Adaptive Multi-Arm Bandit Framework for Detecting Network Intrusions in Real Time

Yaolin Chen

School of Cyber Science and Engineering, Wuhan University, Wuhan, Hubei Province, 430072, China

Abstract. Network intrusion detection remains a critical challenge in cybersecurity, particularly in dynamic environments with evolving attack patterns. This paper proposes an innovative Multi-Armed Bandit (MAB) framework that addresses the limitations of traditional static machine learning models by dynamically selecting and adapting among three detection algorithms: Random Forest (RF), Recurrent Neural Network, and Extreme Gradient Boosting (XGBoost) in real-time. This approach sets each machine learning model as an independent arm and uses real-time F1-scores as rewards for optimal arm selection. Evaluations on the CIC-IDS2017 dataset demonstrate the framework's good performance, achieving an exceptional F1-score of 0.969 and accuracy of 0.943, significantly outperforming both random selection (F1: 0.939) and static Random Forest (F1: 0.950) approaches. Among various MAB algorithms tested, Thompson Sampling (TS) exhibited the strongest performance with an F1-score of 0.963 and accuracy of 0.941, demonstrating remarkable effectiveness in balancing exploration-exploitation trade-offs. The framework's consistent performance across multiple datasets (including CIC-IDS2018, UNSW-NB15, and NSL-KDD) confirms its robustness and generalization capability, suggesting substantial potential for real-world deployment in complex network environments. These results highlight the MAB framework's adaptive advantages and position it as a promising solution for next-generation intrusion detection systems.

1 Introduction

In the digital revolution era, the volume and sophistication of network attacks are growing really fast. Modern network environments are facing increasingly complex and ever-evolving cyber threats, making intrusion detection and response one of the obvious challenges in cybersecurity. Network intrusions can bring about severe consequences, like the leakage of sensitive information, the disruption of services, and financial losses for individuals, organizations, and even whole economies. Intrusion Detection Systems (IDS) have become essential in people's lives, as they monitor network traffic to spot malicious activities and protect the integrity of communication and data. However, traditional IDSs depend on predefined rules or fixed machine learning models, and it has been found that they don't work well against fast-changing environments [1]. The application of supervised and unsupervised

2023302181047@whu.edu.cn

machine learning techniques in IDS has enhanced the performance of IDS solutions to a significant extent, yet they have been found to be rather ineffective against the fast-changing adversarial environments [2]. To deal with this limitation, an adaptive MAB framework that can dynamically select optimal detection strategies in real time is put forward.

The MAB framework, which gets its inspiration from the exploration-exploitation trade-off issue in decision-making, has come up as a hopeful way to handle such difficulties. In the setting of network intrusion detection, the MAB framework is able to actively pick out the most useful detection strategies from a group of available choices. Through constantly learning from the feedback that comes from the network environment, the IDS based on MAB can adjust to the changing network situations and intrusion patterns right away in real time.

This article takes the data-driven intrusion detection models built with machine learning classification techniques as its foundation [3]. It picks out three models in total: RF, k-Nearest Neighbors (KNN), and XGBoost. The approach put forward here uses a MAB framework to actively select the best ML model according to the real-time detection performance. In this way, it keeps the accuracy of the IDS intact. This article uses Accuracy and F1-score as metrics to assess its performance. By constantly adjusting the model choice, the IDS is able to do online learning and strengthen its capacity to respond to threats that are evolving in real time.

The remaining part of this article is arranged like this. Section 2 takes a look at the relevant literature. Section 3 gives an account of the methodology. The results are presented in Section 4, and then the discussion and conclusions follow in Section 5.

2 Related Work

This article shows that traditional IDS are relying more and more on machine learning (ML) to spot cyber threats. That's because the signature-based methods find it hard to keep up with the ever-evolving attack techniques [4]. Machine learning algorithms like RF, KNN, and XGBoost have performed well in this task [5-7]. These algorithms do this by using labeled datasets to tell the difference between normal and abnormal behavior. For example, RF is really good at dealing with high-dimensional data through ensemble voting. KNN, on the other hand, can capture local patterns in the feature space. And XGBoost makes gradient boosting better for imbalanced datasets. However, static ML models have big limitations in dynamic network environments where attack patterns are always changing. A single fixed model might become less accurate over time. To handle this challenge, adaptive frameworks that can do dynamic model selection are really important.

