Hindawi
Security and Communication Networks
Volume 2022, Article ID 1812273, 14 pages
https://doi.org/10.1155/2022/1812273

Research Article

Attention-Based LSTM Model for IFA Detection in Named Data Networking

Xin Zhang, [1,2] **Ru Li**, [1,2] **and Wenhan Hou** [1,2]

[1] *Inner Mongolia Key Laboratory of Wireless Networking and Mobile Computing, Hohhot 010021, China*
[2] *College of Computer Science, Inner Mongolia University, Hohhot 010021, China*

Correspondence should be addressed to Ru Li; csliru@imu.edu.cn

Received 28 April 2022; Revised 21 May 2022; Accepted 28 May 2022; Published 21 July 2022

Academic Editor: Kai Wang

As one of the next generation networks, Named Data Networking (NDN) performs well on content distribution. However, it is vulnerable against a new type of denial-of-service (DoS) attacks, interest flooding attacks (IFAs), one of the fatal threats to NDN. The attackers request nonexist content to occupy the Pending Interest Table (PIT), and it causes the degradation of network performance. Because of the great harm and strong concealment of this attack, it is urgent to detect and throttle the attack. This paper proposes a detection mechanism based on Long Short-Term Memory (LSTM) with attention mechanism, which uses sequence with different treatments. Once IFA is detected, the Hellinger distance is used to recognize malicious Interest prefix. The simulation results show that the proposed scheme can resist IFA effectively compared to state-of-the-art schemes.

1. Introduction

The purpose of traditional network architecture based on TCP/IP is to meet the end-to-end data transmission, which cannot meet the diversified needs today. Therefore, the researchers began to study new network architectures. Information Centric Networking (ICN) [1] aims to build a new content-centric future network architecture, and it transforms the current host-centric communication mode into the content-centric network communication mode. Typical representative projects of ICN include information-oriented network architecture (Network of Information, NetInf) [2], publish/subscribe Internet routing paradigm, and publish/subscribe Internet topology (PSIRP/PURSUIT) [3], Data-Oriented Network Architecture (DONA) [4], Content Centric Networking (CCN) [5], and Named Data Networking (NDN) [6]. The most representative ICN architecture is NDN, which was proposed by Zhang Lixia of UCLA (University of California-Los Angeles) and Van Jacobson of Xerox PARC (Xerox Palo Alto Research Center) in 2010. The architecture of NDN is shown in Figure 1.

In the NDN network, there are two types of packets: Interest packet and Data packet [6]. The users send Interest packet to request content, and the returned content is called Data packet. There are three data structures in NDN: content store (CS), Pending Interest Table (PIT), and forwarding information base (FIB) [6]. NDN implements routing and forwarding via these three data structures:

(i) FIB: it stores the interface information pointing to the specified content, and the Interest packet is forwarded according to the FIB.
(ii) PIT: it records the unsatisfied Interest packet and the corresponding interfaces and can aggregate the Interest packets, and the Data packets are returned in the original way according to the interface information of the PIT.
(iii) CS: the router caches the received Data packet to realize intranetwork caching and reduces the delay for users to obtain data.

The NDN forwarding process of Interest packet and Data packet is shown in Figure 2.

When an NDN router receives an Interest packet, first it checks if CS has a matching data. If so, the router returns the Data Packet. Otherwise, the router checks whether PIT has a matching entry. If it exists, the router adds incoming

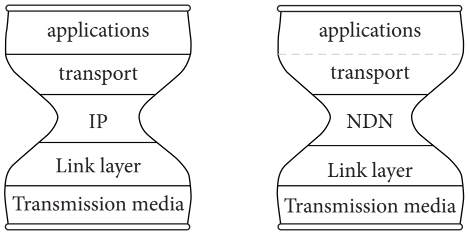

Figure 1: TCP/IP architecture vs NDN architecture [6, 7].

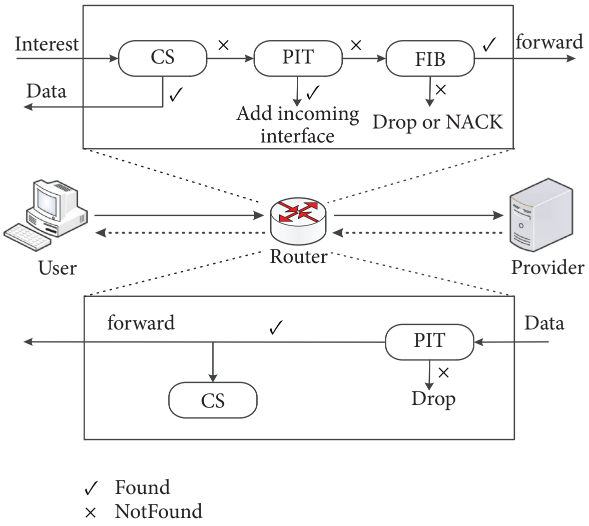

Figure 2: NDN forwarding process [6].

interface of the Interest packet to the entry. Otherwise, the router forwards Interest packet based on the FIB. When receiving a Data packet, the router first checks if PIT has a matching entry. If it exists, the router returns the Data packet based on the information of the PIT and caches the Data Packet. Otherwise, the router will drop the Data packet.

Denial of service (DoS) and distributed denial of service (DDoS) are rampant in the traditional TCP/IP architecture [8]. NDN can mitigate the impact of DDoS in TCP/IP architecture. However, the researchers discover a new type of DDoS attack called IFA [8]. As shown in Figure 3, the attacker forges a number of fake Interest packets to consume the memory resources of routers, which cause the degradation of network performance.

The IFA attack has great harm and strong concealment, and the researchers have tried various defend mechanisms, mainly including machine learning and statistical method. Due to the characteristics of network traffic, it is difficult to accurately identify attacks of a single time interval, resulting in low accuracy of attack detection. This paper uses past data through sliding window and proposes an attention-based Long Short-Term Memory (LSTM) [9] for IFA detection. Once IFA is detected, the Hellinger distance [10] is used to identify the malicious prefix.

The contributions of this paper are summarized as follows:

(1) This paper uses the LSTM model with attention mechanism to detect IFA by exploiting the past data sequence and with different treatments

(2) This paper proposes a Hellinger distance-based malicious Interest prefix identify mechanism

(3) The simulation results show that the scheme proposed can detect IFA effectively

The rest of the paper is organized as follows: Section 2 gives a review of related works. Section 3 presents detection mechanism and mitigation mechanism in detail. Section 4 gives an evaluation of the proposed mechanism and compares the proposed mechanism with state-of-the-art mechanism. Finally, Section 5 concludes the paper.

2. Related Works

Various literature works have been proposed on detecting and mitigating the IFA. Some approaches use machine learning to detect IFA. In paper [11], linear SVM and SVM with Gaussian radial basis kernel function were used to detect IFA. It consisted of two phases: the training phase and the test phase. In paper [12], the Isolation Forest was used to calculate the abnormal score of each Interest prefix at the end of each fixed time interval to detect abnormal Interest packet prefix. In paper [13], the deep reinforcement learning was used to detect IFA. In paper [14], the naïve Bayes (NB), J48 decision tree, multilayer perceptron with backpropagation (BP), and radial basis function (RBF) network were used to detect IFA. In paper [15], the authors used multilayer perceptron (MLP) with backpropagation (BP), radial basis function (RBF) network with particle swarm optimization (PSO), JAYA and teaching–learning-based optimization (TLBO), linear support vector machine (SVM), and fine k-nearest neighbours (KNN) to detect the attack. In paper [16], the authors used association rule algorithm to find the correlation between features and used decision tree algorithm to detect the attack.

