BGP anomaly detection in different application scenarios. Yang et al. [4] proposed a new path-based BGP routing anomaly detection algorithm, which can capture the underlying structure of the routing table, and fuse the attribute information and topology information of the BGP routing table to detect BGP anomaly events. Huang et al. [5] proposed a method to construct a network graph by extracting information from BGP updates, used centrality information as features to model the graphical structure of the network, which can detect CenturyLink network outage events. Latif et al. [6] introduced a BGIN model based on Graph Isomorphism Networks (GIN) to detect anomalies in automatically generated graph representation vectors integrated with topological features. Hoarau et al. [7] developed a tool named BML, designed for the creation of BGP message datasets, which extracts both statistical and graph features from BGP datasets to enhance the detection of BGP anomalies. Additionally, the team proposed a BGNN model [8] based on Graph Convolutional Networks (GCN), which automatically extracts graph features for BGP anomaly detection by concatenating graph representation vectors from multiple consecutive time frames. Liu et al. [9] presented the ASSTFR BGP anomaly detection model, which is based on AS spatio-temporal feature representation, thereby improving anomaly detection accuracy by ranking AS node anomaly scores. Collectively, these algorithms are capable of automatically extracting features from BGP data and effectively detecting known types of anomalies; however, they necessitate labeled training datasets and exhibit limited efficacy in identifying unknown anomalies.

Deep reinforcement learning (DRL) integrates the feature representation capabilities of deep learning with the decision-

Supported by the Natural Science Foundation of He'nan Province (No. 242300421415),the Science and Technology Major Project of He'nan Province (No. 221100240100).

making processes inherent in reinforcement learning, thereby facilitating robust end-to-end control learning [10]. This approach effectively addresses critical challenges in anomaly detection, enabling the identification of novel anomaly types that extend beyond the confines of labeled datasets [11]. Furthermore, it enhances the robustness and adaptability of systems when confronted with unknown or newly emerging anomaly events. Traditional anomaly detection tasks primarily focus on the classification of normal or anomaly samples, which complicates the establishment of an intelligent agent interaction framework and the requisite reward feedback environment for deep reinforcement learning. Nevertheless, advancements in deep reinforcement learning algorithms have prompted researchers to progressively incorporate these techniques into the domain of anomaly detection. For instance, Hsu et al. [12] introduced ANIDS, an anomaly network intrusion detection system that leverages DRL to process network traffic at a scale of millions in real-time. Similarly, Alavizadeh et al. [13] presented a DRL model grounded in deep Q-learning, which synergizes Q-learning-based reinforcement learning with deep feedforward neural network methodologies for network intrusion detection. Moreover, Ma et al. [14] proposed AESMOTE, a comprehensive framework that integrates reinforcement learning algorithms with techniques to address class imbalance, a prevalent issue in existing solutions, thereby facilitating anomaly-based intrusion detection. Pang et al. [15] introduced DPLAN, a DQN model framework that operates on partially labeled anomalies, enhancing detection accuracy through training with a reduced amount of labeled anomaly data while effectively identifying unknown anomalies. The authors conducted experiments across 48 real datasets, all of which demonstrated superior performance compared to existing methodologies.

In summary, this paper integrates deep reinforcement learning algorithms into the investigation of BGP anomaly detection. The primary contributions of this research are outlined as follows:

- A semi-supervised BGP anomaly detection method, named BAD-DRL, is proposed, which is based on DRL. This method trained an agent specifically designed for anomaly detection scenarios, thereby enhancing the accuracy of anomaly detection and facilitating the identification of unknown anomalies. This is achieved through the design of both external and internal reward functions, which contribute to improved model generalization across new datasets and a reduction in the labeling costs associated with training sets. Experiments have shown that, in comparison to existing anomaly detection methods and the most recent BGP anomaly detection techniques, BAD-DRL achieves an improvement in average accuracy of at least 1.35% on datasets from the same source and at least 7.57% on datasets from different sources.