The MAB algorithm, which is a reinforcement learning algorithm, has gained traction for decision-making in uncertain environments by balancing exploration and exploitation. In cybersecurity, MAB has been used for problems such as honeypot allocation. Here, the attackers' preferences aren't known. It also simulates the problem of adversarial attacks and does hyperparameter optimization for ML models [2, 8, 9]. But its potential for adaptive model selection in IDS hasn't been explored enough yet.

This article fills the research gap by coming up with an MAB-driven approach to switch among RF, KNN, and XGBoost in line with real-time performance metrics. The MAB agent keeps assessing each model's effectiveness through rewards and tweaks its selection approach to boost the cumulative performance.

3 Methodology

3.1 Overview of the Methodology

This section focuses on data pre-processing and the detailed implementation of three deep learning algorithms. This study proposes a MAB framework for the purpose of dynamically picking optimal intrusion detection strategies from pre-defined machine learning models. As shown in Fig. 1, the methodology part consists of three key components: the Dataset and Preprocessing, the MAB-Based Model Selection Framework, and performance evaluation.

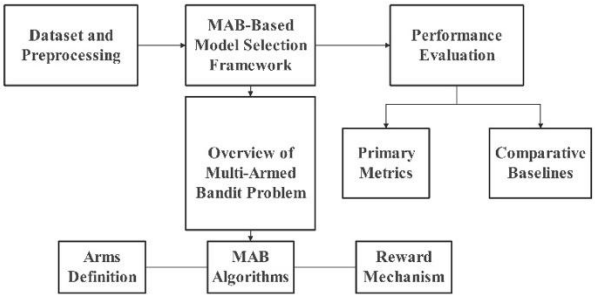

Fig. 1 The flow chart of this study (Photo credit: Original)

3.2 Dataset and Preprocessing

This article uses the CIC-IDS2017 dataset for this framework. Data preprocessing is a crucial step before putting data into machine learning models. The whole dataset is made up of eight CSV files, and these were joined together to form a single unified dataset. The preprocessing stage had the following steps: Firstly, special characters like commas in column names were spotted and taken out. To deal with Infinite Values, it was found that the columns 'flowbytes/s' and 'flowpackets/s' had 1,509 and 2,867 infinite values, respectively, and these were sorted out. After that, the Null values were checked in all columns. The columns 'flowbytes/s' and 'flowpackets/s' each had 2,867 missing entries. These nulls were replaced with zeros. Then, create the description of Statistics, which includes count, mean, standard deviation, min/max values, and quartiles (25%, 50%, 75%).

This article first applies principal component analysis (PCA) in the exploratory analysis. This is aimed at getting rid of multicollinearity. It reduces the dimensions to 46 features, and the label is included. After that, standardization and categorical encoding are done. The processed data is split into training-test sets in a 70-30 ratio. Both of these sets then go through normalization, mathematical transformation, and the generation of summaries with precision formatting (to 3 decimal places). This way, the complete preparation for the model is achieved.

3.3 MAB-Based Model Selection Framework

3.3.1 Overview of MAB Problem

The MAB problem is a class of sequential decision-making tasks. It involves dynamically allocating limited resources among competing alternatives [10]. Essentially, the MAB framework focuses on the basic trade-off between exploitation (choosing actions that yield immediate rewards) and exploration (gathering information to improve long-term outcomes).

The bandit consists of a set of arms representing action or decision possibilities. Each arm is selected to give the rewards sampled from a distribution. The basic algorithm for a bandit having K arms with a stationary distribution.

3.3.2 Arms Definition

In the MAB framework put forward, each 'arm' stands for a separate ML model that's been trained to detect intrusions. These three arms are picked because they have complementary strengths when it comes to dealing with network attack data. Now, let's go ahead and explain each arm's design reasoning and the process of hyperparameter tuning within the MAB in a more detailed way.

First up is the RF. This article constructs multiple decision trees during training and combines the predictions through majority voting in the case of classification or averaging for regression. The inherent randomness it has in feature and sample selection lessens the problem of overfitting, which makes it a good fit for intrusion detection datasets such as CIC-IDS2017. When it comes to Hyperparameter tuning, the Number of estimators was optimized by way of grid search (with a range of 50 to 200 trees), and it ended up with a final value of 150. The Max depth was capped at 10 to avoid excessive complexity.