Some approaches use the mathematical model to detect IFA. In paper [17], every NDN router computed the Gini impurity to detect IFA by measuring the Interest name in a router. In paper [18], the Theil index was used to detect IFA and the Interest packets were divided into groups by Theil entropy to evaluate the intragroup and intergroup difference of Interest name distribution. In paper [19], two traffic features were used to establish confidence interval, respectively, to detect IFA. In paper [20], the authors used mean and variance of packet hop counts to distinguish legitimate users from malicious users. In paper [21], the authors used hash-based security label to identify the malicious prefix. In paper [22], the authors used wavelet analysis to detect IFA. In paper [23], the routers used active queue management (AQM) to defend IFA. In paper [24], each edge router used token-based router monitoring policy (TRM) to mitigate the IFA by controlling the data requestors. The detection method used in the related work is shown in Table 1. The main drawback of existing IFA detection method is counting the traffic information on a fixed time interval, which ignores the temporal relationship of traffic.

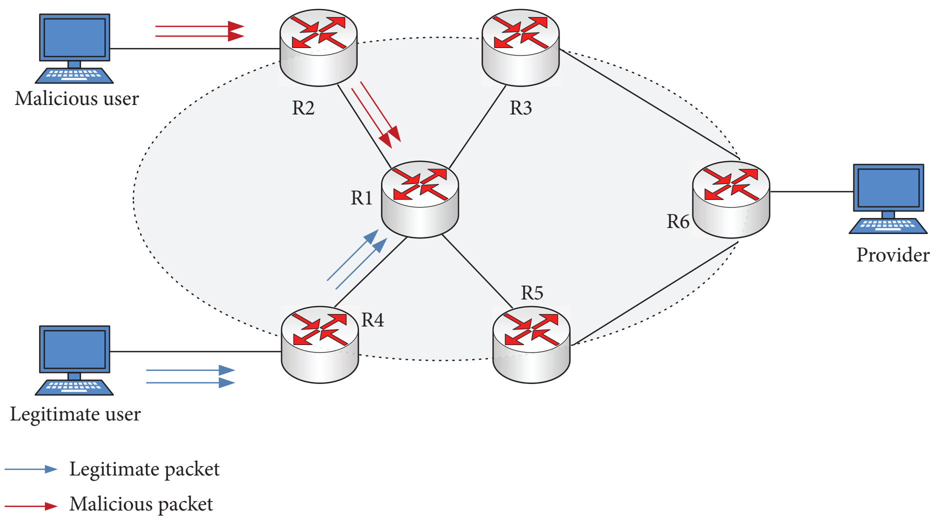

Figure 3: IFA sample.

Table 1: Comparison between related paper.

Paper	Year	Offline	Online	Detection method
[12]	2021	✗	✓	Isolation forest
[16]	2021	✓	✗	Association rules + decision tree
[23]	2021	✗	✓	AQM
[24]	2021	✗	✓	Token
[19]	2020	✗	✓	Confidence interval
[21]	2020	✗	✓	Hash
[11]	2019	✓	✓	SVM
[18]	2019	✗	✓	Theil index
[25]	2019	✗	✓	Hypothesis testing
[26]	2019	✗	✓	AQM
[13]	2020	✗	✓	Deep reinforcement learning
[17]	2018	✗	✓	Gini impurity
[14]	2019	✓	✗	MLP with BP RBF classifier J48 Naive Bayes
[20]	2018	✗	✓	Mean-variance
[15]	2017	✓	✗	MLP with BP RBF with PSO RBF with JAYA RBF with TLBO SVM linear Fine KNN
[22]	2017	✗	✓	Wavelet analysis

3. Detection Mechanism Based on Attention Mechanism with LSTM

This section gives an overview of proposed defend mechanism, detection mechanism, and mitigation mechanism.

3.1. Overview. The defend mechanism mainly consists of five parts, the data collection module, the data preprocessing module, the detection module, the response module, and the mitigation module, as shown in Figure 4.

In the data collection module, the traffic data is collected and it is then input to the preprocessing module. In the preprocessing module, the traffic characteristics are extracted. The traffic characteristics are used to detect IFA in the detection module. Once IFA is detected, the response module will start identify the malicious prefix. Finally, the mitigation module uses malicious prefix to limit the malicious Interest packet.

3.2. Long Short-Term Memory. Deep learning is popular and is used in various applications. Recurrent neural network (RNN) [27] is a type of deep learning methods, which can be used to detect anomaly. However, there is a gradient vanishing problem in RNN [28]. Long Short-Term Memory (LSTM) [9] is an improved version of RNN, which solves the problem of RNN. The LSTM structure is shown in Figure 5.

It mainly includes three structures, input gate, forget gate, and output gate, which are used to update the LSTM cell as follows [9]:

$$\begin{aligned}
f_t &= \sigma\left(W_f\left[h_{t-1}, x_t\right] + b_f\right),\\
i_t &= \sigma\left(W_i\left[h_{t-1}, x_t\right] + b_i\right),\\
\widetilde{C}_t &= \tanh\left(W_C\left[h_{t-1}, x_t\right] + b_C\right),\\
C_t &= f_t * C_{t-1} + i_t * \widetilde{C}_t,\\
o_t &= \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right), \text{ and}\\
h_t &= o_t * \tanh\left(C_t\right),
\end{aligned} \tag{1}$$

where W is the weight, b is the bias, h_t is the hidden state at time step t, and x_t is the input at time step t.

3.3. Attention Mechanism. The Attention mechanism is inspired by human attention behaviour and is well applied to deep learning.

In paper [29], the attention mechanism was proposed. Given an input $X = [x_1, x_2, \ldots, x_N] \in R^{D\times N}$, where N is the length of input, $x_n \in R^D$, $n \in [1, N]$, and D is the number of dimensions in each time step, the calculation of the attention mechanism is divided into two steps: first calculate the attention probability of all input and then calculate the

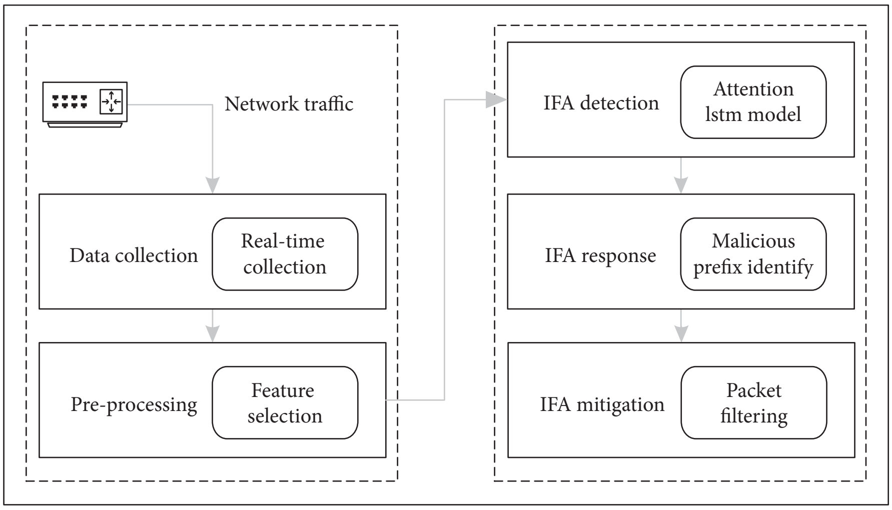

Figure 4: The architecture of defend mechanism.

