Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 9, No. 5, October, 2025

Performance Evaluation of Significant Feature for Interest Flooding Attack Detection on Named Data Networking

Jupriyadi [1, 2*], Nana R. Syambas [1], Eueung Mulyana [1]

[1] *School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung 40132, Indonesia.*

[2] *Faculty of Engineering and Computer Sciences, Universitas Teknokrat Indonesia, Bandarlampung 35142, Indonesia.*

Abstract

One of the internet architectures of the future that has advantages over the current system is Named Data Networking (NDN). However, Denial of Service (DoS) attacks, such as interest flooding attacks (IFA), can still disrupt the network. Detecting IFA attacks is crucial for preventing further damage. Several approaches to detection systems have been proposed, including a classification approach to detecting attacks with multiple detection parameters or features. However, the many detection system features that can be extracted from the network result in longer computation times for the classification algorithms. This research focuses on enhancing the detection of IFA by evaluating the features of the detection system and identifying significant features to improve detection accuracy and reduce computation time. We employed various feature selection algorithms, including information gain, wrapper naive Bayes, gain ratio, and correlation-based feature selection (CFS). The selected features are tested to detect attacks using several classification algorithms, including naive Bayes, random forest, J48, and Bayesian network. Our proposed method found only three essential features for detecting IFA from 18 features available, resulting in better detection accuracy and increasing by 47.8% the time to build the model. This study enhances NDN security while reducing computational cost, making real-time attack detection more feasible.

Keywords:

Named Data Networking;
NDN;
Feature Selection;
Classification;
Interest Flooding Attack.

Article History:

Received:	04	March	2025
Revised:	17	August	2025
Accepted:	22	August	2025
Published:	01	October	2025

1- Introduction

A future internet architecture called named data networking (NDN) uses a different paradigm than existing networks [1, 2]. The NDN network no longer focuses on where data is but on what users need content. To get content, users can make requests without knowing where the requested content is Afanasyev et al. [3], and Saxena et al. [4]. There were two (two) different types of packets used in NDN communication: interest packets and data packets [5]. Consumers send interest packets to request specific types of content. Producers or routers send out data packets with the requested data content. The NDN communication model is inseparable from security threats that can disrupt network performance [6]. Denial of service (DoS) attacks are still possible on NDN by sending many interest packets. One of the DoS attacks on NDN is an interest flooding attack, which can disrupt the network. It requires an excellent mechanism to detect attacks to minimize the negative impacts that arise. Several systems detection approaches have been proposed, including the classification approach. Detection systems using a classification approach can detect IFA attacks using more than two parameters [7]. However, the more parameters used, the more computation time will be burdensome. The essential parameters (features in the dataset) must be selected considering the many parameters extracted from the network. Unimportant parameters can increase the processing time and accuracy of the detection system. This study aims to improve the efficiency of detecting Interest Flooding Attacks (IFAs) by improving feature selection. The main contributions of this study are:

* **CONTACT**: jupriyadi@teknokrat.ac.id

DOI: http://dx.doi.org/10.28991/ESJ-2025-09-05-07

- Improvement of Feature Selection: We review and compare alternatives to feature selection techniques with the primary focus on Correlation-Based Feature Selection (CFS). CFS allows us to quickly determine the most relevant parameters (features) to select for detection. Regularly compared to wrapper and ranking techniques, CFS balances accuracy and computational expense.
- Development of an Optimized IFA Detection Model: The algorithm's chosen parameters are then used to develop a classification-based detection model that minimizes the resources required for scanning while accurately detecting attacks.
- Validation with Machine Learning Algorithms: The selected parameters are then checked for effectiveness across several classifiers, including Naïve Bayes, Random Forest, J48, and Bayesian Networks, to discover the best fit for real-time IFA detection.

This research enhances feature selection and classification approaches to improve computational efficiency and the detection of IFAs. The findings advance NDN security to defend against scanning Interest Flooding Attacks.

1-1- Security Attack in NDN

Some attacks that may occur on NDN networks are interest flooding attacks, cache privacy attacks, cache poisoning attacks, and cache pollution attacks [7].

1-2- Interest Flooding Attack (IFA)

IFA is a kind of NDN denial of service attack, or DoS attack, executed by sending many interest packets. This attack focused on making the PIT table full-fixed and disrupting searching and matching interest packets so legitimate requests become unserved. An attacker launches this type of attack by sending many interest packets continuously. The goal is to make the network service unavailable. IFA attacks are desired at the Pending Interest Table (PIT) to fill the PIT capability with the attacker's interest packets so legitimate consumers cannot request the producer. Interest requests can be made in several ways, namely by sending request packets that are not in the service, sending request packets that are in the service, and in a hybrid manner. Attackers can also send interest packets statically and dynamically, meaning that the interests sent are fixed and changing packets to avoid existing detection mechanisms. Figure 1 illustrates the IFA attack model on the NDN network.