Next up is KNN. KNN is a lazy learning algorithm. Its classification principle is simple: it predicts the label of a sample based on the majority vote of its k nearest neighbors in the feature space. As a non-parametric model, KNN can flexibly adapt to local patterns in the data. After tuning, the k-value was set to 5 and Euclidean distance as the metric.

XGBoost improves predictions step by step through gradient boosting, combining multiple weak learners (decision trees). The model also includes L1/L2 regularization to control complexity. In the experiments, the learning rate was adjusted to 0.1 and the maximum tree depth to 6.

3.3.3 MAB Algorithms

This article implemented and compared three basic MAB algorithms for the selection of dynamic models.

First up is TS with Beta Priors. This Bayesian way of doing things models the reward distribution of each arm as a Beta(α, β). Here, α stands for the counts of successes, that is, the correct detections, and β indicates the failures. Posterior sampling allows for a natural kind of exploration. What happens is that the arms, which are estimated to have a higher probability of success, are more likely to get selected. And at the same time, it keeps a nonzero exploration probability for all the arms. This is especially useful in situations where there's only a limited amount of initial data. That's because it can automatically balance the trade-offs between exploration and exploitation, and it doesn't need any explicit tuning of parameters.

Next up is the ϵ-Greedy Strategy. This algorithm is a simple and effective heuristic. It picks the arm that's currently performing best (using empirical F1-scores) with a probability of 1 - ϵ. At the same time, it randomly explores other arms with a probability of ϵ. Here, ϵ follows an exponential decay schedule. The initial value of ϵ is 0.3, and the decay rate is 0.95 per epoch. This way, it gradually moves from exploration to exploitation. It offers a strong baseline that ensures continuous exploration. However, it might waste resources on arms that are clearly not optimal in the later stages.

The last one is the Upper Confidence Bound (UCB). This algorithm chooses the arms by making the upper confidence bound as large as possible.

$$UCB(i) = \widehat{\mu_i} + \sqrt{\frac{2 \ln n}{n_i}} \quad (1)$$

where $\widehat{\mu_i}$ is the empirical reward mean of an arm. n is the total trials, and n_i is the arm's trials. - The second term creates an optimism-under-uncertainty principle that systematically reduces uncertainty about less-tried arms. Eventually, it converges to the optimal arm.

3.3.4 Reward Mechanism

The reward for each selected ML model is defined as its F1-score on a held-out validation set. This validation set is periodically updated to reflect the model's current performance.

The F1-score is chosen as the reward because it effectively balances precision (the fraction of correctly predicted attacks out of all predicted attacks) and recall (the fraction of correctly predicted attacks out of all actual attacks). This makes it particularly suitable for imbalanced intrusion detection tasks, where both false alarms and missed attacks matter.

The F1-score gets calculated as the harmonic mean of precision and recall.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (2)$$

where:

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

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

TP, FP, and FN stand for True Positives, False Positives, and False Negatives, respectively. Precision reflects the proportion of true positive samples among the samples predicted as positive by the model. It emphasizes reducing false positives and is suitable for scenarios where the cost of misjudgment is high. Recall measures the model's ability to cover true positive examples and focuses on reducing false negatives. The F1-score balances these two through the harmonic mean. The closer its value is to 1, the better the model's performance. It is especially applicable to scenarios with class imbalance or when both false positives and false negatives need to be taken into account.

3.4 Performance Evaluation

3.4.1 Primary Metrics

This article conducted a performance evaluation of the proposed MAB framework through a comprehensive experimental study that aimed to assess both the detection effectiveness and the operational efficiency. The evaluation used multiple metrics. The F1-score was taken as the primary performance indicator, as it can balance between precision and recall. Accuracy was used as a supplementary measure, too. The specific meaning and calculation method of the F1 score and accuracy have been mentioned above, and the calculation method here is the same.

3.4.2 Comparative Baselines

To evaluate the effectiveness of the MAB-based intrusion detection system, a comparative experiment was conducted against two baseline approaches. First is a traditional static model approach represented by RF. The second is a random selection strategy that switches between the three models with equal probability (1/3). The RF model served as a performance benchmark for conventional static detection systems, while the random selection strategy provided a reference for assessing the advantage of intelligent model switching. All experiments were repeated 10 times with different random seeds to ensure statistical reliability. The final results were calculated as the average across these trials. This experimental design enables comprehensive validation of this system's optimization capabilities while controlling for potential random variations.