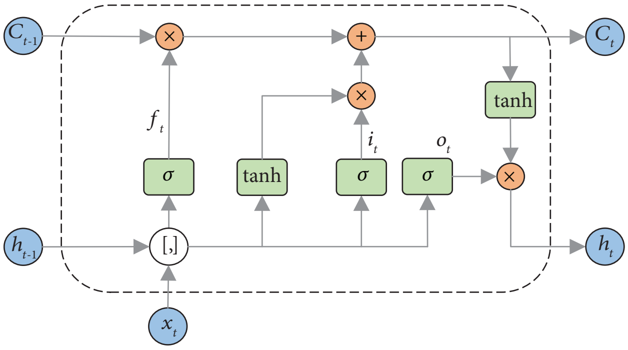

Figure 5: An LSTM cell structure [9].

weighted average of the input information according to the attention probability.

3.4. Detection Mechanism. This section presents the detection mechanism in detail. First, some used notations are listed and some features are defined. The notations used are listed in Table 2.

Definition 1. (Router PIT Utilization Size). It denotes the number of PIT entries in PIT during one time slice.

$$U(t_i, R_j) = e(t_i, R_j). \tag{2}$$

Definition 2. (Router Interest Satisfaction Ratio). It denotes the number of Data packets received to the number of Interest packets received in one time slice.

$$S(t_i, R_j) = \frac{\varphi(\phi(t_i, R_j))}{\phi(t_i, R_j)}. \tag{3}$$

Definition 3. (Router Interest Request Frequency). It denotes the number of Interest packets received in one time slice.

$$I(t_i, R_j) = \phi(t_i, R_j). \tag{4}$$

Definition 4. (Router Data Reply Frequency). It denotes the number of Data packets replied in one time slice.

$$r(t_i, R_j) = \varphi(\phi(t_i, R_j)). \tag{5}$$

The feature calculation is shown in Algorithm 1.

The detection mechanism detects IFA through a sliding window, as shown in Figure 6.

A network traffic formally as a time series: $Z = \{z^1, z^2, \ldots, z^i, \ldots, z^F\}$, which consists of F time steps. $z^i\,(1 \le i \le F)$ represents the i th time step. For each sliding window, which consists of φ time steps, the detection model is used to classify the sliding window as legitimate or malicious.

Figure 7 shows the LSTM with attention mechanism for IFA detection. The attention mechanism can improve the performance of LSTM by discriminatively utilizing each step of hidden state information [30]. Therefore, this paper uses the traditional LSTM with attention mechanism to detect IFA. The hidden states of each step are multiplied with attention weights.

In LSTM layer, the input of each step is mapped to a hidden state.

$$h_i = \mathrm{LSTM}(z_i), \quad i \in [1, F], \tag{6}$$

where h_i is the hidden state at time step i and z_i is the input at time step i.

In attention layer, the hidden state of each step is input to a subsequent attention layer. It takes the form as follows [31]:

$$\begin{aligned} \mathscr{H} &= \sum_{t=1}^{N} \alpha_t h(t) \text{ and} \\ \alpha_t &= \frac{\exp(g_t(W_t, h(t)))}{\sum_{t=1}^{N} \exp(g_t(w_t, h(t)))}, \end{aligned} \tag{7}$$

Table 2: Notation used.

Notation	Description
t_i	The i-th time slice
R_j	The j-th router
$\phi(t_i, R_j)$	The number of receiving Interests of the j-th router in the i-th time slice
$\varphi(\phi(t_i, R_j))$	The number of receiving corresponding Data packets
$e(t_i, R_j)$	The number of PIT entry of router j at the i-th time slice

```
Input:
    $\varepsilon$ ▷ The time slice size
Output:
    $i$ ▷ The request frequency
    $r$ ▷ The reply frequency
    $s$ ▷ The satisfaction ratio
procedure IncomingInterest(slice $\varepsilon$)
    $i \longrightarrow i + 1$
end procedure
procedure IncomingData(slice $\varepsilon$)
    $r \longrightarrow r + 1$
end procedure
$s \longrightarrow r/i$
return $i$ $r$ s
```

Algorithm 1: Interest features computing.

```
Input:
    $\varepsilon$ ▷ The time slice size
    $\varphi$ ▷ The sliding window size
    $Thr$ ▷ Detection threshold
Output:
    Detection result
Compute the metrics during time slice $\varepsilon$
for the consecutive time step with length $\varphi$ do
    fed the sequence $Z$ to the detection model
    $y = \mathrm{LSTM}Att(Z)$
    if $y > Thr$ then
        return legitimate
    else
        return malicious
    end if
end for
```

Algorithm 2: LSTM with attention mechanism-based detection.

where α_t is the weight for each time step and $g_t(\cdot)$ is a fully connected layer with ReLU activation and parameters W_t.

The illustration of attention mechanism is shown in Figure 8.

In output layer, the attention layer results $\mathscr{H}$ is input to a fully connected layer with sigmoid activation to obtain the final result.

$$\text{output} = \text{simoid}(v). \tag{8}$$

The detection mechanism is shown in Algorithm 2.

The algorithm works as mentioned in the following steps:

Step (1): count the traffic information in time slice ε, use Algorithm 1

Step (2): when the sliding window size is φ, fed to the detection model, get output y

Step (3): if the detection result is legitimate, forward the sliding window and return to Step (2)

Step (4): if the detection result is malicious, trigger the malicious prefix identification mechanism

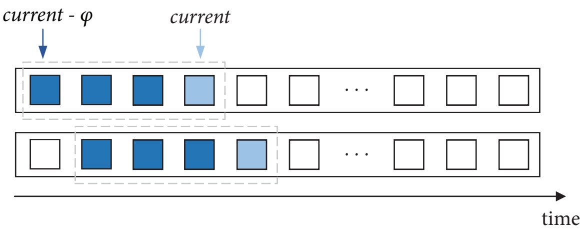

Figure 6: The sliding window.

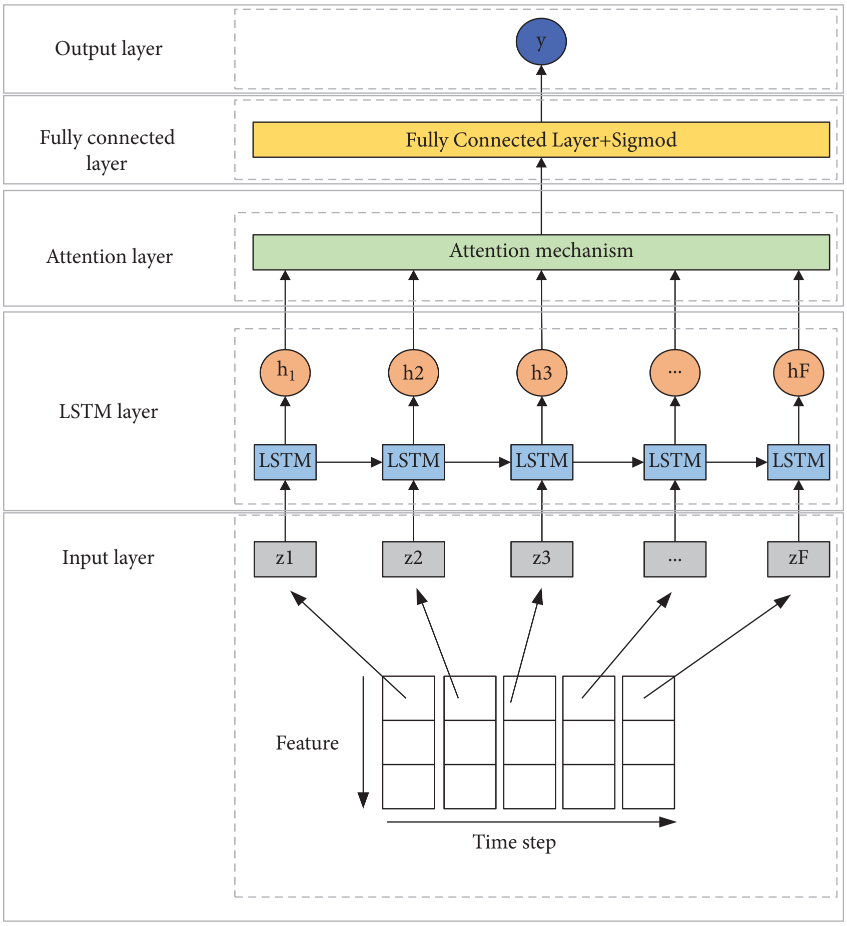

Figure 7: The LSTM with attention mechanism.

