Indonesian Journal of Electrical Engineering and Computer Science
Vol. 39, No. 3, September 2025, pp. 1913~1922
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v39.i3.pp1913-1922

Jellyfish optimized deep learning framework for cache pollution attack detection in NDN environment

Varghese Jensy Babu[1], Victor Jose Marianthiran[2]
[1]Department of Computer Science and Engineering, Vidya Academy of Science and Technology, Thrissur, India
[2]Department of Computer Science and Engineering, Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College (Autonomous), Chennai, India

Article Info

Article history:

Received May 9, 2024
Revised Apr 10, 2025
Accepted Jul 3, 2025

Keywords:

Attack detection
Cache pollution attack
Deep learning
Jellyfish search optimization
Named data network
Network security

ABSTRACT

Named data networking (NDN) is a promising paradigm that replaces the traditional connection-based model with a content-based approach for future Internet infrastructures, allowing data retrieval by unique names. However, NDN faces threats like cache pollution attacks (CPA) which can lead to increased cache misses and data retrieval delays, and pose significant risks to its efficiency and security. In this paper, a novel jellyfish optimized deep learning (DL) framework for cache pollution attack detection in NDN environment (DSODAL) technique has been proposed to detect the CPA attack with high accuracy. To detect CPA in NDN, a dual-gate attention-based long short-term memory (LSTM) (DA-LSTM) network is used which is optimized using the jellyfish search optimization (JSO) algorithm. The DA-LSTM analyzes request sequences to identify malicious patterns, enhancing cache pollution detection. Nodes manage these requests using the content store (CS) for caching frequently accessed data, optimizing retrieval efficiency, and the pending interest table (PIT) to track and process incoming requests. The DA-LSTM analyzes request sequences to identify malicious patterns and detect CPA attacks. The DSODAL approach performance is evaluated using accuracy, precision, recall, F1-score, average delay time, and mean square error (MSE). The DSODAL model advances the overall accuracy by 1.74%, 2.34%, and 2.7%, over existing HCDLP, ACISE, and AHISM techniques.

Corresponding Author:

Varghese Jensy Babu
Department of Computer Science and Engineering, Vidya Academy of Science and Technology
Thalakkottukara, Thrissur, Kerala, India
Email: jensy1019@gmail.com

1. INTRODUCTION

Named data networking (NDN) represents a groundbreaking shift in Internet architecture, moving away from traditional connection-based models towards a content-centric paradigm [1]. In NDN, data is identified and retrieved based on unique names, eliminating the reliance on specific IP addresses [2], [3]. This novel approach enables consumers to access content directly by name, streamlining data retrieval and fostering an environment conducive to efficient in-network caching [4], [5]. With a hierarchical naming structure similar to URLs, NDN enhances network usability by offering a more intuitive and flexible method of accessing and distributing data, ultimately enhancing overall network efficiency and scalability [6], [7].

NDN also leverages in-network caching, where routers along the data path can cache and serve frequently requested data [8], [9]. This caching mechanism reduces the need to fetch data repeatedly from sources, improving network efficiency and reducing latency [10]. Moreover, caching enables efficient content distribution, particularly for popular or widely accessed data [11], [12]. Another notable feature of NDN is its dynamic adaptability to network conditions [13]. Routers maintain state information about pending data requests (interest packets) and use this information to forward data back along the reverse path of the interest [14]. This dynamic forwarding mechanism ensures reliable data delivery even in dynamic or changing network topologies [15], [16].

NDN offers a multitude of advantages over traditional IP networking. One key benefit is its inherent security model, where data integrity is ensured through digital signatures [17]. This built-in security mechanism mitigates risks associated with unauthorized data tampering or access [18]. Recent research on NDN security includes various approaches. Hussain *et al.,* [19] developed a certificateless signature scheme for cache pollution attacks (CPA), showing reduced communication overhead. Al-Share *et al.*, [20] introduced a CIFA detection algorithm with rapid detection times and better pending interest table (PIT) utilization. Li *et al*., [21] proposed an access control system (ACISE) with reduced response delays during attacks. Anisetti *et al*. [22] presented a certification approach for ongoing security verification. Li and Ma [23] introduced a blockchain-based security mechanism (AHISM) with reduced response delays. Hidouri *et al*., [24] developed Q-ICAN, achieving 95.09% accuracy in CPA detection and reducing average response delay (ARD) by 18%. Babu and Jose [25] proposed DFORS-CSA, enhancing precision and reducing false positives.