- A feature for enhanced perception of BGP anomalies is proposed, which is based on the importance ranking of key inter-domain routing nodes. This approach aims to detect BGP anomalies while minimizing potential noise interference, resulting in an accuracy improvement of 6.6% under equivalent training conditions.

II. The Proposed Method: BAD-DRL

This paper focuses on the simulation environment for unknown anomaly detection, mainly training an intelligent agent based on the DQN algorithm, which is the core of the BAD-DRL method. In this method, the environment is a collection of BGP packet feature data from the training set, and the observation function provides the current state value and next state value to the intelligent agent. The state is the current network's BGP packet feature, the action is the determination of whether the current network is anomaly, and the reward is the confidence score of the intelligent agent's current action, composed of an internal reward function and an external reward function. The core algorithm flow is shown in Fig.1.

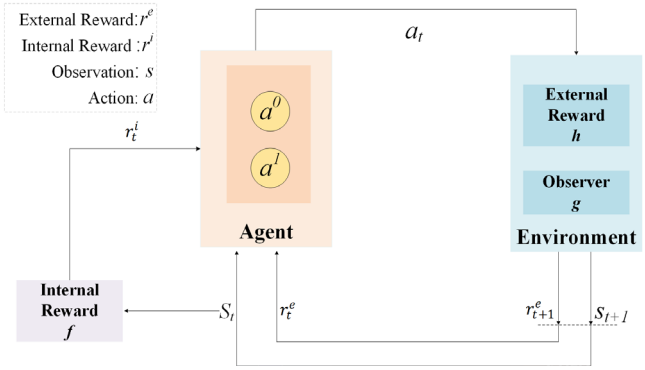

Figure 1. BAD-DRL Core Algorithm Flow Graph

Given the training datasets $D=\{D^a, D^u\}$, where $D^a \cap D^u = \varnothing$, D^a is the labeled datasets consisting entirely of labeled anomaly BGP message data, and D^u is the unlabeled datasets consisting of normal and anomaly BGP message data. The goal of the agent A trained by the algorithm is to learn an anomaly score function $Q(S)$ about the current environment state S, such that $Q(S_i) > Q(S_j)$, where S_i and S_j are BGP message data from datasets D, S_i stands for anomaly BGP message data. S_j represents the normal BGP message data.

During training, the algorithm selects the best action according to the given environment state: the BGP message data in the current state is judged as "normal" (a^0) or "anomaly" (a^1), where the simulated environment E is driven by an external reward function h and an observer g. Specifically, at each time t, agent A receives an observation S_t generated by an observer g, takes an action a_t, and then receives an external reward r_t^e generated by an external reward function h. When the agent successfully identifies the anomaly data in the D^a, the external reward function h gives positive feedback $r_t^e = 1$. When the agent identifies the data in D^u as normal data, the external reward function h value is 0; Other cases the external reward function gives negative feedback $r_t^e = -1$. The outer function aims to make the agent take full advantage of all the labeled anomaly data.

The observer g is composed of two functions, g_a and g_u. g_a samples randomly and uniformly from g_a, and g_a samples the observation S_{t+1} from D^u according to the current observation S_t. In order to effectively and efficiently explore D^u datasets, g_u is defined as:

$$g_u(S_{t+1} \mid S_t, a_t; \theta^e) = \begin{cases} \arg\min d(S_t, S; \theta^e), a_t = a^1 \\ \arg\max d(S_t, S; \theta^e), a_t = a^0 \end{cases}. \quad (1)$$

That is, when the agent considers the current observation S_t as an anomaly and takes action a^1, g_u returns the nearest neighbor of that observation; When the agent considers the current observation S_t as normal and takes action a^0, g_u returns the farthest neighbor of that observation. This approach allows the agent to quickly move away from normal values to explore potential anomalies, and can fully explore similar results to anomaly observations. During training, the simulator performs g_a with probability p and g_u with probability $1 - p$, to achieve a balance between exploring the unlabeled data and fully utilizing the labeled data.