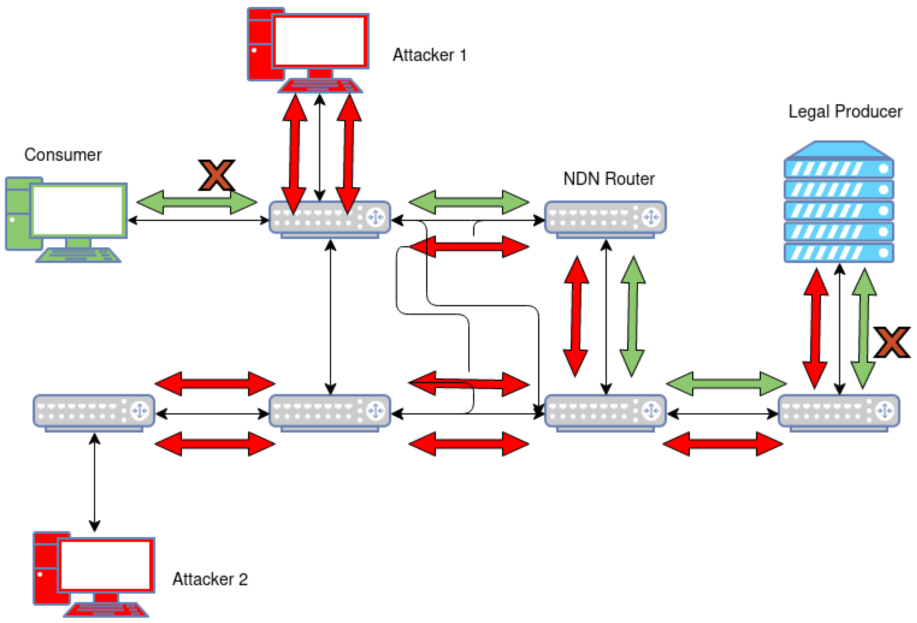

Figure 1. Figure 1. IFA attack model on NDN

1-3- Cache Pollution Attack

This attack aims to slow access to the router's cache by sending unpopular content requests to the router. Each router can store content temporarily and delete it with particular mechanisms. Frequently accessed content will be maintained in the router. This attack keeps content that is rarely accessed inside the router, causing the content that users need to access more slowly [8].

1-4- Cache Privacy Attack

Cache privacy attacks aim to access the sensitive data content of users or groups accessing the network. They exploit the information stored in caches by malicious entities to compromise users' privacy. Caches recreate a crucial role in

NDN by storing frequently requested data closer to consumers, reducing the network load, and improving overall performance. However, this caching mechanism introduces potential privacy concerns.

1-5- Content Poisoning Attack

The attack aims to overload routers by injecting high amounts of bogus or corrupt data, ultimately causing network disruption. A content poisoning attack forces routers to cache and send out corrupted or malicious data instead of legitimate content. To do this, a fake producer that appears to be a legitimate content provider into the network. When users or routers request content, the fake producer sends them corrupted content, which spreads throughout the system. Legitimate users may, even unknowingly, download and later cache the insufficient data, which is a form of service degradation and increased security risk.

Each attack has different effects and characteristics. Table 1 below shows the characteristics of attacks on NDN.

Table 1. **Attack Characteristics on NDN**

Attack Type	Victim	Attacker	Security Goal Effect	Associated NDN Structure
Interest Flooding Attack (IFA)	Consumer, router, and producer	Consumer	Availability	Pending Interest Table (PIT)
Cache Poisoning Attack	Producer and consumer	Producer or router	Availability and integrity	Content Store (CS)
Cache Pollution Attack	Consumer	Consumer	Availability	Content Store (CS)
Cache Privacy Attack	Consumer	Consumer	Confidentiality	Content Store (CS)

1-6- Related Work

An interest flood attack is an attack in which an attacker performs a DDoS attack that harms the NDN network. Attackers conduct DDoS attacks using various techniques to have a high impact. Researchers have suggested various mitigation and detection techniques, including rate-limiting, feature-based classifications, or machine learning-based anomaly detection. These techniques still suffer from issues related to levels of accuracy, real-time detection, and the ability to adapt to new attack patterns. More studies could alleviate issues in battle detection efficiency and improve police readiness to act quickly and accurately against IFA threats in an NDN environment [9].

Overview of the IFA and its countermeasure study. They suggested using a statistics approach to detect the attack. Using a pushback mechanism, the router in this technique alerts other routers to the attack after detecting IFA [10]. The author introduces an additional approach based on pushback. In this method, routers observe the names of incoming Interests and calculate the cumulative entropy of the interface. Once the cumulative entropy exceeds a specific threshold, the router confirms the presence of an attack [11]. Subsequently, the router transmits counterfeit data to the origin of the negative prefix. This approach allows false detection due to incorrect threshold determination and only sees attacks based on incoming interest prefixes. The proposed Reconstruction Forest-based Detection Method (RFDM) tackles the problem of Interest Flooding Attacks (IFAs) in Named Data Networking (NDN), which is an acute security issue that can impair network performance by flooding routers with illegitimate interest packets [12]. Existing detection mechanisms are limited in performance under traffic variations and usually produce detectable false positives. In contrast, RFDM uses RecForest to calculate reconstruction errors, allowing for rapid classification of legitimate versus malicious interest packets. As a reactive approach to stopping malicious content from forwarding, RFDM can effectively reduce IFAs while preserving the network's stability. Simulation results show that RFDM outperforms previous detection techniques in speed, accuracy, and resilience to network variations. The overall approach can reduce excessive Pending Interest Table (PIT) caused by IFAs while increasing the likelihood of identifying a true positive.

Anytime a border router receives this counterfeit data, it restricts the rate of source interfaces. However, it is worth noting that this detection process may exhaust resources and, in specific scenarios, valid interfaces could punished by rate limitation. Another disadvantage is that attackers can exploit fake data to overwhelm the network. Disabling PIT exhaustion (DPE) suggested a countermeasure for IFA. Each router in DPE keeps an m-list, which is utilized as a detection parameter and keeps track of how many expired interest packets there are in each namespace [13]. When this importance transcends a specified threshold, the interest packets with harmful namespace lag are usually transmitted rather than kept in the PIT. As a result, routers are not affected by malicious interest packets because they do not store their entries in the PITs. This system does not prevent IFA and mitigates its impact—a two-stage detection technique conceived. The first phase is wild detection, which uses the interest packet satisfaction rate as an index. If ISR reaches a particular threshold, it determines the attacked condition, and the second level detection will initiated. The second phase involves detecting the extinction of PIT entries by matching the most extended prefix of the extinct entries. When the prefix's expiration ratio exceeds a specific level, determine that the modified prefix has been attacked [14]. This method allows false detection because it only looks at the interest prefix and expiration time. When the interest expires, the router from the PIT will drop it.