Several challenges in NDN include vulnerability to CPA, which involve malicious entities flooding the network with harmful data requests. This results in increased cache misses, data retrieval delays, and resource exhaustion. These challenges significantly compromise the network’s efficiency and security making it difficult to maintain optimal performance in NDN environments. However, existing methods suffer from several drawbacks, such as low accuracy, potentially impacting system performance, scalability, and efficiency. To overcome these challenges a Jellyfish optimized deep learning framework for CPA detection in NDN environment technique has been proposed to improve security and detect CPA attacks accurately. The major contribution of the work has been followed by,

- Initially data requests originate from regular users, legitimate consumers, and potentially malicious sources, the network utilizes interest packets to request specific content, and data packets to deliver the requested content in response.
- After the interest packet is received, nodes check their content store (CS) and PIT to serve content locally or forward requests based on the forwarding information base (FIB).
- Detection of CPAs is enhanced using a jellyfish-optimized dual-gate attention-based long short-term memory (LSTM) network, trained to analyze request sequences and predict malicious behaviors
- The efficacy of the proposed methodology has been evaluated based on recall, F1-score, accuracy, precision, average delay time and mean square error (MSE).

The remaining components of the suggested approach are described below. Section 2 explains the DSODAL technique. Section 3 explores the results as well as the discussion. Section 4 explains the conclusion.

2. JELLYFISH OPTIMIZED DEEP LEARNING FRAMEWORK FOR CACHE POLLUTION ATTACK DETECTION IN NDN ENVIRONMENT

In NDN, where data requests originate from regular users, legitimate consumers, and potentially malicious sources, the network utilizes interest packets to request specific content, and data packets to deliver the requested content in response. If data requests from malicious sources are accepted, it can lead to potential CPA attacks and vulnerabilities within the system. To detect the CPA a novel jellyfish optimized DL framework for CPA detection in NDN environment technique has been proposed to improve the security and CPA accurately. When an interest packet is received, nodes check their PIT and CS to serve content locally or forward requests based on the FIB.

Data packets are then routed back to requesting interfaces using PIT information, potentially caching the data in the CS. For detecting CPAs, a DA-LSTM network is employed. Before training the DA-LSTM model, jellyfish optimization is used for hyperparameter tuning, optimizing parameters relevant to the LSTM model architecture and training process. The DA-LSTM network is trained to analyze request sequences and detect patterns indicative of CPAs, where malicious consumers flood the cache with unwanted or harmful content. The dual-gate attention mechanism within the LSTM helps prioritize important features within the request sequences, enhancing the network’s ability to accurately predict cache pollution behaviors. Figure 1 shows the proposed DSODAL.

2.1. DA-LSTM structure

An attention-based encoding-decoding layer and an input layer make up the two primary parts of the DA-LSTM model presented in this work. Figure 2 illustrates the architecture of the DA-LSTM.

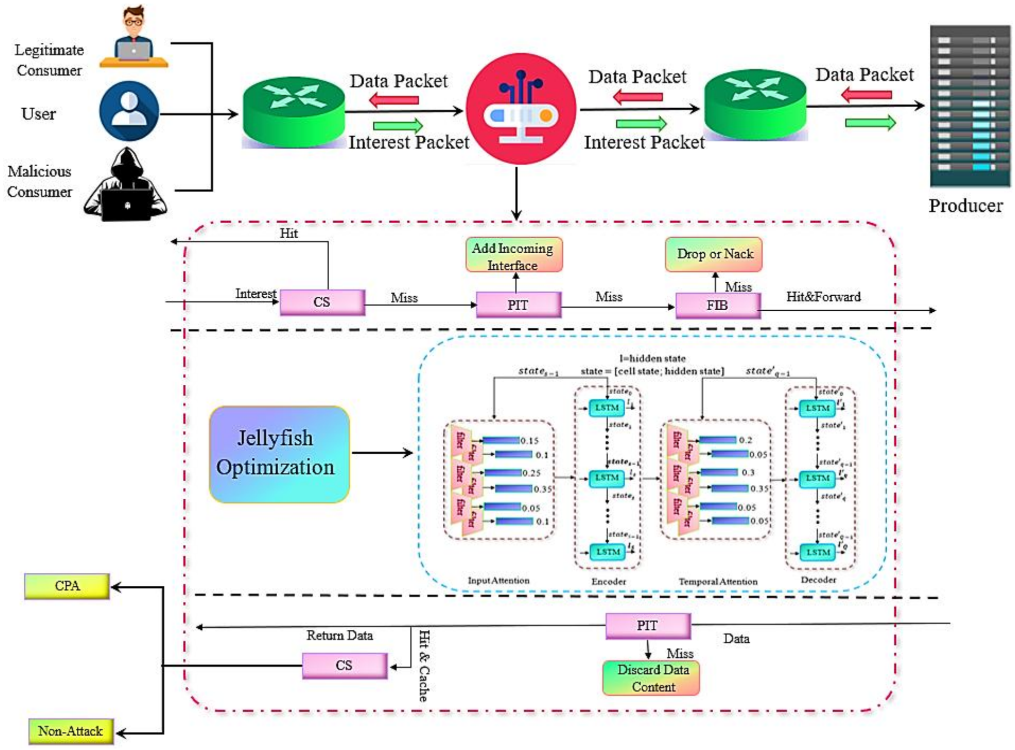

Figure 1. Jellyfish optimized DL framework for CPA detection in NDN environment framework