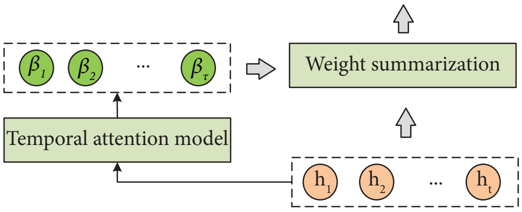

Figure 8: Illustration of temporal attention mechanism.

3.5. Response Mechanism. This paper recognizes the malicious Interest prefixes based on the Hellinger distance [10]. The Hellinger distance is used to measure the deviation between two probability distributions independent of parameters.

$$H(\mathbb{P},\mathbb{Q}) = \frac{1}{\sqrt{2}}\sqrt{\sum_{i=1}^{n}\left(\sqrt{p_i}-\sqrt{q_i}\right)^2}, \quad p_i \geq 0;\ q_i \geq 0, \tag{9}$$

where $\mathbb{P}$ and $\mathbb{Q}$ are two probability distributions, $\mathbb{P}$ and $\mathbb{Q}$ are n-tuples $(p_1, p_2, .., p_n)$, and $(q_1, q_2, .., q_n)$, $\sum_i p_i = 1$, and $\sum_i q_i = 1$.

The malicious prefix recognition process is shown in Algorithm 3.

3.6. Mitigation Mechanism. When malicious prefixes are recognized, the router will send notification packet that

```
Input:
    Interest prefix distribution when IFA is detected: $\mathbb{P}$
    Interest prefix distribution before IFA is detected: $\mathbb{Q}$
    Interest prefix set: $I$
Output:
    Malicious prefix set
$\mathbb{Q}' = \mathbb{Q}$
for prefix$_i \in I$ do
    $\mathbb{Q}'_i = \mathbb{P}_i$
    calculate the Hellinger distance $H(\mathbb{Q}'_i, \mathbb{Q})$
    if $H(\mathbb{Q}'_i, \mathbb{Q}) > thr$ then
        add prefix$_i$ to malicious prefix set
    end if
end for
return malicious prefix set
```

Algorithm 3: Hellinger distance-based malicious prefix recognition.

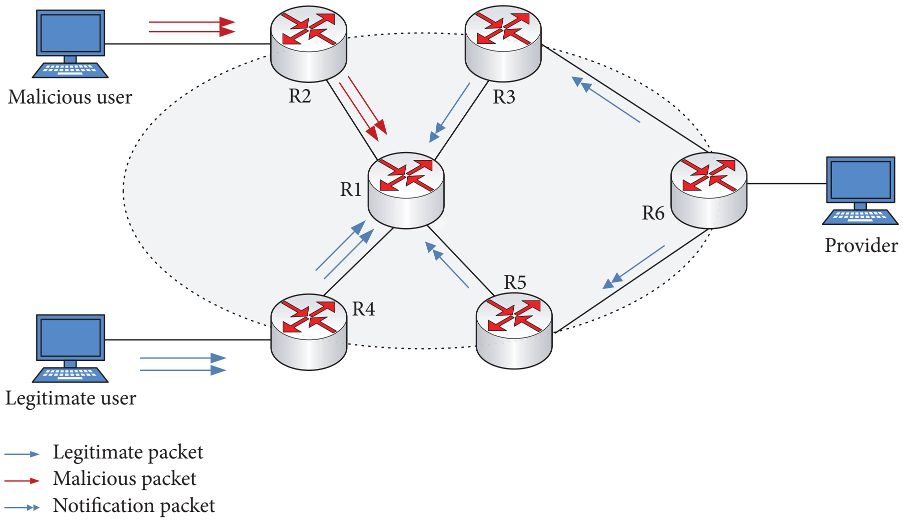

Figure 9: IFA mitigation sample.

includes the malicious prefixes to the downstream router, as shown in Figure 9. The downstream routers extract the malicious prefix and limit its sending rate when receiving the notification packet.

4. Performance Evaluation

In order to evaluate the performance of the proposed scheme, this paper conducts a set of simulations in ndnSIM [32]. Then, this paper compares the proposed scheme with the state-of-the-art defend scheme. The simulations parameters are shown in Table 3.

This paper considers tree topology as shown in Figure 10. The tree topology which is one of the most severely affected by the IFA is widely used in detection mechanism evaluation of IFA.

In tree topology, *Rx* denotes the NDN router, *Cx* denotes the legitimate user, *Px* denotes the data provider, and *Ax* denotes the malicious user. The red lines denote connections between the malicious user and NDN router, the green lines denote connections between the legitimate user and NDN router, the black lines denote connections between NDN routers, and the blue lines denote the connections between the data provider and NDN router.

In tree topology, there are 9 legitimate users and 7 malicious users. The simulation lasts 800s. The legitimate users issue Interest with the Zipf-Mandelbrot distribution [33], and the malicious users issue Interest with uniform distribution. In Zipf-Mandelbrot distribution, the content items with k-th rank in the whole content popularity ranking list are requested with probability $\{q_k\}_{k=1,2\ldots K}$, where $q_k = c/(k+q)^s$, $c = \left\{\sum_{k=1}^{K} 1/(k+q)^s\right\}^{-1}$, K is the size of the popularity list, and q and s are parameters.

4.1. Performance Metrics. The performance of detection mechanism is evaluated by the confusion matrix, as shown in Figure 11, where TP represents the number of abnormal traffic, which is classified as abnormal, TN represents the number of normal traffic, which is classified as normal, FP represents the

Table 3: Simulation parameters.

Parameters	Value
Legitimate request distribution	Zipf-Mandelbrot
Malicious request distribution	Uniform
Number of content types	1000
Malicious request rate	100
Legitimate request rate	100
Lifetime of PIT entries (second)	1
Attack time (second)	400
Simulation time	800

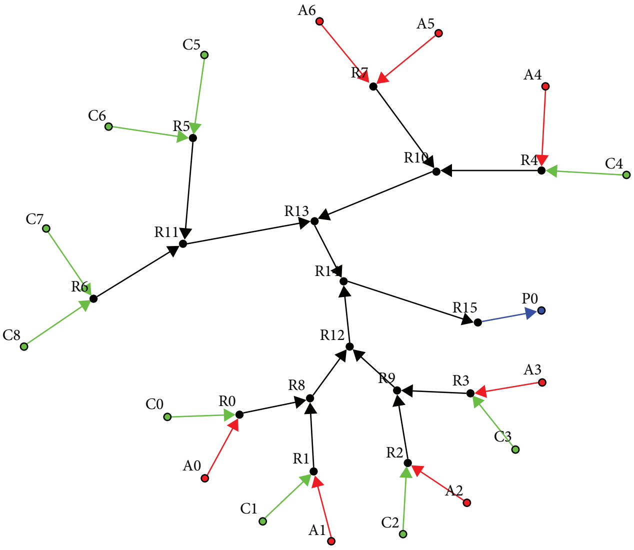

Figure 10: Tree topology.

number of normal traffic, which is classified as abnormal, and FN represents the number of abnormal traffic, which is classified as normal. This paper compares the detection mechanism with SVM and LSTM from the following metrics:

(i) Interest satisfaction ratio: it is defined as the ratio between the number of Data packets received and the number of Interest packets sent.