A statistical approach using Gini impurity suggests addressing the issue. The router determines the set's disparity through Gini impurity by maintaining a record of received Interests' names and calculating their probabilities [15]. It compares it to a predetermined threshold to assess the occurrence of an IFA. Furthermore, the router employs the Gini impurity to approximate the essential set with an earlier stored set to identify the harmful prefix. It is important to note that this proposed method carries the risk of false-positive detections. Similarly, a different entropy-based approach was introduced in reference [16]. These two methods also make it difficult to determine the proper threshold because the network conditions and topology are diverse and complex. The chi-square-based anomaly detection works by observing regular NDN traffic and comparing that to real-time NDN traffic (while monitoring whether deviations are statistically significant) and reporting those deviations as an indicator of potential attacks [17]. A large number of experiments (with varying parameters) supports findings provide confidence that the items observed as a result of the chi-square-based anomaly detection system can lead to high accuracy detection rates across different forms of attack active intrusion detection system (IDS) units and ultimately improving NDN's resilience in the face of induced IFAs. However, while the proposed method is effective, it has certain limitations. The extent and future security techniques added to deterred enhance overall NDN and additional security applications continuously.

Unlike earlier solutions, an AI-based mechanism utilizes Radial Basis Function or RBF neural networks to alleviate IFA and cache poisoning [18]. It takes as inputs a set of statistics such as the timed-out Interests and the numeral of satisfied. When the sensor or detector module detects opposing traffic, the router delivers an alarm notification to source interfaces. The new lowered rate, the generating timestamp, and the decrease period are all included in the alert message. However, rate-limiting can hurt legitimate users. The detection technique is yet another using a machine learning-based approach [19, 20]. In this explanation, a router continuously organizes and monitors the satisfaction ratio, entropy of interest names, and PIT utilization of interfaces. These metrics are inputs for the classifier's support vector machine (SVM). When the detector identifies an abnormality, the router generates a report indicating the occurrence of an IFA. The router operates the Jensen-Shannon separation to extract the opposing prefixes. The fraudulent prefixes report to downstream routers. This method has an impact on valid Interest packets sent to prohibited prefixes. Like the prior solution, the detecting method may use significant resources. Developing a comprehensive attack dataset and implementing a unified detection and classification strategy-using machine learning [21]. The proposed approach demonstrates its effectiveness in identifying threats. These disadvantages highlight the need for continuous improvements and lightweight solutions to enhance security in IoT-NDN environments.

To mitigate IFA, a strategy involving producers' and routers' collaboration was introduced [22]. This strategy addresses conventional distributed IFA and spread IFA where attackers imitate regular transmission rates. When requests overwhelm a producer, it sends a signed interest to routers, asking them to limit the request rate for the affected prefix explicitly. Upon receiving such an Interest, the router halts forwarding it and notifies other routers accordingly. A border router identifies a customer as potentially harmful if they persistently send submissions to the infected prefix, exhibit a low satisfaction percentage, and experience many timed-out interests. While this suggested technique can effectively mitigate various types of IFA, unlike prior solutions, the approach outlined primarily targets collusive IFA [23]. Routers utilize defined time intervals to monitor interface performance and PIT (Pending Interest Table) utilization to detect opposing traffic. If these metrics surpass their respective points, the router cleanses the most aging entries from the PIT. However, it is essential to note that this technique risks inadvertently deleting legitimate requests.

Different techniques offered an Active Queue Management (AQM)-based solution to mitigate IFA [24]. The router unsystematically selects an uncertain interest and compares its name with the obtained Interest for each welcomed Interest. Then, it approximates their original interfaces and determines the source interface's satisfied ratio. If it falls below a certain point, the router cancels both Interests. Otherwise, it probabilistically drops the accepted Interest. Regardless, attackers can time their attacks to suffice the target's PIT, causing the router to drop incoming interests. The writers altered its technique to incorporate securing [25]. When a boundary router notices an interface's PIT occupancy and satisfied ratio are over criteria, it secures it. The paper presents the Hybrid Proactive Reactive Defense Scheme (HPR-DS), which comprises two key components: Proactive Resource Management (PRM) and Reactive User Behavior Analysis (UBA) [26]. The PRM component employs time-series analysis and a sliding window-based resource allocation approach to manage content-specific resources to restrict excessive access based on the Interest Satisfaction Ratio (ISR) and Round Trip Time (RTT). The UBA component utilizes multidimensional clustering to identify anomalous traffic patterns and report suspicious users at the network's edge. In the simulation, the results have shown that the HPR-DS works to maintain user transmission quality, improves attack detection accuracy, and adapts to user behaviours that may change over time. The current approach assumes that attackers will focus on content that has similar prefixes.

On the other hand, the system still punishes legitimate traffic, and the penalty grows as it approaches the centre network. Each router chooses random prefixes from a registered list of computed metrics for individual prefixes [27]. Then, it generates a set of binary tree searches using the minimum and maximum matters of the nominated prefixes. It then computes the intermediate traverse path of the selected prefixes and utilizes it to calculate an anomaly score for individual prefixes. Finally, it sends the malicious prefixes to downstream routers. Legitimate Interest is penalized by namespace-based rate-limiting. Furthermore, the detection procedure might require significant resources, particularly on core routers. This approach can take some duration to detect threats. This research proposes a new two-phase model for detecting interest flooding attacks in NDN [28]. The first phase involves using key features (e.g., a node behaviour, distribution, pattern, frequency, and runtime) extracted and processed using a Deep Convolutional Neural Network (CNN) algorithm to detect malicious activity such as interest flooding for existing and non-existing data, hijacking interest packets, and signing data with invalid keys. The second phase mirrors the feature extraction process but focuses on identifying interest flooding attacks using a Fuzzy Decision Tree (FDT). The study improves detection accuracy by utilizing a newly proposed Decision Rider Optimization Algorithm (DO-ROA) to optimize the CNN and FDT classifiers, creating an improved version of the first optimization algorithm (Rider Optimization Algorithm - ROA). The performance of both classifiers is evaluated to determine the method efficacy using comparative evaluations via Type I and Type II performance metrics. The results indicate that the DO-ROA optimizations improve accuracy compared to the traditional models, demonstrating an enhanced mechanism for enabling NDN security. This approach didn't implement feature selection to improve speed computation. Based on the analysis carried out on the previously proposed detection system, a summary and comparison were obtained, which can be seen in Table 2.