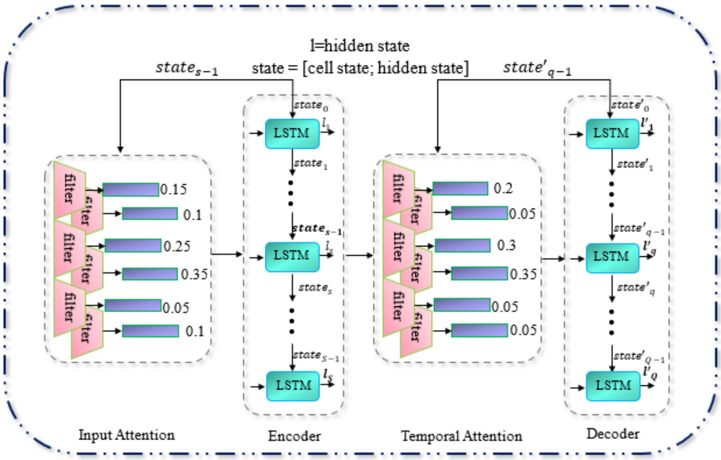

Figure 2. Structure of a DA-LSTM

2.1.1. Input attention layer

A deterministic attention system that uses the state vector state $_{(s-1)}$ from the encoder LSTM unit can be used to generate an input attention layer at time step d for the input sequence $u^d = (u^1, u^2, \cdots, u^{15}) \in H^S$.

$$e_s^d = f(\text{ state }_{(s-1)}, u^d = y_e^S \tanh\left(F_e[l_{s-1}; g_{s-1}] + X_e u^d + a_e\right) \tag{1}$$

The weight vector in the attention mechanism is represented by (1). LSTM unit, and the hidden state $h_{(s-1)}$. Bias terms are indicated by the symbol a_e. The matrices $y_e^S \in H^{(Sp)}, F_e \in H^S, X_e \in H^{(SS)}$ are weight matrices subject to training. This normalized computation is represented as shown in (2):

$$\delta_s^d = \mathrm{SoftMax}\left(e_s^d\right) = \frac{\exp\,(e_s^d)}{\sum_{d=1}^{15} \exp\,(e_s^d)} \tag{2}$$

The normalized attention weight, denoted by the number δ_s^d, evaluates the d -th input feature's significance at time s. By running e_s^d through a SoftMax function, one may make sure that the total of all attention weights is 1.

2.1.2. Encoding-decoding layer

In DL, the encoder-decoder is a widely used model. This enables the framework to concentrate on pertinent segments of an input. When forecasting time series, the input $\tilde{u}_s = \tilde{u}_1, \tilde{u}_2, \tilde{u}_3, \cdots, \tilde{u}_S$ undergoes encoding to establish the mapping from $\tilde{u}_s$ to l_s at time step s using the input sequence $\tilde{u}_s$ is detailed in (3).

$$l_s = \mathrm{LSTM}(l_{s-1}, \tilde{u}_s) \tag{3}$$

The LSTM unit processes the input $\tilde{u}_s$ from the Input-attention layer s, producing the hidden state l_s. To be more precise, the decoder state $\mathrm{state'}_{n-1}$ and the hidden state l_s of the encoder, LSTM unit are used to calculate the updated attention weight for each hidden encoder at a time s is evaluated as per (4).

$$r_q^s = z(\mathrm{state'}_{n-1}, l_s) = y_r^{\mu} \tanh\,(F_r\left[l'_{q-1}; g'_{q-1}\right] + X_r l_s + a_r) \tag{4}$$

The decoder of the conventional encoder-decoder framework directly uses the input $g_q$in the context of energy forecasting. In this case, the hidden state in the decoder is shown by l'_{q-1} and the predicted value s-1 is represented by v_{q-1}. This mapping process is detailed by (5).

$$l'_s = \mathrm{LSTM}\left(l'_{q-1},\right) g'_q \tag{5}$$

The resulting transformed input g'_q is then fed into the decoder to generate the output series l'_s using an LSTM architecture, which incorporates three distinct gates to compute the mapping from the input g'_q to the output l'_s.

2.2. Hyperparameter tuning: JSO algorithm

The jellyfish search optimization (JSO) algorithm's hyperparameters are adjusted to enhance the classification outcomes. The JSO algorithm is a metaheuristic method inspired by jellyfish behavior when they look for food. The migration of the jellyfish is notable for two reasons: it drifts with the currents in the ocean, and its particles within the swarm cause the creation of a bloom. The first step in the JSO implementation method is to randomly distribute the solution throughout a region. The best option was chosen as a plentiful supply of food, and the answer was then organized based on fitness values. Next, each jellyfish movement is updated either toward the ocean's flow or toward the swarming's progress, based on the time-control component. To control the current direction and drift (bring them close) jellyfish with the optimal outer position, the vector-averaging method is utilized (6).