In order to encourage the agent to actively explore the unlabeled BGP message data and quickly find the anomaly data in the unlabeled datasets D^u to detect possible unknown anomaly events, we also define an internal reward function f to provide an internal reward r^i to the agent based on the anomaly value of the observation S_t. In the real BGP data, anomaly data accounts for a small proportion of the total data. Under the condition of unlabeled training set, unsupervised anomaly detection algorithms are first considered, including K-Means clustering anomaly detection algorithm, isolation forest algorithm, local anomaly factor algorithm and so on. Among them, the isolation forest algorithm is an unsupervised Ensemble based fast outlier detection algorithm, which is one of the best outlier detection algorithms in the background of big data. Compared with other unsupervised algorithms, the isolation forest algorithm has smaller memory footprint, faster speed and higher accuracy. Therefore, the internal reward function f adopts the unsupervised isolation Forest algorithm [16] to measure the anomaly of the data in D^u and give the internal reward r^i. The overall reward received by the agent performing the action a_t at time t is defined as $r_t=r_t^e+r_t^i$.

After training, the algorithm returns an anomaly score function $Q(s, a; \theta^*)$, which is a near-optimal action-value function. When the model is used to perform anomaly detection on test data T in the test set t, for each test data t($t \in T$) in the test set, the model performs a forward pass through its network, and then obtains the best action a for the anomaly detection agent under the current observation: a^0 corresponds to "normal" data and a^1 corresponds to "anomaly" data. Finally, the set of test results for the test set is returned.

III. BGP Anomaly Enhanced Awareness Feature

The paper proposes a BGP anomaly enhancement perception feature model based on key nodes, which comprehensively considers how to improve the influence of key nodes in the feature representation model without causing information loss. The BGP anomaly enhanced perception feature model is the basis of the BGP anomaly detection algorithm proposed in this paper. Feature selection can transform data into features that can better represent business logic, so as to improve the performance of machine learning. This section introduces the typical BGP features commonly used in current BGP anomaly detection algorithms, and then conducts feature optimization and key node weight assignment. Finally, the BGP feature representation model fusing key node weights is formed.

A. BGP Feature Analysis and Optimization

In the existing BGP anomaly detection algorithms based on machine learning, the commonly used BGP features are divided into two categories. One is the statistical features, including the number of packets to reflect the stability of BGP and the AS_PATH related features to reflect the topology changes. The other category is graph features that reflect the underlying graph structure of BGP.As shown in Table I and Table II.

TABLE I. Typical Statistical Characteristics of BGP

Name	Meaning	Name	Meaning
nb_A	Number of route announcements	max_editdist	Maximum path edit length for all messages
nb_W	Number of route withdrawals	avg_editdist	Average path edit length for all messages
nb_implicit_W	Number of route implicitly withdrawn	avg_interarrival	Average time interval between all messages
nb_dup_A	Number of route repeatedly announcements	editdist_7	Number of AS paths with Edit Distance of 7
nb_dup_W	Number of route repeatedly withdrawns	editdist_8	Number of AS paths with Edit Distance of 8
nb_A_prefix	Number of announced prefixes	editdist_9	Number of AS paths with Edit Distance of 9
nb_W_prefix	Number of withdrawn prefixes	editdist_10	Number of AS paths with Edit Distance of 10
max_A_prefix	Maximum number of announcements per prefix	editdist_11	Number of AS paths with Edit Distance of 11
avg_A_prefix	Average number of announcements per prefix	editdist_12	Number of AS paths with Edit Distance of 12
max_A_AS	Maximum number of announcements per AS	editdist_13	Number of AS paths with Edit Distance of 13
avg_A_AS	Average number of announcements per AS	editdist_14	Number of AS paths with Edit Distance of 14
nb_orign_change	Number of origin attribute changes	editdist_15	Number of AS paths with Edit Distance of 15
nb_new_A	Number of new announcements not stored in RIB	editdist_16	Number of AS paths with Edit Distance of 16
nb_new_A_afterW	Number of re-announcements after withdrawns	editdist_17	Number of AS paths with Edit Distance of 17
max_path_len	Maximum path length for all messages	nb_tolonger	Number of update messages longer than the previous path
avg_path_len	Average path length of all messages	nb_toshorter	Number of update messages shorter than the previous path