Table 2. Related work summary

Reference	Detection approach	Feature selection	Independent module	Attack Type	Location development
Kumar et al. [7]	Classification approach	No	Yes	IFA	All routing node
Gasti et al. [10]	Push back mechanism	-	No	IFA	Specific router node
Xin et al. [11]	Relative entropy theory	-	No	IFA	Specific router node
Xing et al. [12]	Reconstruction Forest-based Detection Method (RFDM)	-	No	IFA	Specific router node
Wang et al. [13]	PIT exhaustion (DPE)	-	No	IFA	Specific router node
Zhi et al. [19]	Support vector machine (SVM)	No	Yes	IFA	Specific router node
Afanasyev et al. [20]	Classification approach	No	Yes	IFA	All routing node
Mandapati et al. [21]	Hidden Markov Model (HMM)	No	Yes	IFA	All routing node
Xing et al. [27]	Binary tree and min max algorithm	-	Yes	IFA	Specific router node
Joseph [28]	Deep Convolutional Neural Network (CNN)	No	Yes	IFA	Specific routing node
Kumar et al. [29]	Classification approach with J48 and RBF classifier	Ranking based	Yes	IFA	All routing node
This paper	Classification approach + NBC & Random Forest + feature selection module	Filter based	Yes	IFA	All routing node

This research proposes a detection system using a classification approach by applying correlation-based feature selection (CFS). Correlation-based feature selection aims to find the right features to use as a reference for the detection system and reduce the amount of data to be processed, thereby increasing data processing time and speeding up the detection system. The selected features will be used as a reference for the detection system, and their performance will be compared with previous research.

2- Research Methodology

2-1- Detection Approach

The detection system utilizes a machine learning approach using the available datasets—the classification algorithm to classify traffic into attacks or not. The dataset must be pre-processed by selecting significant features using an attribute selection algorithm to input the classification algorithm. There are 4 (four) significant attribute selection algorithms used, namely gain ratio, information gain, wrapper naive Bayes, and CFS. Next, the dataset with selected features will tested using 4 (four) classification algorithms: J48, naive Bayes, random forest, and Bayesian network. Figure 2 shows the flowchart of the detection approach.

- ndnSIM IFA Simulation: IFA simulation using ndnSIM is at the forefront of the process. It stimulates the ndnSIM IFA environment to generate the data that will be used for the detection module.
- Feature Extraction: After the simulation, relevant features from the generated data are extracted. These features will then be the key indicators to identify the events you want to detect.
- Data Preprocessing: The extracted features are then preprocessed to prepare them for the machine learning model. Data Preprocessing involves cleaning the data, normalizing it, and potentially transforming it into a more suitable format.

- Feature selection: As the name suggests, Feature selection selects important features from preprocessed data for further utilization. It reduces the noise and enhances the model's performance by relying only on the most relevant information.
- Training and testing set: The selected features are then divided into a training set and a test set. The training set develops or trains the detection approach based on the machine learning model. In contrast, the testing set assesses the model's performance or accuracy on unseen data.
- Classification-based detection module: Classification-based is the essence of the process. One then trains a classification model (e.g., support vector machine, random forest, neural network) on the training set. Once the model is trained, new data points can be fed to the model to determine if an event of interest has taken place.
- Model evaluation: The qualified model is assessed on the testing set to evaluate its performance. Model evaluation helps ensure the model is accurate and reliable before it goes into actual deployment.

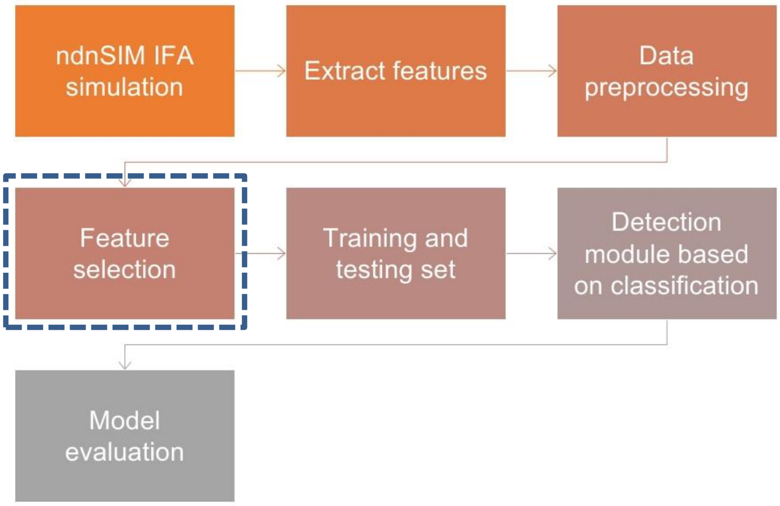

Figure 2. **The flow chart of detecting IFA**

2-2- Experiment Setup

The dataset was created using of ndnSIM on tree topology (Figure 3) [29]. The features used in the IFA detection system are generated via simulation settings, as indicated in Table 3. Experiments were conducted using a laptop with a 4-core i3 processor and 8GB of RAM.