$$\overrightarrow{\mathrm{dr\imath ft}} = \frac{1}{P}\sum \overrightarrow{\mathrm{dr\imath ft}}_d = \frac{1}{P}\sum(m^* - r_s m_d) = m^* - r_s \frac{m_d}{P} = m^* - r_s \varphi' \tag{6}$$

The jellyfish in the population now occupying the optimal location is denoted by m^* in (6). The total number of jellyfish in the population is indicated by the variable N. The JSO mean location is shown by φ' , whereas the parameter r_s represents the attraction strength or parameter.

$$\mathrm{DP} = r_s \times \varphi' \tag{7}$$

The difference factor (DF) in this case represents the difference between a jellyfish's present location and the swarm's average location. Each dimension of spatial allocation follows a specific pattern, which dictates the likelihood of placing each jellyfish. The position of each jellyfish can vary within a range of $\pm\rho\omega$. Here, ρ represents the cognitive distribution coefficient, and ω denotes the standard deviation used for discretized distribution.

$$\mathrm{DF} = \rho \times \mathrm{rand}^{\delta}(0,1) \times \omega \tag{8}$$

$$\omega = \mathrm{rand}^{\delta}(0,1) \times \varphi' \tag{9}$$

As a result, (10) can be used to define drift mathematically.

$$\overrightarrow{\mathrm{drıft}} = \mathrm{m}^{*} - \rho \times \mathrm{rand}^{\delta}(0,1) \times \varphi' \tag{10}$$

Therefore (11) is used to define each jellyfish's new position.

$$\mathrm{m_d}(\mathrm{w}+1) = \mathrm{M_D} + \mathrm{rand}^{\delta}(0,1) \times \overrightarrow{\mathrm{drıft}} \tag{11}$$

Where the position of the $\mathrm{d^{th}}$ jellyfish at time w is indicated by $\mathrm{m_d}(\mathrm{w})$. Time, w, is comparable to the iteration in this work. Jellyfish in a swarm can move in two different ways: passively (Type A) or aggressively (Type B).

$$\mathrm{m_d}(\mathrm{w}+1) = \mathrm{m_d}(\mathrm{w}) + \alpha \times \mathrm{rand}(0,1) \times (\mathrm{UB} - \mathrm{LB}) \tag{12}$$

In (12), the parameter α signifies the motion constant, which determines the distance of movement around each jellyfish's position. Typically, α is set to a fixed value, such as 0.1. The search region's upper and lower boundaries are denoted as UBandLB, representing the limits within which the jellyfish can explore and navigate.

In Type B movement, jellyfish (z) are picked at random and then used to create a vector from another jellyfish (z) to the jellyfish of interest (d) to determine the paths. The food supply that is currently available at the jellyfish's current location (z) can be used to specify the migration path. Proceed later if there is more food at site z than at place d. However, the jellyfish will move away from it if there is less food at position z than at site d. If there is more food at site z than at place d, proceed later.

Type B movement involves selecting a jellyfish (denoted as z) at random and then drawing a vector from this randomly chosen jellyfish z to another jellyfish of interest d to determine their pathways. The migration path is determined based on the food resource availability at the present site of jellyfish z. In the event that there is more food at location z than at site z, then the jellyfish d will move towards z; however, if there is less food available at location z compared to location d, then the jellyfish $\mathrm{d^{th}}$ will move away from z.

$$\overrightarrow{\mathrm{dırectıon}} = \begin{cases} \mathrm{m_z}(\mathrm{w}) - \mathrm{m_d}(\mathrm{w}); \mathrm{ffm_d}(\mathrm{w}) \geq \mathrm{ffm_z}(\mathrm{w}) \\ \\ \mathrm{m_d}(\mathrm{w}) - \mathrm{m_z}(\mathrm{w}); \mathrm{ffm_d}(\mathrm{w}) \geq \mathrm{ffm_z}(\mathrm{w}) \end{cases} \tag{13}$$

Where ff shows the fitness function.

$$\overrightarrow{\mathrm{step}} = \mathrm{rand}(0,1) \times \overrightarrow{\mathrm{dırectıon}} \tag{14}$$

$$\overrightarrow{\mathrm{step}} = \mathrm{m_d}(\mathrm{w}+1) - \mathrm{m}(\mathrm{w}) \tag{15}$$

$$\mathrm{m_d}(\mathrm{w}+1) = \overrightarrow{\mathrm{step}} + \mathrm{m_d}(\mathrm{w}) \tag{16}$$

A temporal control procedure is used to define the type of motion that jellyfish employ. The time component controls motion, such as the jellyfish (Types A and B) moving toward the water flow and into the swarm.

3. RESULT AND DISCUSSION

The DSODAL method's experimental results are analyzed in this section. The DSODAL approach utilizes the ndnSIM module of network simulator 3 (NS-3). The suggested model's efficacy is contrasted with that of the existing HCDLP [19], ACISE [21], and AHISM [23] techniques regarding accuracy, precision, recall, F1-score, average delay time and MSE.