4 Results and Analysis

4.1 Performance Evaluation on CIC-IDS2017

A comprehensive evaluation of the MAB framework was conducted using the complete CIC-IDS2017 dataset, which contains five consecutive days of network traffic data encompassing normal activities as well as seven types of attacks, including Distributed Denial of Service (DdoS), port scanning, and brute-force attacks. In the experimental setup, the model automatically switches detection strategies through the MAB framework every 50-100 samples upon detecting changes in attack types.

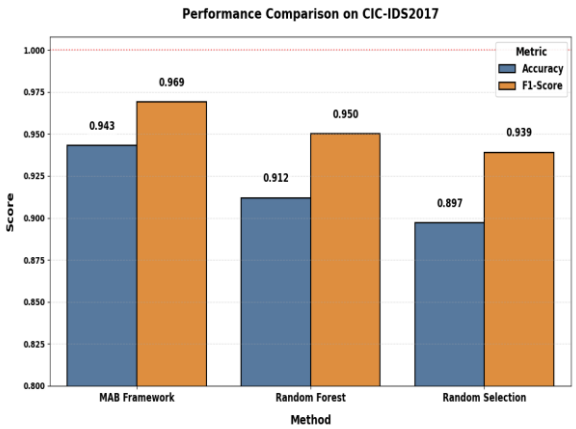

Fig. 2 Performance Evaluation (Photo credit: Original)

As shown in Fig. 2, the MAB framework achieved an outstanding F1-score of 0.969. In comparison, the Random Forest method and the random selection strategy only attained F1-scores of 0.950 and 0.939, respectively, along with significantly lower accuracy than the MAB framework. These results robustly demonstrate that the MAB framework exhibits superior adaptability and real-time decision-making advantages in dynamic network attack environments.

4.2 Comparison of MAB Algorithm

To further assess the impact of this adaptive framework, a detailed comparison of three MAB algorithms was carried out to evaluate their performance on dynamic model selection in intrusion detection.

As shown in Fig. 3, TS showed excellent performance, attaining the top F1-score of 0.963 and the highest accuracy level at 0.941. It effectively balanced exploration and exploitation by modeling reward distributions like Beta priors.

The ε-Greedy approach managed to keep a reasonably good F1-score of 0.949. It's easier to put into practice. However, it showed a slower rate of reaching convergence. This is because it has a fixed exploration rate like this.

UCB gets a competitive F1-score, yet the accuracy was on the lower side. It came up with a way of systematic exploration. But its deterministic side led to too much exploration.

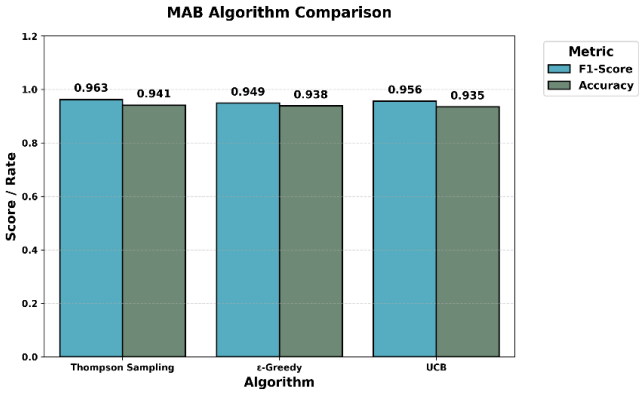

Fig. 3 MAB Comparison((Photo credit: Original)

4.3 Evaluation of Generalization across Datasets

To validate the robustness and generalization capability of the MAB framework, a comprehensive testing on three benchmark datasets: CIC-IDS2018, UNSW-NB15, and NSL-KDD was conducted. All datasets underwent similar data preprocessing procedures to CIC-IDS2017.

Table 1 MAB Framework Performance Across Multiple Datasets

	F1-Score	Accuracy
CIC-IDS2018	0.957	0.932
UNSW-NB15	0.943	0.928
NSL-KDD	0.953	0.934

As Table 1 shows, on the CIC-IDS2018 dataset, the framework got an F1-score of 0.957 and an accuracy of 0.932. On the UNSW-NB15 dataset, it managed to achieve an F1-score of 0.943 and an accuracy of 0.928. And on the NSL-KDD dataset, the MAB framework reached an F1-score of 0.953 and an accuracy of 0.934. These results strongly confirm the framework's adaptability to diverse network environments and attack patterns. The experimental results demonstrate the MAB framework's outstanding performance across all datasets. The consistent performance across multiple datasets highlights its potential for real-world deployment scenarios.