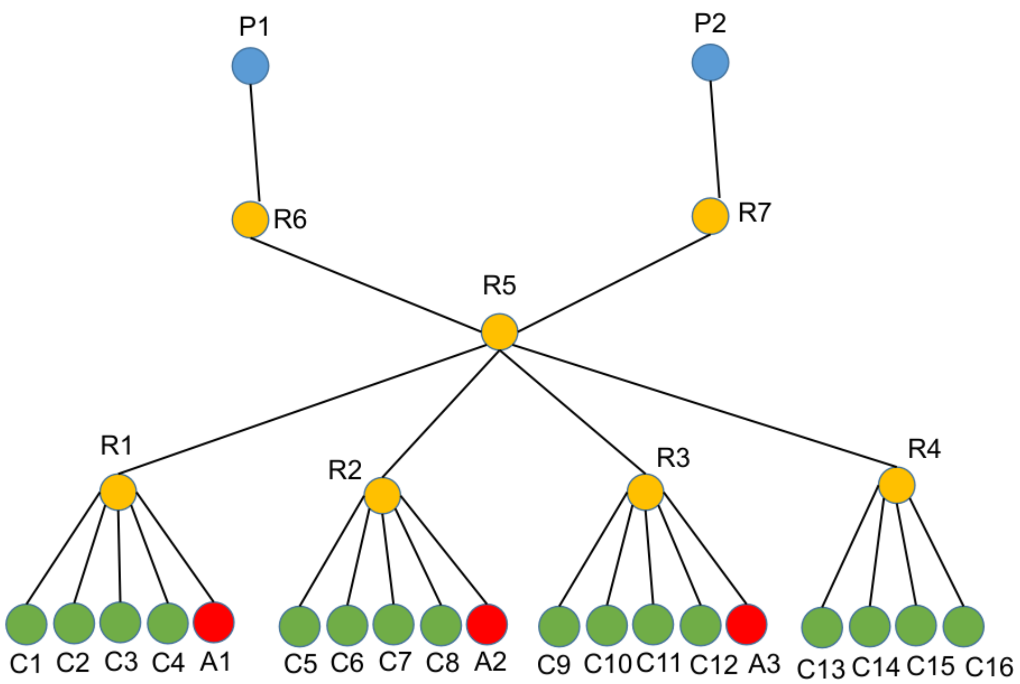

Figure 3. **Attack model on tree topology**

Table 3. Full feature name on dataset for interest flooding attack detection

No.	Feature Name	Description
1	Time	Simulation time when data is taken based on (data taken every second starting from 1-300 seconds)
2	Router	The router used to build the topology (routers 1-7)
3	Interface	Router interface
4	PITSize	PIT size on the router during the simulation
5	PITSizeInt	PIT size on the router during the simulation in integer
6	DropData	The number of packets dropped during the simulation based on a several time
7	OutInterests	The number of outgoing interests on a router
8	DropInterests	The number of drop interest
9	InData	The amount of incoming data on the router
10	InTimedOutInterests	The incoming interest packet expiration time
11	OutTimedOutInterests	The outgoing interest packet expiration time
12	DropNacks	The number of NACK drop
13	InInterests	The numeral of interest packets that enter the router
14	OutData	The amount of data coming out of the router.
15	InNacks	NACK messages that go to the router
16	OutNacks	NACK messages coming out of the router
17	InsatisfiedInterests	Interest packet that are entered and served (per interface)
18	OutSatisfiedInterests	Outgoing and serviced interest packet (per interface)

2-3- Feature Selection Module

Feature selection is an essential technique in machine learning for datasets to reduce and increase accuracy. In this study, features were selected by wrapping methods and filters to get optimal results. Figures 4 and 5 below indicate the selection of the mentor features using the driving and wrapper approaches.

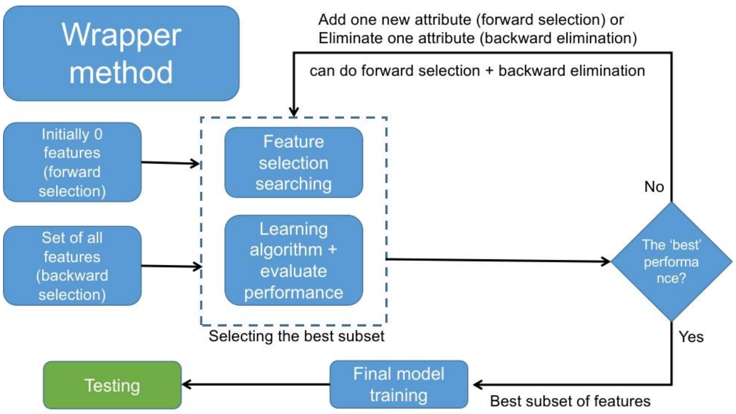

Figure 4. Wrapper method feature selection

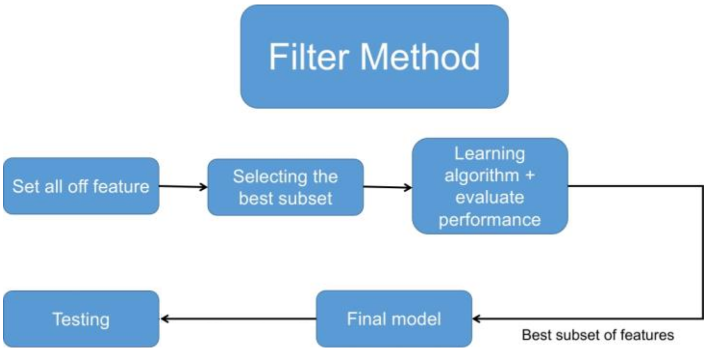

Figure 5. Filter method feature selection

The wrapper method is a feature selection technique that uses a learning algorithm to evaluate different subsets of features. The goal is to find the subset of features that results in the best performance on a given task. Below are the steps of the wrapper method feature selection:

- Start with an initial set of features. This could be an empty set or include all of the available features..
- Train a model on the underlying set of features using a learning algorithm. Evaluate a model. The model's performance could be based on any number of metrics, such as accuracy, precision, or recall.
- Split and select a new subset. It is done either by adding a new feature to the set or removing a feature from the set. Every new combination is chosen, hoping it may have a better effect on improving performance.
- Keep looping through steps 2 - 4 above in order until satisfactory performance.
- If the best subset happens, all final model training will rest on that set.

Search types are forward selection and backward selection. Forward Selection starts with an empty set of features and adds features one at a time until performance plateaus. Backward selection begins with the complete set of features and removes features one at a time until performance plateaus.

The filter method for feature selection is a technique that evaluates the relevance of features independently of any machine learning algorithms. This approach is instrumental in flooding attack detection, as it helps identify the most significant features that contribute to detecting such attacks without the computational overhead of training models. The filter method is a powerful approach for feature selection in flooding attack detection, providing a means to identify relevant features while minimizing the risk of overfitting efficiently. By leveraging statistical measures, filter methods enhance the overall performance of machine learning models in cybersecurity applications.