(ii) PIT size: it is defined as the number of entries in the PIT.

(iii) Accuracy: it is defined as the overall performance of the model and is calculated as follows:

$$\text{accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}}. \tag{10}$$

(iv) Recall: it is defined as the proportion of attack samples that are correctly identified as attacks, and it is calculated as follows:

$$\text{recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{11}$$

4.2. Hyperparameter Tuning. The detection model's architectures are built using Pytorch in Python on a machine with 32 GB RAM. This paper trains detection model for 50 epochs with Adam optimizer [34] at a learning rate of 0.001.

4.3. Loss Function. The binary cross entropy is a loss function that is used in binary classification problems. The objective of the detection mechanism is to label time window as normal or abnormal; therefore, this paper uses binary cross entropy loss function for training the LSTM and LSTM with attention mechanism, which is computed as follows:

$$L = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log\left(p\left(y_i\right)\right) + \left(1 - y_i\right) \cdot \log\left(1 - p\left(y_i\right)\right). \tag{12}$$

where y_i is the binary label and N is the total number of samples in training set.

4.4. Impact of the IFA. Attack intensity (λ) is defined as the ratio of malicious user's sending rate to the legitimate user's sending rate. In this section, this paper evaluates the impact of the IFA and considers two types of routers: the router only connected to legitimate user and the router connected to legitimate user and malicious user.

In Figure 10, this paper evaluates PIT size of the routers $R11$, $R10$, and $R8$ under IFA and evaluates the Interest satisfaction ratio of normal users under the IFA.

Figure 12 shows the PIT size under IFA with different attack intensities. When there is no attack, the routers have a constant PIT size. When IFA is launched at the 400th second, the PIT size begins to increase and the greater the attack intensity, the greater the PIT size. The impact on PIT size is also different for routers in different locations; the router R11 is least affected by the attack because it is not connected to a malicious user; the router R10 is greatly affected by the attack because it has the most connections with malicious users.

Figure 13 shows the Interest satisfaction ratio of normal user under IFA with different attack intensities. The Interest satisfaction ratio is stable without IFA, and the Interest packet sent by the user can receive the corresponding Data packet. At the 400th second, the IFA is launched, the Interest packets sent by users can hardly receive the corresponding Data packets, and the Interest satisfaction ratio decreases instantaneously. Moreover, with the increase of attack intensity, more malicious Interest packets are sent and the impact on Interest satisfaction ratio of normal users is greater.

4.5. Performance of Detection Mechanism. In this section, this paper compares our detection mechanism with SVM and LSTM from detection accuracy and recall. Then, this paper evaluates the defend mechanism from Interest satisfaction ratio and PIT size with expired-PIT-based defend mechanism [35].

Firstly, the learning rate and batch size used in this paper are introduced. This paper selects learning rate and batch size by comparing the detection accuracy. The learning rate is 0.001, 0.005, and 0.01, respectively. The batch size is 512, 256, and 128, respectively. The simulation results of different learning rates and batch sizes on the detection accuracy are shown in Figures 14–16, respectively.

	normal	abnormal
normal	TN	FP
abnormal	FN	TP

Figure 11: Confusion matrix.

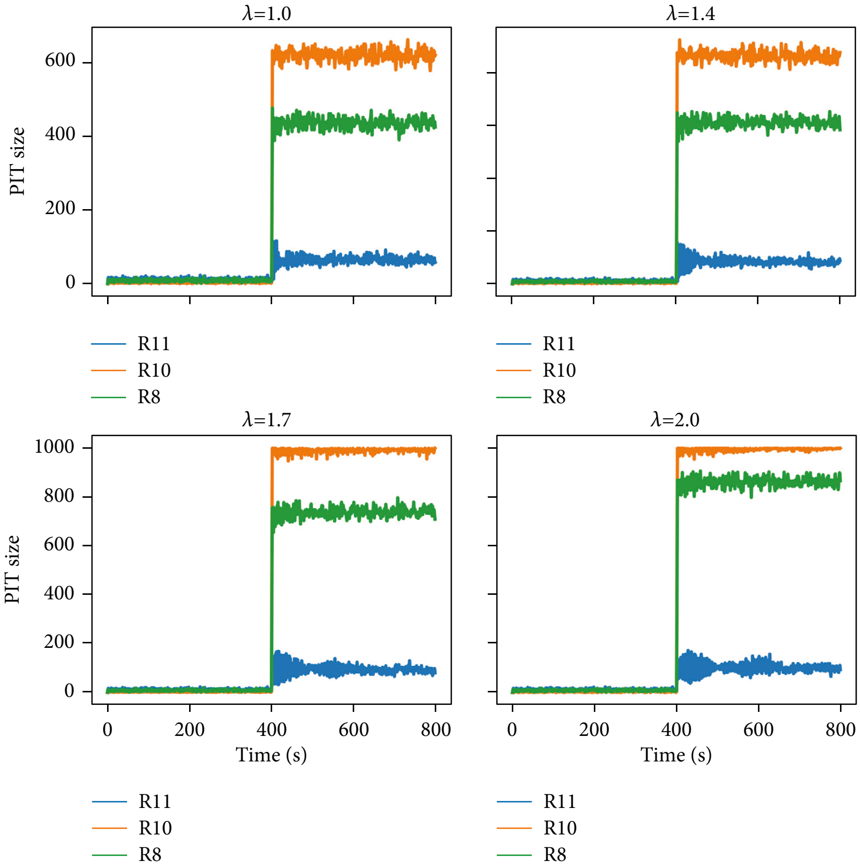

Figure 12: PIT size under IFA.

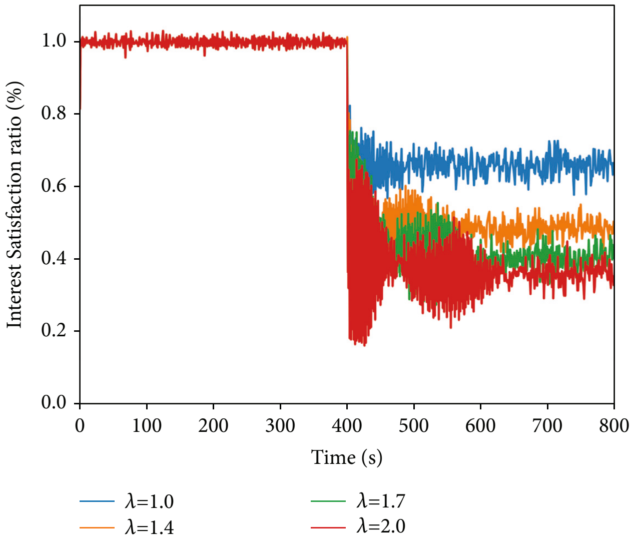

Figure 13: Interest satisfaction ratio under IFA.