TABLE II. TYPICAL GRAPH CHARACTERISTICS OF BGP

Name	Meaning	Name	Meaning
algebraic_connectivity	Algebraic connectedness of the graph	diameter	Diameter of the graph
assortativity	Assortativity of the graph	largest_eigenvalue	Largest eigenvalue of the graph
avg_cluster_coef	Average degree of connection of the graph	effective_graph_resistance	Effective graph resistance of the graph
natural_connectivity	The natural connectivity of the graph	symmetry_ratio	Symmetry_ratio of the graph
edge_connectivity	The edge connectivity of the graph	node_connectivity	Node connectivity of the graph
clustering	Clustering coefficient of the graph	nb_spanning_trees	Number of spanning trees
triangles	Number of triangles in the graph	weighted_spectrum_3	Weighted spectrum ranks 3
nb_of_nodes	Number of nodes in the graph	weighted_spectrum_4	Weighted spectrum ranks 4
nb_of_edges	Number of edges of the graph	percolation_limit	Percolation threshold of the graph

The goal of feature selection is to find the optimal feature subset. Feature selection can eliminate irrelevant or redundant features, so as to reduce the number of features, reduce the running time, and improve the accuracy of the model. However, not all features can contribute to BGP anomaly detection. Too many redundant features will increase noise interference, which will affect the accuracy and sensitivity of the BGP anomaly detection model.

According to the form of feature selection, there are three main categories of feature selection methods [17] : filter, wrapper and embedded. Among them, the process of filtering method feature selection is completely independent of any machine learning algorithm, so this paper uses filtering method to select BGP features, specifically using the Max-Relevance and Min-Redundancy (mRMR) algorithm. The algorithm will consider the redundancy between features when selecting features, and try to eliminate these redundant features, which helps to reduce the risk of model overfitting and improve the generalization ability of the model. At the same time, the computational complexity of the algorithm is relatively low, and it is more suitable for dealing with large-scale high-dimensional data sets.

B. BGP Anomaly Enhancement Perception Feature Construction based on Key Nodes

When the network topology changes, BGP can quickly adapt to the change, recalcate the path and update the routing table. BGP also adopts routing feedback, routing aggregation, routing policy and other mechanisms to improve the stability of the network. In addition, BGP also supports multi-path routing, which improves the load balancing ability of the network. Therefore, in the inter-domain routing network, the short-term failure of ordinary nodes may not affect the network connectivity, and the anomaly of critical nodes is more likely to lead to large-scale network anomalies, so the change of characteristics of critical nodes can better reflect the occurrence of BGP anomaly events. The BGP feature representation model with the weight of key nodes aims to make full use of the key as nodes in the inter-domain routing network to describe the overall network characteristics, and improve the weight of key nodes in the network characteristics related to statistical data. In order to avoid the loss of information of the nodes at the end of the ranking, the calculation weight assigned to the key nodes should not be too large.

In this paper, the key nodes of the inter-domain routing network are found and ranked according to the key nodes discovery algorithm KNI-DRL[18]. Therefore, when calculating the BGP statistical characteristics, the weight of the last ranked AS node is 1, the weight of the first ranked AS node is $n(n\in(1,3))$, and the weight of the ranking node x is $1+x(x-1)/n$. The mRMR value of each feature under different weights was calculated, and the weight assignment of the key node with the largest mRMR value was used. After determining the weights, the BGP features are ranked according to the mRMR value, and the top 30 features are selected to construct the BGP feature representation model.

IV. EXPERIMENTAL VERIFICATION AND ANALYSIS

A. Experimental Datasets

- Anomaly events selection

In this section, seven typical BGP anoamly events are selected to construct the data set. The causes of anomaly events include three different types, such as prefix hijacking, route leakage, and link failure. The basic information of the event is detailed in Table III.