In this experiment, 4 (four) significant feature selection algorithms were used: gain ratio, information gain, wrapper naive Bayes, and correlation-based feature selection (CFS).

1) Information gain algorithm for feature selection

Significant feature selection using information gain is utilized in data modelling to determine the considerable informative or appropriate features. Information gain calculates how significantly information is accepted by separating the dataset based on characteristics. The information gain algorithm calculates the information gain for each feature in the dataset and sorts it based on the resulting gain value. The feature with the most elevated information gain is believed to be the most informative and is selected and used to further data modelling. This approach helps reduce the dataset sizes and eliminates features that have little contribution in diverging the categories in the dataset. Selecting the right features using the significant feature selection algorithm of information gain features can help improve the efficiency and quality of data modelling [30].

2) Gain ratio feature selection

The Gain Ratio is an alternative to Information Gain for choosing the attribute to split in a decision tree. It addresses the issue of attribute bias with many outcomes. The optimum feature to split on is determined using the Gain Ratio, a metric that considers the information gain and the number of outcomes of a feature. The optimal feature for splitting is determined by comparing the gain ratios of each feature; the highest gain ratio from the feature is chosen [31].

3) Wrapper naive bayes feature selection

Wrapper subset evaluation is used in feature selection algorithms or feature subset evaluation in machine learning. It involves evaluating different subsets of features by training and testing a learning algorithm using each subset wrapped in a performance evaluation metric [32]. Wrapper subset evaluation evaluates different subsets of parameters or features to find the most suitable subset that maximizes the performance of a learning algorithm for a specific task. The wrapper approach investigates all feature subsets and then trains and assesses a classifier for each subgroup to ascertain its quality. Nevertheless, this approach may need to be more computationally efficient. Some wrapper techniques include recursive feature elimination, exhaustive feature selection, forward selection, and reverse selection [33].

4) Correlation based feature selection

Correlation-based feature selection (CFS) is an attribute selection approach that evaluates the relevance and redundancy of features by measuring their correlation with the target variable and among themselves. It aims to select a subset of parameters or features positively correlated with the class variable while maintaining low inter-correlations among themselves [34, 35]. It helps improve the interpretability and efficiency of machine learning models by identifying the most informative features for the task at hand. Put otherwise, a feature is meaningful only if associated with or indicative of the class; otherwise, it is unimportant. An attribute If there are any values of V_i and c such that $p(V_i = v_i)>0$, v_i is relevant otherwise irrelevant.

$$p(C = c|Vi = vi) \neq p(C = c) \quad (1)$$

Features with a strong correlation (predictive power) with the class but a low correlation (not predictive power) with one another make up a good feature subset. The correlation between a combined test of the computed elements and the outside variable can be forecasted from the correlation between separate components in a trial, the external variable, and the inter-correlation between individual pairs of elements. CFS's feature subset evaluation function can be used in Equation 2.

$$r_{zc} = \frac{k\overline{r_{zi}}}{\sqrt{k+k(k-1)\overline{r_{ii}}}} \quad (2)$$

where k is the number of elements, $\overline{r_{zi}}$ is the average of the correlations between the elements and the external variable, $\overline{r_{ii}}$ is the average inter-correlation between elements, and r_{zc} is the correlation between the summed elements and the external variable.

2-4- Detection Algorithm

We use 4 (four) classification algorithms to test datasets with selected features, namely Naive Bayes, J48, random forest, and Bayesian network.

1) Naive Bayes Classifier

The naive Bayes classification algorithm is established on Bayes' theorem and computes the likelihood of data. This algorithm classifies new data with the assumption that its attributes are independent. The algorithm calculates the probability of the data against the existing classes. The data is classified based on calculating the highest probability [36].

2) J48 classifier

The J48 classification algorithm performs classification by building a tree. This algorithm produces a binary tree established on the feature values of the training data. The new data can be classified into existing classes from the formed binary tree [37].

3) Random forest classifier

A Random Forest Classifier is an ensemble learning algorithm operated for classification schemes. It merges considerable decision trees to construct predictions [38]. A piece tree in the Random Forest is built alone, and the final prediction is defined by aggregating the forecasts of all the trees. Random Forest Classifier is comprehended for its power to manage high-dimensional datasets, handle missing values, and provide estimates of feature importance. It is a popular and robust algorithm for classification tasks due to its robustness and ability to mitigate overfitting [39].

4) Bayesian Network

A Bayesian Network, or a Bayesian Belief Network, is a graphical prototype illustrating the probabilistic connections among variables. In a Bayesian Network, variables are depicted as nodes, and the connections between variables are described as directed edges or arrows between the nodes. The nodes in the network symbolize arbitrary variables, and the edges indicate the conditional dependencies between the variables. A Bayesian Network Classifier is a classification algorithm based on Bayesian Networks. It uses the principles of Bayesian probability theory and graphical models to classify instances into different classes or categories [40].

2-5- Evaluation Metrics

The confusion matrix exists as an evaluation benchmark utilized to calculate the interpretation of a classification algorithm. The confusion matrix consists of False positives (FP), true positives (TP), true negatives (FN), and false negatives (FN). TP is attack traffic detected as an attack. FP is attack traffic detected as regular traffic. FN is normal traffic but detected as an attack. TN is regular traffic detected as normal traffic. The Confusion matrix as an implementation evaluation benchmark for the classification algorithm is displayed in Figure 6. Using the confusion matrix, we can define accuracy, precision, TP rate, FP rate, and F-measure.