5 Conclusion

This article presents a new framework based on the MAB for adaptive intrusion detection. It gets big improvements compared to traditional static approaches in dynamic network settings. After experiments, the framework shows excellent performance. On the CIC-IDS2017 benchmark, the system gets an outstanding F1-score of 0.969. It does better than the random selection baselines (0.939) and static Random Forest implementations (0.950), respectively. This performance advantage comes from the framework's dynamic adaptive capability that enables automatic detector switching. Among the MAB algorithms that were evaluated, TS turns out to be the most effective strategy. It manages to get an F1-score of 0.963 and an accuracy of 0.941. Its approach of Bayesian posterior sampling is really good at balancing exploration and exploitation in attack environments that are evolving rapidly.

In conclusion, this article has come up with a high-performance adaptive detection system based on MAB. Its advantages in accuracy and efficiency have been empirically validated. Also, it has carried out a comprehensive analysis of MAB algorithms, showing that TS is the best choice for dynamic environments.

This article's proposed framework shows obvious improvements in adaptive detection. However, its dependence on historical feedback mechanisms still faces challenges when it comes to identifying zero-day attacks. Also, its throughput scalability needs further optimization for terabit-scale networks. Future research needs to concentrate on strengthening zero-day detection capabilities by using hybrid machine learning approaches, making hardware acceleration better for extreme throughput requirements, and confirming the framework's generalization across wider threat landscapes. This study regards MAB-driven adaptive detection as a revolutionary approach for next-generation intrusion detection systems, presenting a scalable and self-learning approach.

References

1. Hammad A.A., Ahmed S.R., Abdul-Hussein M.K., Ahmed M.R., Majeed D.A., Algburi S.: 'Deep reinforcement learning for adaptive cyber defense in network security'. Proc. Cognitive Models and Artificial Intelligence Conf., 2024, pp. 292-297
2. Tariq Z.U.A., Baccour E., Erbad A., Guizani M., Hamdi M.: 'Network intrusion detection for smart infrastructure using multi-armed bandit based reinforcement learning in adversarial environment'. Proc. Int. Conf. Cyber Warfare and Security (ICCWS), 2022, pp. 75-82
3. Alqahtani H., Sarker I.H., Kalim A., Minhaz Hossain S.M., Ikhlaq S., Hossain S.: 'Cyber intrusion detection using machine learning classification techniques'. Proc. Int. Conf. Computing Science, Communication and Security (COMS2), Gujarat, India, 2020, pp. 121-131
4. Saranya T., Sridevi S., Deisy C., Chung T.D., Khan M.A.: 'Performance analysis of machine learning algorithms in intrusion detection system: A review', Procedia Computer Science, 2020, 171, pp. 1251-1260
5. Resende P.A.A., Drummond A.C.: 'A survey of random forest based methods for intrusion detection systems', ACM Computing Surveys (CSUR), 2018, 51, (3), pp. 1-36
6. Li W., Yi P., Wu Y., Pan L., Li J.: 'A new intrusion detection system based on KNN classification algorithm in wireless sensor network', Journal of Electrical and Computer Engineering, 2014, 2014, (1), 240217

7. Putrada A.G., Alamsyah N., Pane S.F., Fauzan M.N.: 'XGBoost for IDS on WSN cyber attacks with imbalanced data'. Proc. Int. Symp. Electronics and Smart Devices (ISESD), 2022, pp. 1-7
8. Gutierrez M.P., Kiekintveld C.: 'Bandits for cybersecurity: Adaptive intrusion detection using honeypots'. Proc. AAAI Workshop: Artificial Intelligence for Cyber Security, 2016
9. Yu K., Nguyen K., Park Y.: 'Flexible and robust real-time intrusion detection systems to network dynamics', IEEE Access, 2022, 10, pp. 98959-98969
10. Mahajan A., Teneketzis D.: 'Multi-armed bandit problems'. In: Foundations and Applications of Sensor Management, 2008, pp. 121-151