TABLE III. BASIC INFORMATION ABOUT TYPICAL BGP ANOMALY EVENTS

Name	Type	Begin Time(UTC)	Continue (min)	Anomaly ASN
TTNet	Route leakage	2004/12/24 9:20	630	9121
TM	Route leakage	2015/6/12 8:43	182	4788
IndoSat	Prefix hijacking	2014/4/2 18:25	150	4761
AWS	Prefix hijacking	2016/4/22 17:10	115	200759
Cloudflare	Prefix hijacking	2024/6/27 18:51	456	267613
Moscow	Link failure	2005/5/25 14:00	360	12793
Donetsk	Link failure	2022/9/11 17:30	330	207633

- Datasets construction

BGP feature extraction is performed according to the BML model proposed by Hoarau et al. [7], and the extracted features are described in Section III. The BML approach extracts data from the authoritative BGP routing information collection projects RouteViews and RIPE RIS. BML uses the BGPStream framework at the bottom, which can extract message features, network topology, etc. It also allows users to customize the cleaned BGP data to extract custom features, which is in line with the application scenario of this paper. The BML algorithm is used to sample the data of the selected anomaly BGP events. The sampling time is the whole period of the anomaly event and the 7 days before and after the event, and the sampling frequency is once per minute. The algorithm of fusing the weights of key nodes in Section III was used to extract and construct the features of the sampled data, calculate their mRMR values, and selecte the features. The normal data and a small amount of labeled anomaly data are randomly selected to form the training data set, where the ratio of anomaly data to normal data is 1:100, and the remaining data is used as the test datasets.

B. *Experiment Settings*

- Operating Environment

The server running the experiment is Ubuntu 20.04, equipped with an Intel(R) Xeon(R) Gold 5218 CPU @ 2.30 GHz, 256 GB memory, a total of 13TB hard disk, and two graphics cards NVIDIA RTX A6000. Python 3.8 was used as the programming language to build the deep reinforcement learning model based on PyTorch.

- Methods of Comparing

In this paper, the most widely used classical machine learning algorithms and deep learning methods are selected for comparison. The selected algorithms are as follows: classical machine learning methods: Support Vector Machine (SVM), K-Nearest Neighbor (KNN) and Random Forest(RF). Classical deep learning algorithms: Multi-Layer Perceptron (MLP), Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM). Methods based on graph representation learning: Graph Isomorphic Network(GIN) based method BGIN, Graph Convolutional Network(GCN) based method BGNN and Graph2Vec based method GE-LSTM. The classic algorithm is implemented using a PyOD (Python Outlier Detection) library.

- Evaluation Index

In order to verify the effectiveness of the proposed method in BGP anomaly event detection, the confusion matrix is used to establish the evaluation index. The confusion matrix consists of four classes: True Positive(TP), True Negative(TN), False Positive(FP), and False Negative(FN). Based on the above category combination, four evaluation metrics are defined: Accuracy, Precision, Recall and F1-score. The formula for calculating the index is as follows:

$$\text{Accuracy}=(TP+TN)/(TP+TN+FP+FN) \quad (2)$$

$$\text{Precision}=TP/(TP+FP) \quad (3)$$

$$\text{Recall}=TP/(TP+FN) \quad (4)$$

$$\text{F1-score}=2TP/(2TP+FP+FN) \quad (5)$$

C. *Experimental Results and Analysis*

In this section, the seven datasets constructed in the previous section were trained and tested. At the same time, the models trained on each dataset were also applied to other datasets for testing. The experimental results are shown in Fig.2 to Fig.8.