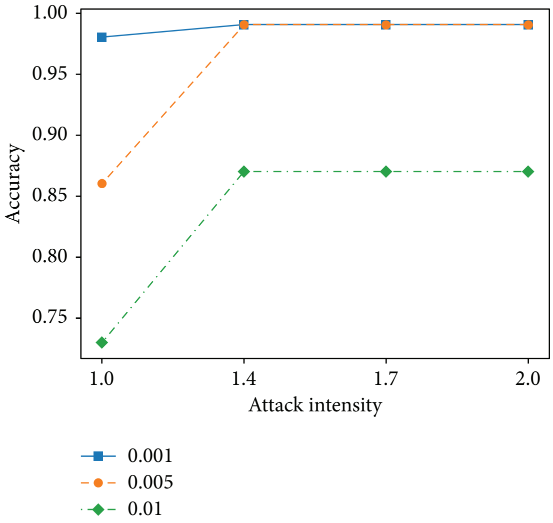

Figure 14: Batch size 512.

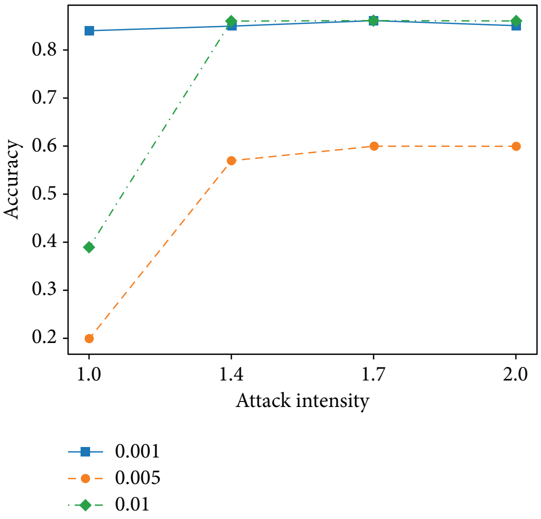

Figure 15: Batch size 256.

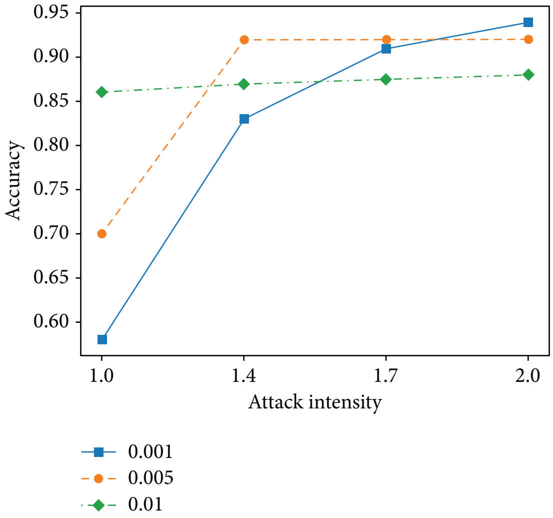

Figure 16: Batch size 128.

Figure 14 shows the detection accuracy under different attack intensities with different learning rates when the batch size is 512. When the learning rate is 0.001, the accuracy is the highest.

Figure 15 shows the detection accuracy under different attack intensities with different learning rates when the batch size is 256. When the learning rate is 0.001, the accuracy is the highest.

Figure 16 shows the detection accuracy under different attack intensities with different learning rates when the batch size is 128.

Finally, this paper sets the batch size 512 and the learning rate is 0.001. As shown in Figures 17 and 18, with the increase in the number of epochs, the accuracy increases and the loss decreases. When the epochs are equal to 50, the model tends to be stable.

Next, this paper compares the accuracy and recall of the detection mechanism with SVM and LSTM, and the results are shown in Figures 19 and 20.

Figure 19 shows the detection accuracy of the proposed detection mechanism under different attack intensities. Compared with LSTM and SVM, the detection mechanism proposed in this paper has the highest accuracy.

Figure 20 shows the recall of the proposed detection mechanism under different attack intensities. Compared with LSTM and SVM, the detection mechanism proposed in this paper has the highest recall.

4.6. Performance of Mitigation Mechanism. This section evaluates our mitigation mechanism on the Interest satisfaction ratio and PIT size.

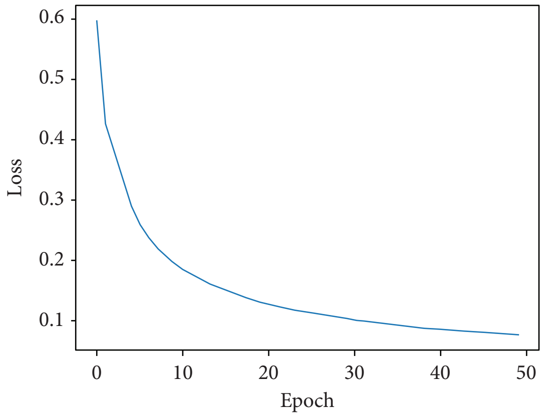

Figure 17: Loss with epoch.

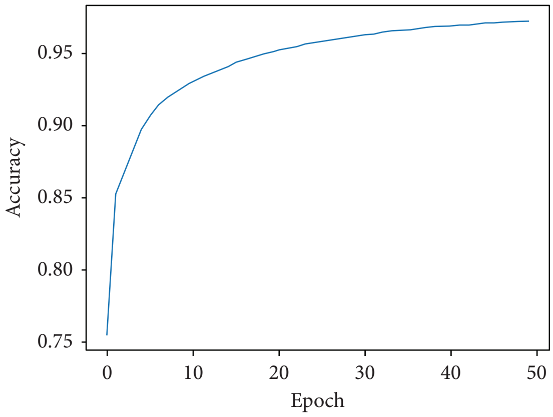

Figure 18: Accuracy with epoch.

Figure 21 shows the Interest satisfaction ratio with the proposed defend mechanism and expired-PIT-based defend mechanism under attack. When the malicious users launch IFA at the 400th second, the Interest satisfaction ratio drops rapidly. Under high attack intensity, the proposed detection mechanism quickly detects the attack and limits the sending of malicious packets and the Interest satisfaction ratio returns to the normal level. This paper also tests the impact of the detection mechanism on the burst traffic of normal users, and the proposed detection mechanism will not misjudge the burst traffic of normal users.

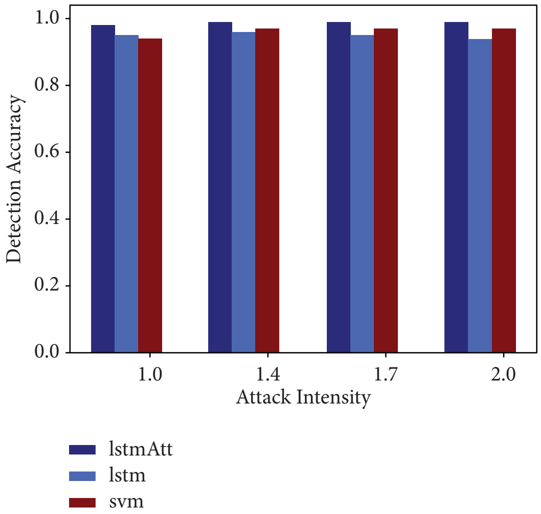

Figure 19: Detection accuracy.