3.1. Performance metrics

The detection efficiency of the suggested approach is measured using the performance metrics. The performance metrics are F1-score (F1S), recall (RC), precision (PR), accuracy (AC), average delay time, and MSE.

- Accuracy: a fundamental metric for measuring correct sensor measurements. In balanced sensor nodes, where false positives and false negatives are nearly equal, statistical accuracy improves as it is proportional to the entire count of values.

$$\mathrm{AC} = \frac{\mathrm{TruePo+TrueNe}}{\mathrm{FalseNe+TruePo+FalsePo+TrueNe}} \tag{17}$$

- Precision: the precision measures how many correctly predicted positive observations there were out of all the positive observations.

$$\mathrm{PR} = \frac{\mathrm{TruePo}}{\mathrm{TruePo+FalseNe}} \tag{18}$$

- Recall: it is a ratio of positive comments that was accurately predicted based on every real observation made in class.

$$\mathrm{RC} = \frac{\mathrm{TruePo}}{\mathrm{TruePo+FalseNe}} \tag{19}$$

- F1-score: recall and precision are averaged and weighted. This score therefore takes into consideration both false positives and false negatives.

$$\mathrm{F1S} = 2 \times \frac{\mathrm{PR.RC}}{\mathrm{PR+RC}} \tag{20}$$

3.2. Comparison analysis

This section includes the simulation to assess how successful the proposed DSODAL technique. HCDLP [19], ACISE [21], and AHISM [23] techniques are compared with the suggested method. The DSODAL approach is evaluated using accuracy, precision, recall, F1-score, average delay time and MSE.

Figure 3 illustrates the performance of recall, accuracy, precision and F1-score with the proposed and existing techniques. For each classification technique, the accuracy, precision, recall, and F1-score of the overall performance are evaluated using the TruePo, TrueNe, FalsePo, and FalseNe. The proposed approach achieves 99.23% accuracy, over HCDLP [19], ACISE [21], and AHISM [23], which achieved 97.5%, 96.9%, and 96.48% respectively.

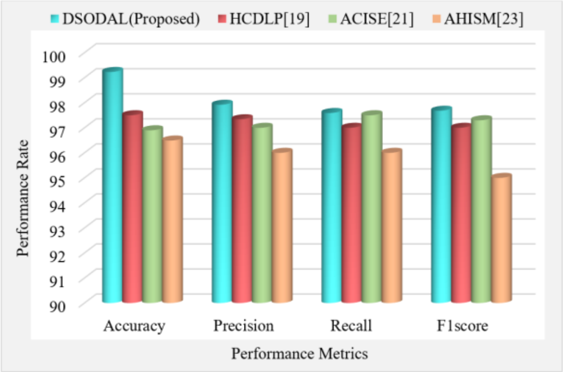

Figure 3. Performance comparison

Figure 4 illustrates a comparison of the average delay time between the proposed DSODAL approach and the existing HCDLP [19], ACISE [21], and AHISM [23] techniques. The proposed scheme achieves a lower delay time than the AHISM [23] scheme when the number of data categories is lower than 150. This advantage diminishes with a larger number of categories (more than 150), resulting in slightly higher delay times for DSODAL. However, this trade-off is acceptable because the DSODAL scheme achieves a significantly higher cache hit rate. This means it finds the requested data more often within the network, reducing the need to fetch it from scratch and leading to overall better performance.

Figure 5 displays a comparison of accuracy among four different methods: HCDLP, ACISE, AHISM, and a proposed method. The proposed method demonstrates a remarkable accuracy of 99.23%, showcasing its strong performance. In contrast, HCDLP achieves 97.5% accuracy, ACISE achieves 96.9%

accuracy, and AHISM achieves 96.48% accuracy. The proposed model outperforms these existing methods by substantial margins, improving accuracy by 1.74% over HCDLP, 2.34% over ACISE, and 2.7% over AHISM. These findings highlight the significant advancement and effectiveness of the Proposed Method in achieving higher accuracy compared to the other evaluated techniques.

Figure 6 illustrates a comparison of error rates between the proposed DSODAL approach and several existing techniques, namely HCDLP [19], ACISE [21], and AHISM [23]. The error rate achieved by the DOLI approach is notably lower at 0.004% compared to the error rates of the existing techniques: HCDLP at 0.013%, ACISE at 0.008%, and AHISM at 0.006%. A lower error rate signifies higher accuracy or better performance in the context of the task being evaluated. Therefore, based on this comparison, the DSODAL approach demonstrates superior performance in minimizing errors compared to the referenced existing techniques.