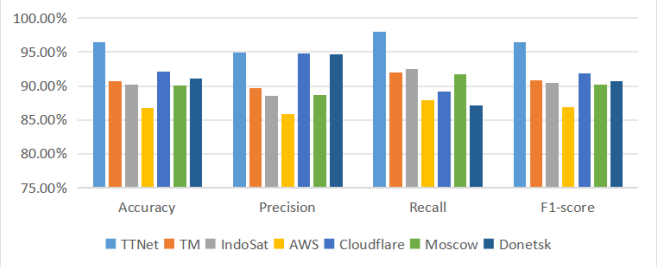

Figure 2. TTNet data as test datasets

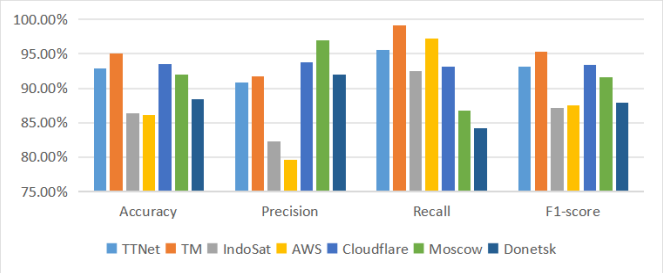

Figure 3. TM data as test datasets

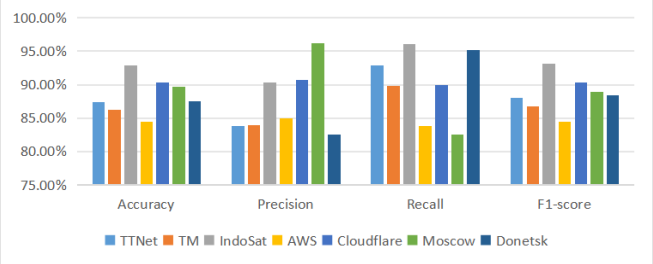

Figure 4. IndoSat data as test datasets

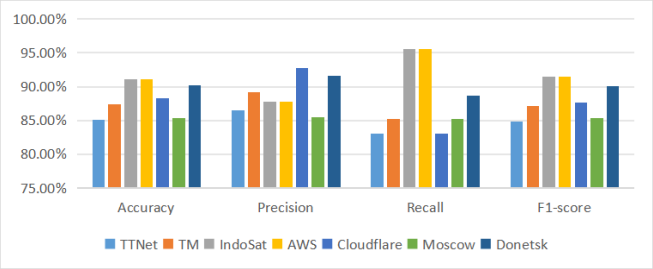

Figure 5. AWS data as test datasets

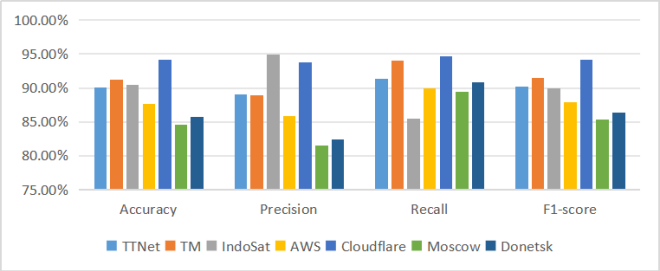

Figure 6. Cloudflare data as test datasets

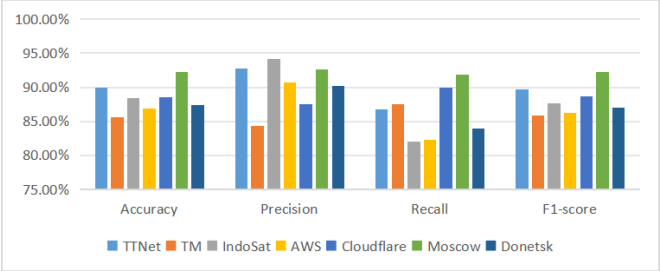

Figure 7. Moscow data as test datasets

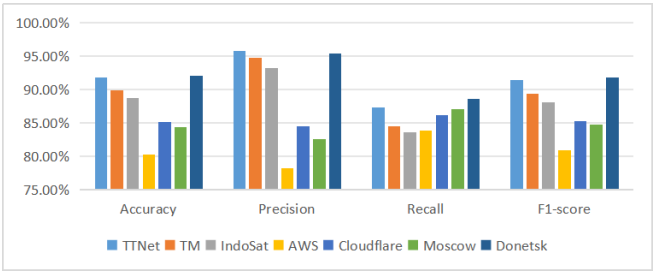

Figure 8. Donetsk data as test datasets

It can be seen that the algorithm model achieves the best test effect when the training set and test set are on datasets from the same source, and the average evaluation index is: The accuracy rate is 93.45%, the precision is 92.36%, the recall rate is 94.85%, and the F1 score is 93.52%, which is 1.35%, 1.87%, 0.77%, 1.27% higher than the current best-performing GE-LSTM algorithm. When tested on datasets from different source, the performance metrics of the model decrease a bit, but still remain high. Overall, on average, the model achieves: The accuracy rate is 89.07%, the precision is 89.14%, the recall rate is 89.38%, and the F1 score is 89.11%, which is 7.57%, 4.89%, 11.90%, 8.39% higher than the current best-performing GE-LSTM algorithm.