	Predicted "Attack"	Predicted "Normal"
Actual "Attack"	True Positive (TP)	False Negative (FN)
Actual "Normal"	False Positive (FP)	True Negative (TN)

Figure 6. Evaluation metrics

Accuracy refers to the model's ability to classify the data correctly. It is defined by computing the proportion of valid predictions (both positive and negative) to the complete quantity of data. Accuracy measures the agreement level between the forecasted and actual values. The accuracy value can be obtained by Equation 3.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3}$$

The True positive rate (TP rate) or recall is the percentage of attack traffic classified correctly compared to the total attack traffic. The TP rate value can be obtained using the Equation 4.

$$\text{TP rate} = \frac{\text{True positive (TP)}}{\text{True positive (TP)} + \text{False negative (FN)}} \tag{4}$$

The false positive rate (FP rate) is the ratio of relevant samples incorrectly classified to all samples incorrectly identified as irrelevant. The FP rate value can be obtained using the Equation 5.

$$\text{FP rate} = \frac{\text{False positive (FP)}}{\text{False positive (FP)} + \text{True negative (TN)}} \tag{5}$$

Precision is the ratio of the appropriate sample calculation to the relevant sample count accurately classified. The precision value can be obtained using the Equation 6.

$$\text{Precision} = \frac{\text{True positive (TP)}}{\text{True positive (TP)} + \text{False positive (FP)}} \tag{6}$$

A harmonic mean of recall and precision is the F-measure. The value of the F-measure can be obtained using the Equation 7.

$$\text{Measure} = \frac{2 * \text{precision} * \text{TP rate}}{\text{Precision} + \text{Recall}} \tag{7}$$

3- Results and Discussion

Simulations were done using ndnSIM to produce traffic data collected every second. As evidenced by the simulation results, the PIT usage of each router is greatly influenced by the Interest Flooding Attack (IFA), as illustrated in Figure 7. The attack was carried out throughout the simulation by sending interest packets at 4x, 8x, and 16xy, the original data rate. Attack packet raises also cause the PIT usage to rise (Figure 8). The interest drops further as the number of attacks increases since the high PIT usage does not serve the interest packets (Figure 9). The interest drop rate rises as the attack intensifies, as the vast PIT usage does not serve the interest packets. The increased usage of PIT due to this excess interest traffic adversely affects the overall number of satisfied interest packets. Excessive interest traffic leads to high PIT usage, affecting the overall number of satisfied interest packets (Figure 10). The IFA is seen to adversely affect the interest satisfied ratio, a key performance indicator of the network, which is the declining overall performance of the network (Figure 11). The findings show that IFA harms the NDN network, which requires efficient detection and mitigation. Understanding the relationship between the intensity of attack and performance metrics will help free NDN networks from IFA. Researchers will be able to come up with stronger IFA defences. It will allow the NDN network's real-world applications.