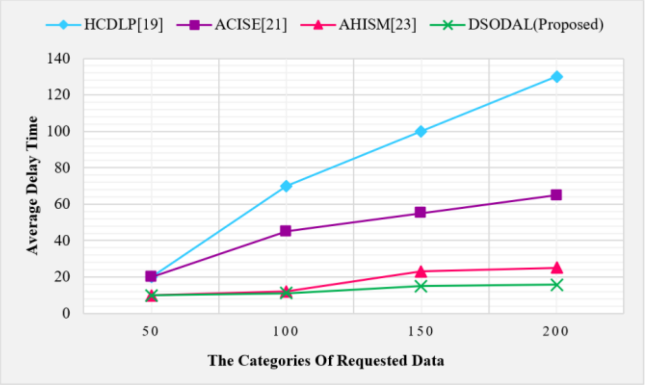

Figure 4. Average delay time comparison

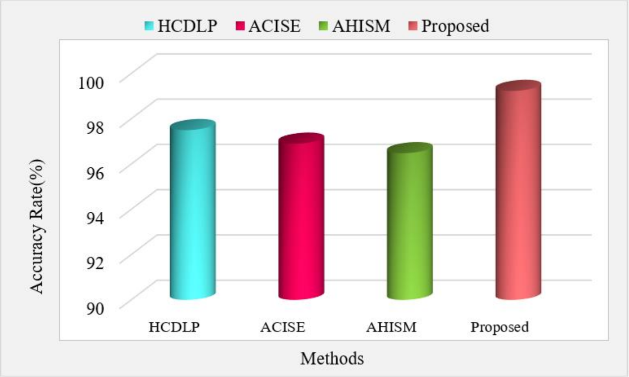

Figure 5. Accuracy-based performance comparison

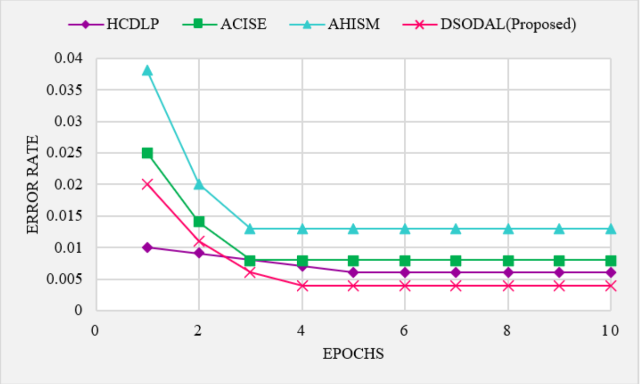

Figure 6. Error comparison

3.3. Discussion

This study investigated the effectiveness of the DSODAL framework for detecting CPA in NDN. While earlier studies have explored various approaches such as HCDLP, ACISE, and AHISM to address *///CPA detection but they do not tackle the challenge of balancing high detection accuracy with reduced delay times and error rates in dynamic NDN environments. The DSODAL framework demonstrates a strong correlation between optimized parameter tuning using the JSO algorithm and improved CPA detection performance. The proposed method achieves an accuracy of 99.23% surpassing existing methods such as HCDLP, ACISE, and AHISM which achieve 97.5%, 96.9%, and 96.48%. Additionally, DSODAL exhibits a significantly lower error rate of 0.004% respectively. This study suggests that higher accuracy in detecting CPA is not associated with increased error rates in smaller data categories. The suggested method benefits from the JSO algorithm for hyperparameter tuning without adversely impacting the detection reliability in dynamic NDN environments. This study explored a comprehensive framework combining the JSO algorithm with a dual-gate attention-based LSTM network for CPA detection. However in-depth studies may be needed to confirm its scalability and real-time performance in highly dynamic NDN environments. The study demonstrates that the DSODAL framework is more resilient to CPA than existing methods such as HCDLP, ACISE, and AHISM. Future studies explore integrating advanced machine learning (ML) and DL classifiers with the DSODAL framework to further enhance efficiency with feasible ways of optimizing delay times for larger data categories and reducing computational overhead. Additionally evaluating the approach using publicly accessible datasets could provide broader validation and applicability. Recent observations suggest that the DSODAL framework significantly improves the detection of CPA in NDN environments. Our findings provide conclusive evidence that this improvement is associated with higher accuracy and lower error rates, not due to elevated computational costs or delays. The DSODAL approach enhances network security and efficiency by accurately identifying malicious behaviors without compromising performance.

4. CONCLUSION

This paper introduces a novel jellyfish optimized deep learning framework for CPA detection in NDN environment to enhance security and accurately detect CPA. In NDN, data requests originate from diverse sources, including both legitimate users and potential malicious entities. Interest packets specify desired content, and data packets deliver requested content in response. Nodes leverage the PIT to track and manage incoming requests, optimizing network performance, while the CS caches frequently accessed data to maximize retrieval efficiency. The proposed technique utilizes the JSO algorithm to optimize a DA-LSTM network, which effectively identifies CPAs within NDN by analyzing request sequences for harmful patterns. By enhancing network security against cache pollution, this approach contributes to increased data delivery resilience and efficiency in NDN environments. The DSODAL method is implemented utilizing the ndnSIM module of network simulator 3 for evaluation. The suggested technique's performance is evaluated using accuracy, precision, recall, F1-score, average delay time, and MSE. The DSODAL model advances the overall accuracy by 1.74%, 2.34%, and 2.7%, over the existing HCDLP, ACISE, and AHISM techniques respectively. In future research, the effectiveness of the CPA detection system could be extended by employing ML and DL classifiers with various publicly accessible datasets.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Varghese Jensy Babu	✓	✓	✓	✓	✓	✓		✓	✓	✓		✓	✓	
Victor Jose Marianthiran	✓		✓	✓	✓		✓		✓	✓	✓		✓	✓