The comparison experiment was carried out for the selected comparison algorithms, and the average value of related evaluation indicators was calculated. The comparison situation is shown in Fig.9 and Fig.10. It can be seen from Fig.9 that the performance index of the BAD-DRL method proposed in this paper is significantly better than the classical machine learning algorithm models, and is similar to most of the algorithm models based on deep learning and graph representation learning. Fig.10 clearly shows that the BAD-DRL method has a relatively superior generalization ability and outperforms all the compared methods. Based on the above experiments, the BAD-DRL method has higher recognition accuracy and better transferability, which is more in line with realistic application scenarios. The application of semi-supervised learning greatly reduces the labeling requirements of experimental data sets and reduces labor costs.

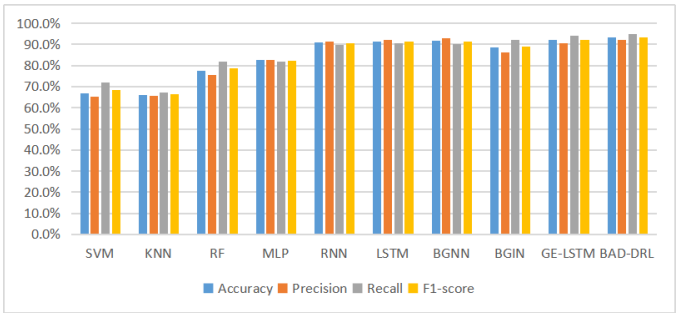

Figure 9. Comparison of algorithm performance

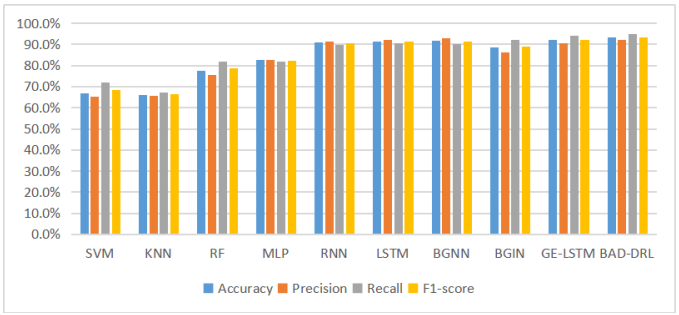

Figure 10. Comparison of algorithm migration ability

V. Conclusions

In this paper, we propose a deep reinforcement learning-based method for BGP anomaly event detection. This method constructs an interactive environment for BGP anomaly detection by designing an observation algorithm, and designs internal and external reward functions to drive the agent to make full use of labeled anomaly data and actively explore unlabeled BGP data to find potential anomalies. In order to improve the accuracy of detection, a BGP anomaly enhanced perception feature model is constructed according to the importance of key nodes in the inter-domain routing network to reduce the possible noise interference. The experimental results show that the average accuracy of BAD-DRL method is increased by at least 1.35%on its homologous data sets, and the average accuracy is increased by at least 7.57%on non-homologous data sets, showing good generalization ability. In addition, semi-supervised learning also greatly reduces the labeling requirements of the training set.

References

[1] A. Siddiqui, “What happened? The Amazon route 53 BGP hijack to take over ethereum cryptocurrency wallets,” https://www.internetsociety.org/blog/2018/04/amazons-route-53-bgp-hijack/, (2018-04-27).

[2] C. Cimpanu, “For two hours, a large chunk of European mobile traffic was rerouted through China,” https://www.zdnet.com/article/for-two-hours-a-large-chunk-of-european-mobile-traffic-was-rerouted-through-china/, (2019-06-07).