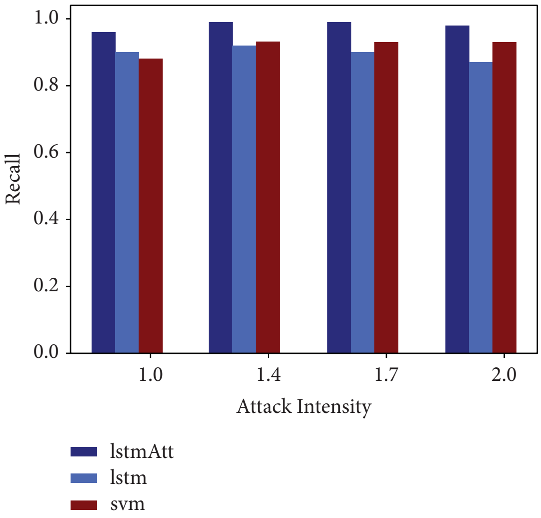

Figure 20: Recall.

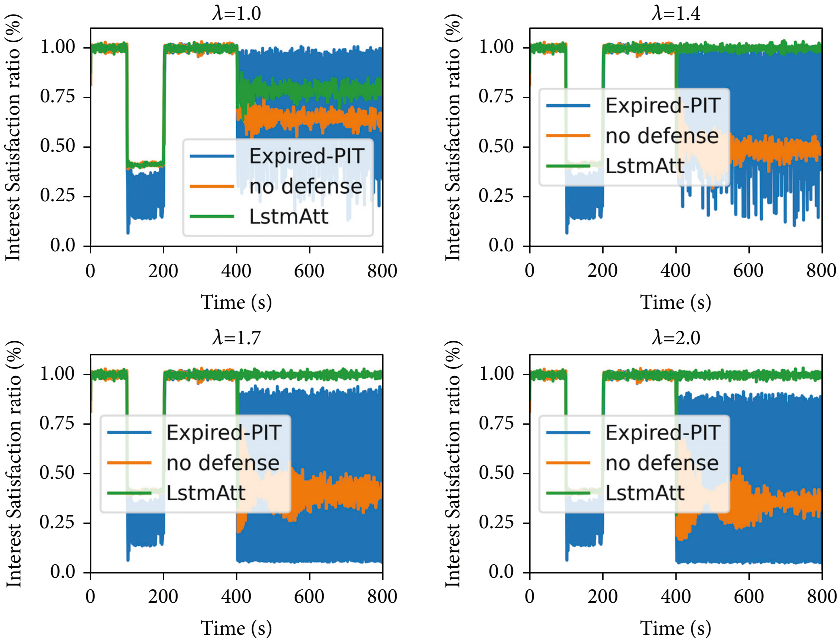

Figure 21: Interest satisfaction ratio with different defend mechanisms.

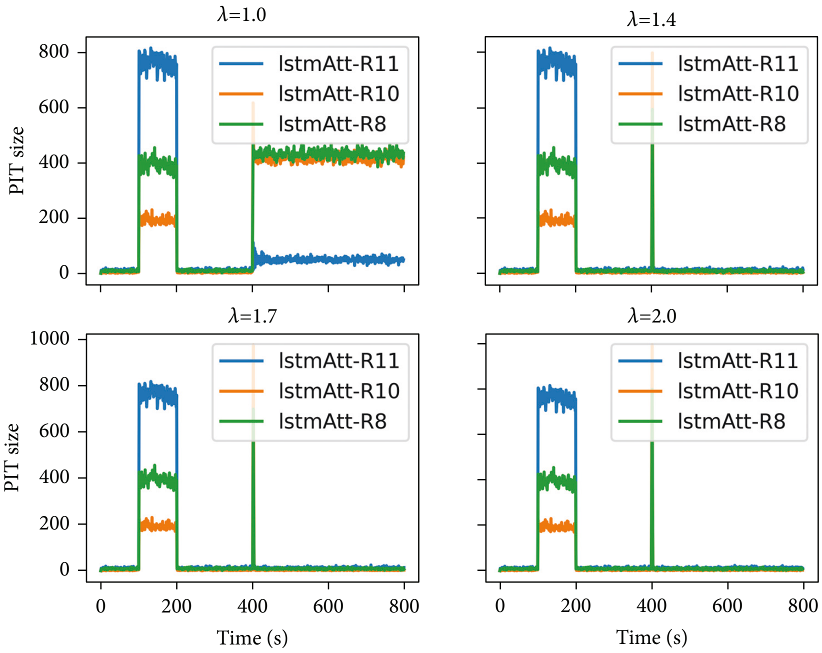

Figure 22: PIT size with the proposed defend mechanism.

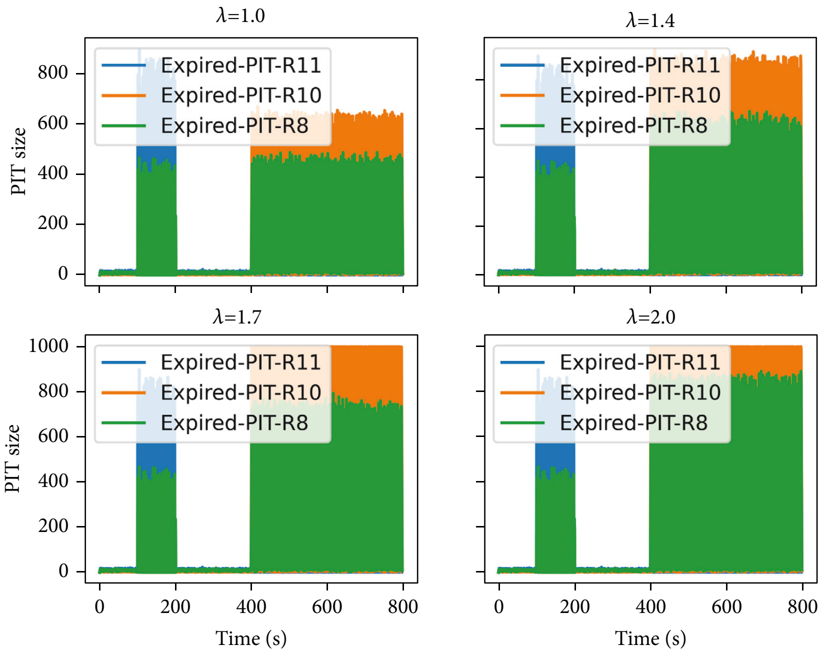

Figure 23: PIT size with expired-PIT-based defend mechanism.

Figures 22 and 23 show the PIT size with the proposed defend mechanism and expired-PIT-based defend mechanism under attack. When the attacker starts the attack at the 400th second, the PIT size rises rapidly. Under high attack intensity, the detection mechanism quickly detects the attack of different attack intensities and limits the sending of malicious packets and the PIT size returns to the normal level.

5. Conclusions

This paper proposes a defend mechanism for Interest flooding attack in NDN. The defend consists of three parts: detection, response, and mitigation. The LSTM with attention mechanism is used to detect IFA; once IFA is detected, the Hellinger distance is used to identify malicious Interest packet prefix. Finally, the malicious prefix is sent to the downstream routers to cooperate to limit the attack. The experimental results show that the LSTM with attention mechanism shows better performance than the LSTM and SVM. In future work, this paper will consider multiple attacks in NDN, such as collusive attack, low-rate IFA, and large-scale topology.

Data Availability

The data used to support the findings of this study have not been made available because the data also form part of an ongoing study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61862046); Huhhot Science & Technology Plan (2021-KJXM-TZJW-04); and the Science and Technology Program of Inner Mongolia Autonomous Region (2020GG0188).