C :	**C**onceptualization	I :	**I**nvestigation	Vi :	**Vi**sualization
M :	**M**ethodology	R :	**R**esources	Su :	**Su**pervision
So :	**So**ftware	D :	**D**ata Curation	P :	**P**roject administration
Va :	**Va**lidation	O :	Writing - **O**riginal Draft	Fu :	**Fu**nding acquisition
Fo :	**Fo**rmal analysis	E :	Writing - Review & **E**diting		

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

INFORMED CONSENT

We certify that we have explained the nature and purpose of this study to the above-named individual, and we have discussed the potential benefits of this study participation. The questions the individual had about this study have been answered, and we will always be available to address future questions.

ETHICAL APPROVAL

Our research guide reviewed and ethically approved this manuscript for publishing in this journal.

DATA AVAILABILITY

Data sharing not applicable to this article as no datasets we regenerated or analyzed during the current study.

REFERENCES

[1] A. Kaci and A. Rachedi, "Named data networking architecture for internet of vehicles in the era of 5G," *Annals of Telecommunications*, vol. 76, pp. 717-729, 2021, doi: 10.1007/s12243-021-00866-8.

[2] A. H. Magsi, L. V. Yovita, A. Ghulam, G. Muhammad, and Z. Ali, “A content poisoning attack detection and prevention system in vehicular named data networking,” *Sustainability*, vol. 15, no. 14, pp.10931, 2023, doi: 10.3390/su151410931.

[3] P. Kar, L. Chen, W. Sheng, C. F. Kwong, and D. Chieng, “Advancing NDN security: efficient identification of cache pollution attacks through rank comparison,” *Internet of Things*, pp.101142, 2024, doi: 10.1016/j.iot.2024.101142.

[4] M.S.M. Shah, Y.B. Leau, M. Anbar, and A.A. Bin-Salem, “Security and integrity attacks in named data networking: a survey,” *IEEE Access*, vol. 11, pp.7984-8004, 2023, doi: 10.1109/access.2023.3238732.

[5] N. Kumar, and S. Srivastava, “IBPC: an approach for mitigation of cache pollution attack in NDN using interface-based popularity,” *Arabian Journal for Science and Engineering*, vol. 49, no. 3, pp. 3241-3251, 2024, doi: 10.1007/s13369-023-07919-1.

[6] A. Anjum, P. Agbaje, A. Mitra, E. Oseghale, E. Nwafor, and H. Olufowobi, “Towards named data networking technology: emerging applications, use cases, and challenges for secure data communication,” *Future Generation Computer Systems*, vol. 151, pp.12-31, 2024, doi: 10.1016/j.future.2023.09.031.

[7] S. Bilgili, A. K. Demir, and S. Alam, “IfNot: an approach towards mitigating interest flooding attacks in named data networking of things,” *Internet of Things*, vol. 25, pp. 101076, 2024, doi: 10.1016/j.iot.2024.101076.

[8] R. Alubady, M. Salman, and A. S. Mohamed, “A review of modern caching strategies in named data network: overview, classification, and research directions,” *Telecommunication Systems*, vol. 84, no. 4, pp. 581-626, 2023, doi: 10.1007/s11235-023-01015-3.

[9] Q. Xia, I. A. Obiri, J. Gao, H. Xia, X. Zhang, K. O. Asamoah, and S. Amofa, “PRIDN: a privacy preserving data sharing on named data networking,” *IEEE Transactions on Information Forensics and Security*, 2023, doi: 10.1109/tifs.2023.3327660.

[10] M. K. Sameer, and M. I. Salman, “Detection and mitigation of cache pollution attack using popularity variation in information centric networking based on SDN,” *Iraqi Journal of Science*, vol. 64, no. 3, pp. 1442-1462, 2023, doi: 10.1109/tifs.2023.3327660.

[11] K. Ding, J. Yang, J. Han, B. Wang, R. Li, and K. Xue, “RPBV: reputation-based probabilistic batch verification scheme for named data networking,” In *2023 IEEE/ACM 31st International Symposium on Quality of Service (IWQoS)*, pp. 01-10, 2023, doi: 10.1109/iwqos57198.2023.10188736.

[12] L. V. Yovita, and N. R. Syambas, “Caching on named data network: a survey and future research,” *International Journal of Electrical & Computer Engineering*, vol. 8, no. 6, pp. 2088-8708, 2018, doi: 10.11591/ijece.v8i6.pp4456-4466.