[3] A. Siddiqui, “A major BGP hijack by AS55410-Vodafone Idea Ltd,” https://manrs.org/2021/04/a-major-bgp-hijack-by-as55410-vodafone-idea-ltd/, (2021-04-17).

[4] C. Yang and W. Jia, “BGP anomaly detection - a path-based apporach,”in 2023 3rd Asia-Pacific Conference on Communications Technology and Computer Science. IEEE, 2023, pp. 408–414.

[5] J. Huang, M. Odiathevar, A. C. Valera, J. Sahni, M. Frean, and W. K. G.Seah, “Realtime BGP anomaly detection using graph centrality features,”in International Conference on Advanced Information Networking and Applications. Springer Nature Switzerland, 2024, pp. 222–233.

[6] H. Latif, J. Pailliss´e, J. Yang, A. Cabellos-Aparicio, and P. Barlet-Ros,“Unveiling the potential of graph neural networks for BGP anomalydetection,” in 1st International Workshop on Graph Neural Networking. Association for Computing Machinery, 2022, p. 7–12.

[7] K. Hoarau, P. U. Tournoux, and T. Razafindralambo, “BML: An efficient and versatile tool for BGP dataset collection,” in 2021 IEEE International Conference on Communications Workshops. IEEE, 2021, pp.1–6.

[8] K. Hoarau, P. U. Tournoux, and T. Razafindralambo, “BGNN: Detection of BGP anomalies using graph neural networks,” in 2022 IEEE Symposium on Computers and Communications. IEEE, 2022, pp. 1–6.

[9] Z. Liu, H. Qiu, R. Wang, J. Zhu, and Q. Wang, “Detecting BGP anomalies based on spatio-temporal feature Representation Model for Autonomous Systems,” in 2023 IEEE 22nd International Conference on Trust, Security and Privacy in Computing and Communications. IEEE, 2023, pp. 404–411.

[10] X. Wang, S. Wang, X. Liang, D. Zhao, J. Huang, X. Xu, B. Dai, and Q. Miao, “Deep reinforcement learning: A survey,” IEEE Transactions on Neural Networks and Learning Systems, vol. 35, no. 4, pp. 5064–5078, 2022.

[11] K. Arshad, R. F. Ali, A. Muneer, I. A. Aziz, S. Naseer, N. S. Khan, and S. M. Taib, “Deep reinforcement learning for anomaly detection: A systematic review,” IEEE Access, vol. 10, pp. 124 017–124 035, 2022.

[12] Y.-F. Hsu and M. Matsuoka, “A deep reinforcement learning approach for anomaly network intrusion detection system,” in 2020 IEEE 9th International Conference on Cloud Networking. IEEE, 2020, pp. 1–6.

[13] H. Alavizadeh, H. Alavizadeh, and J. Jang-Jaccard, “Deep Q-Learning based reinforcement learning approach for network intrusion detection,” Computers, vol. 11, no. 3, p. 41, 2022.

[14] X. Ma and W. Shi, “AESMOTE: Adversarial reinforcement learning with SMOTE for anomaly detection,” IEEE Transactions on Network Science and Engineering, vol. 8, no. 2, pp. 943–956, 2021.

[15] G. Pang, A. van den Hengel, C. Shen, and L. Cao, “Toward deep supervised anomaly detection: Reinforcement learning from partially labeled anomaly data,” in 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. Association for Computing Machinery, 2021, p. 1298–1308.

[16] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation forest,” in 2008 Eighth IEEE International Conference on Data Mining. IEEE, 2008, pp. 413–422.

[17] A. Moslemi, “A tutorial-based survey on feature selection: Recent advancements on feature selection,” Engineering Applications of Artificial Intelligence, vol. 126, no. PD, p. 107136, 2023.

[18] R. Wang, H. Qiu and Z. Liu, “KNI-DRL: Key Nodes Identification Method Based on Deep Reinforcement Learning in Inter-domain Routing Networks,” in 2024 9th International Conference on Computer and Communication Systems, Xi'an, China, 2024, pp. 698-705.