References

[1] G. Xylomenos, C. N. Ververidis, V. A. Siris et al., "A survey of information-centric networking research," *IEEE communications surveys & tutorials*, vol. 16, no. 2, pp. 1024–1049, 2014.

[2] B. Ahlgren, M. D'ambrosio, and C. Dannewitz, "Second netinf architecture description," *4WARD EU FP7 Project*, Deliverable D-6.2 v2. 0, 2010.

[3] M. Ain, D. Trossen, and P. Nikander, "D2. 3–architecture definition, component descriptions, and requirements," *Deliverable*, PSIRP 7th FP EU-funded project, vol. 11, 2009.

[4] T. Koponen, M. Chawla, B.-G. Chun et al., "A data-oriented (and beyond) network architecture," *s* in *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communication*, vol. 37, no. 4, pp. 181–192, Kyoto, Japan, August 2007.

[5] V. Jacobson, M. Mosko, D. Smetters, and J. Garcia-Luna-Aceves, "Content-centric networking," *Palo Alto Research Center*, White Paper, pp. 2–4, 2007.

[6] L. Zhang, A. Afanasyev, J. Burke et al., "Named data networking," *ACM SIGCOMM - Computer Communication Review*, vol. 44, no. 3, pp. 66–73, 2014.

[7] H. Zhang, Y. Li, Z. Zhang, A. Afanasyev, and L. Zhang, "NDN host model," *ACM SIGCOMM - Computer Communication Review*, vol. 48, no. 3, pp. 35–41, 2018.

[8] P. Gasti, G. Tsudik, E. Uzun, and L. Zhang, "DoS and DDoS in named data networking," in *Proceedings of the 2013 22nd International Conference on Computer Communication and Networks (ICCCN)*, pp. 1–7, Nassau, Bahamas, August 2013.

[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] A. Basu, A. Mandal, and L. Pardo, "Hypothesis testing for two discrete populations based on the Hellinger distance," *Statistics & Probability Letters*, vol. 80, no. 3-4, pp. 206–214, 2010.

[11] T. Zhi, Y. Liu, J. Wang, and H. Zhang, "Resist interest flooding attacks via entropy-SVM and jensen-shannon divergence in information-centric networking," *IEEE Systems Journal*, vol. 14, no. 2, pp. 1776–1787, 2020.

[12] G. Xing, J. Chen, R. Hou et al., "Isolation forest-based mechanism to defend against interest flooding attacks in named data networking," *IEEE Communications Magazine*, vol. 59, no. 3, pp. 98–103, 2021.

[13] J. Zhou, J. Luo, L. Deng, and J. Wang, "Defense mechanism of interest flooding attack based on deep reinforcement learning," in *Proceedings of the 2020 3rd International Conference on Hot Information-Centric Networking (HotICN)*, pp. 65–70, Hefei, China, December 2020.

[14] N. Kumar, A. K. Singh, and S. Srivastava, "Feature selection for interest flooding attack in named data networking," *International Journal of Computers and Applications*, vol. 43, no. 6, pp. 537–546, 2021.

[15] N. Kumar, A. K. Singh, and S. Srivastava, "Evaluating machine learning algorithms for detection of interest flooding attack in named data networking," in *Proceedings of the 10th International Conference on Security of Information and Networks*, pp. 299–302, Jaipur, India, October 2017.

[16] Z. Wu, R. Zhang, and M. Yue, "A method for joint detection of attacks in named data networking," *Journal of Computer Research and Development*, vol. 58, no. 3, pp. 569–582, 2021.

[17] T. Zhi, H. Luo, and Y. Liu, "A Gini impurity-based interest flooding attack defence mechanism in NDN," *IEEE Communications Letters*, vol. 22, no. 3, pp. 538–541, 2018.

[18] R. Hou, M. Han, J. Chen et al., "Theil-based countermeasure against interest flooding attacks for named data networks," *IEEE Network*, vol. 33, no. 3, pp. 116–121, 2019.

[19] Z. Wu, W. Feng, M. Yue, X. Xu, and L. Liu, "Mitigation measures of collusive interest flooding attacks in named data networking," *Computers & Security*, vol. 97, Article ID 101971, 2020.

[20] Y. Nakatsuka, J. L. Wijekoon, and H. Nishi, "FROG: a packet hop count based DDoS countermeasure in NDN," in *Proceedings of the 2018 IEEE Symposium on Computers and Communications (ISCC)*, pp. 00492–00497, Natal, Brazil, June 2018.

[21] J. Dong, K. Wang, W. Quan, and H. Yin, "InterestFence: simple but efficient way to counter interest flooding attack," *Computers & Security*, vol. 88, Article ID 101628, 2020.

[22] Y. Xin, Y. Li, W. Wang, W. Li, and X. Chen, "Detection of collusive interest flooding attacks in named data networking using wavelet analysis," in *Proceedings of the 2017 IEEE Military Communications Conference (MILCOM)*, pp. 557–562, Baltimore, MD, USA, October 2017.

[23] A. Benarfa, M. Hassan, E. Losiouk, and A. M. B. M. Compagno, "ChoKIFA+: an early detection and mitigation approach against interest flooding attacks in NDN," *International Journal of Information Security*, vol. 20, no. 3, pp. 269–285, 2021.

[24] D. Qu, G. Lv, S. Qu, and H. Y. Z. Shen, "An effective and lightweight countermeasure scheme to multiple network attacks in NDN," *IEEE/ACM Transactions on Networking*, vol. 30, no. 2, pp. 515–528, 2022.

[25] T. Nguyen, H.-L. Mai, R. Cogranne et al., "Reliable detection of interest flooding attack in real deployment of named data networking," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 9, pp. 2470–2485, 2019.

[26] A. Benarfa, M. Hassan, A. Compagno, E. Losiouk, M. B. Yagoubi, and M. Conti, "ChoKIFA: a new detection and mitigation approach against interest flooding attacks in

NDN," , Springer, Bologna, Italy, 2019pp. 53–65, Lecture Notes in Computer Science, vol. 11618.

[27] K. Cho, B. V. Merrienboer, C. Gulcehre et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, https://arxiv.org/abs/1406.1078.

[28] J. Schmidhuber, "Deep learning in neural networks: an overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[29] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, https://arxiv.org/abs/1409.0473.

[30] G. Zhang, V. Davoodnia, A. Sepas-Moghaddam, Y. Zhang, and A. Etemad, "Classification of hand movements from EEG using a deep attention-based LSTM network," *IEEE Sensors Journal*, vol. 20, no. 6, pp. 3113–3122, 2020.

[31] Y. Ding, Y. Zhu, J. Feng, P. Zhang, and Z. Cheng, "Interpretable spatio-temporal attention LSTM model for flood forecasting," *Neurocomputing*, vol. 403, pp. 348–359, 2020.

[32] A. Afanasyev, I. Moiseenko, and L. Zhang, "ndnSIM: NDN simulator for NS-3," 2012, https://named-data.net/wp-content/uploads/TRndnsim.pdf.

[33] Z. K. Silagadze, "Citations and the Zipf-Mandelbrot's law," 1999, https://arxiv.org/abs/physics/9901035.

[34] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.

[35] V. G. Vassilakis, B. A. Alohali, I. D. Moscholios, and M. D. Logothetis, "Mitigating distributed denial-of-service attacks in named data networking," in *Proceedings of the 11th Advanced International Conference on Telecommunications (AICT)*, pp. 18–23, Brussels, Belgium, June 2015.