[13] A. Agiollo, E. Bardhi, M. Conti, N. Dal Fabbro, and R. Lazzeretti, “Anonymous federated learning via named-data networking,” *Future Generation Computer Systems*, vol. 152, pp. 288-303, 2024, doi: 10.1016/j.future.2023.11.009.

[14] A. Rosli, S. Hassan, and M.H. Omar, “Blockchain consensus mechanism in named data networking: enabling trust in Industry 5.0,” In the *Future of Human-Computer Integration*, pp. 39-52, 2024, doi: 10.1201/9781003479727-4.

[15] B. Li, M. Zheng, and M. Ma, “A novel security scheme supported by certificateless digital signature and blockchain in named data networking,” *IET Information Security*, 2024, doi: 10.1049/2024/6616095.

[16] Y. Meng, and A.B. Ahmad, “Performance measurement through caching in named data networking based internet of things,” *IEEE Access*, 2023, doi: 10.1109/access.2023.3290312.

[17] M. Hussaini, S. A. Nor, and A. Ahmad, “Analytical modelling solution of producer mobility support scheme for named data networking,” *International Journal of Electrical & Computer Engineering (IJECE)*, vol. 9, no. 5, pp. 3850-3861, 2019, doi: 10.11591/ijece.v9i5.pp3850-3861.

[18] S. V. Karthik, and J. Selvi, "NDN content poisoning mitigation using bird swarm optimization and trust value," *Intelligent Automation & Soft Computing*, vol. 36, no. 1, 2023, doi: 10.32604/iasc.2023.025404.
[19] S. Hussain, S. S. Ullah, A. Gumaei, M. Al-Rakhami, I. Ahmad, and S. M. Arif, "A novel efficient certificateless signature scheme for the prevention of content poisoning attack in named data networking-based internet of things," *IEEE Access*, vol. 9, pp. 40198-40215, 2021, doi: 10.1109/access.2021.3063490.
[20] R. A. Al-Share, A. S. Shatnawi, and B. Al-Duwairi, "Detecting and mitigating collusive interest flooding attacks in named data networking," *IEEE Access*, vol. 10, pp. 65996-66017, 2022, doi: 10.1109/access.2022.3184304.
[21] B. Li, M. Ma, Y. Zhang, and F. Lai, "Access control supported by information service entity in named data networking," In *2022 5th International Conference on Hot Information-Centric Networking (HotICN)*, pp. 30-35, 2022, doi: 10.1109/hoticn57539.2022.10036170.
[22] M. Anisetti, C. A. Ardagna, F. Berto, and E. Damiani, "A security certification scheme for information-centric networks," *IEEE Transactions on Network and Service Management*, vol. 19, no. 3, pp. 2397-2408, 2022, doi: 10.1109/tnsm.2022.3165144.
[23] B. Li, and M. Ma, "An advanced hierarchical identity-based security mechanism by blockchain in named data networking," *Journal of Network and Systems Management*, vol. 31, no. 1, pp. 13, 2023, doi: 10.1007/s10922-022-09689-x.
[24] A. Hidouri, H. Touati, M. Hadded, N. Hajlaoui, P. Muhlethaler, and S. Bouzefrane, "Q-ICAN: a Q-learning-based cache pollution attack mitigation approach for named data networking," *Computer Networks*, vol. 235, pp. 109998, 2023, doi: 10.1016/j.comnet.2023.109998.
[25] V. J. Babu, and M. V. Jose, "Dynamic forest of random subsets-based one-time signature-based capability enhancing security architecture for named data networking," *International Journal of Information Technology*, vol. 15, no. 2, pp.773-788, 2023, doi: 10.1007/s41870-021-00786-9.

BIOGRAPHIES OF AUTHORS

Varghese Jensy Babu received B.E. (computer science and engineering) and M.E. (computer science and engineering) from Mumbai University. Currently perusing Ph.D. in computer science and engineering from Nurul Islam University, Kumarakoil. At present working as an Assistant Professor in Vidya Academy of Science and Technology, Thrissur. She is having 15 years of teaching experience. Her area of interest is named data networking. She can be contacted at email: jensy1019@gmail.com.

Victor Jose Marianthiran is working as Professor in Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Avadi, Tamil Nadu, India. He has obtained his bachelor degree B.E. in computer science and engineering from Manonmaniam Sundaranar University, Tamil Nadu, India and master degree M.E. in computer science and engineering from Madurai Kamaraj University, Tamil Nadu, India. He has received Ph.D. degree in computer science and engineering from Anna University, India during the year 2016. He has 26 years of experience in teaching and 14 years of experience in research. He has published 36 papers in international journals, 6 papers in national journals, 8 papers in international conferences and 30 papers in national conferences. He is the recognized supervisor in Anna University, Tamil Nadu, India. His research interests include cloud computing, network security, multimedia wireless communications and medical image processing. He is the life member of The Indian Society for Technical Education. He can be contacted at email: mvictorjose@gmail.com.