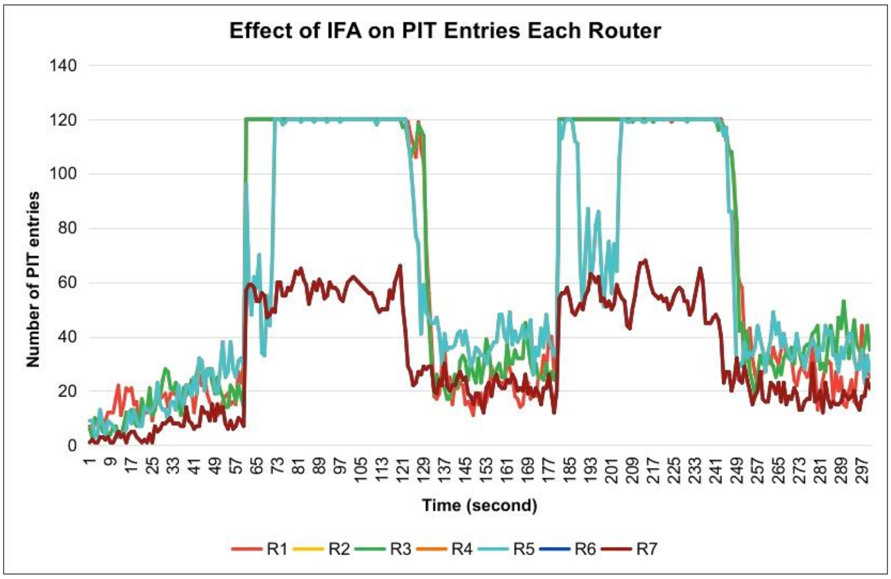

Figure 7. PIT entries each router

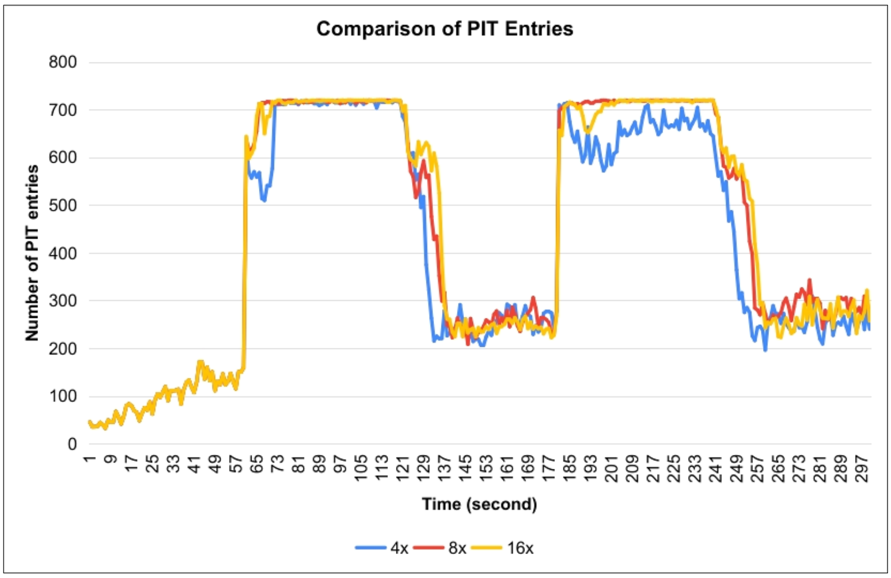

Figure 8. Total PIT entries 4x, 8x, 16x

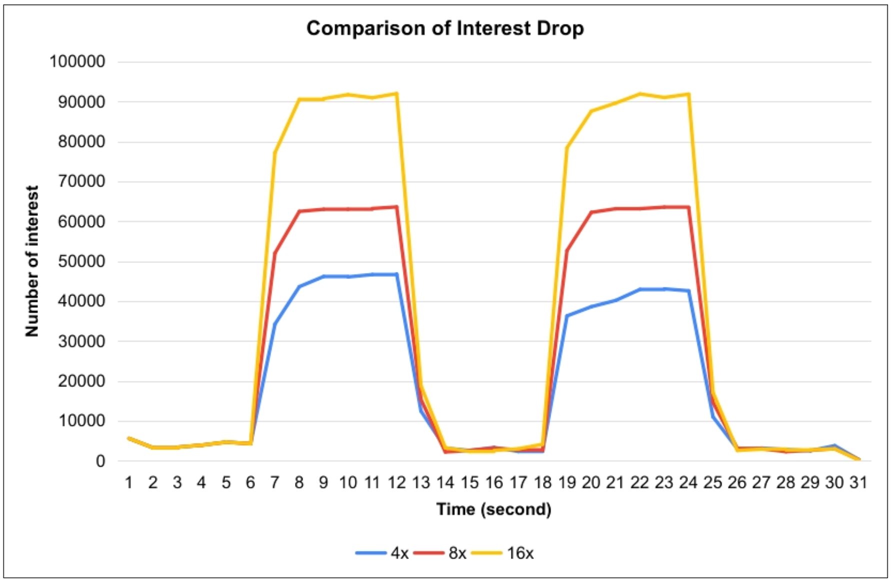

Figure 9. Comparison of interest drop 4x, 8x, 16x

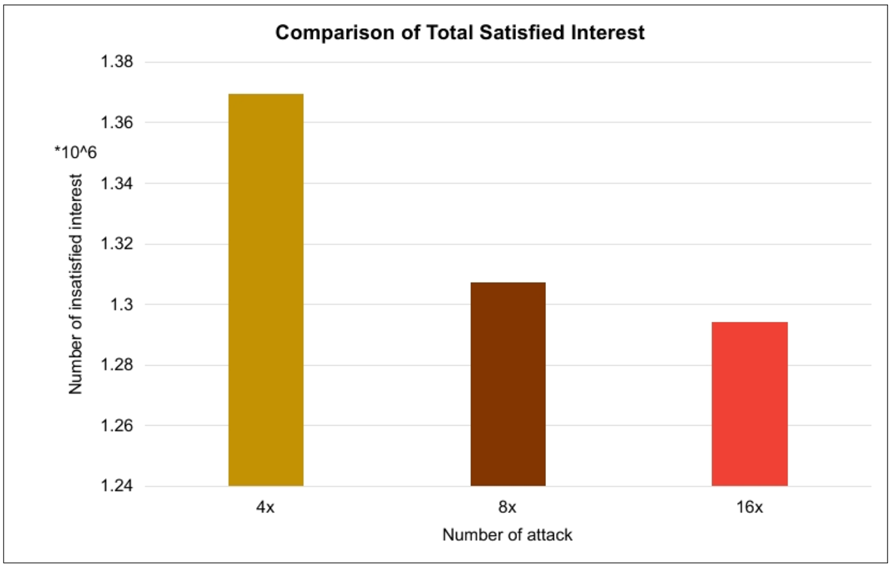

Figure 10. Comparison of total satisfied interest 4x, 8x, 16x

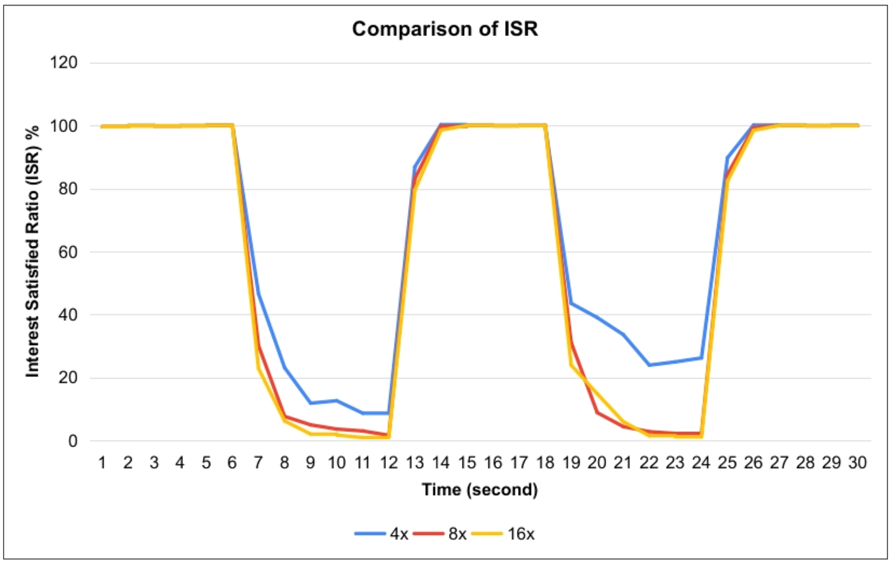

Figure 11. Comparison of ISR 4x, 8x, 16x

Interest Flooding Attack (IFA) affects all routers as it floods all the routers' Pending Interest Table (PIT) with high interest entering each router, as shown in Figure 7. The high PIT entries during IFA show the network is congested, which may lead to network issues. The representation shows that the IFA significantly affects the network's performance. As we analyze the patterns of individual routers against the general directional flow of PIT entries, we get helpful insights into how successful the attack is currently, and later on, we may see its impact on the network. The rise in PIT entries indicates that routers are getting flooded with interest packets, which can lead to delays or drops and add to congestion. Understanding how and where multiplication happens is essential to developing suitable proactive defence strategies. By identifying routers that show a significant increase in PIT usage, administrative actions can be taken for those routers that can help mitigate IFA. This assessment effectively illustrates the outcomes of the attack and indicates the necessity of making defences proactive. The eventual outcome of this work is to give insights that can help design a better architecture of NDNs that are resilient to IFA attacks and their subsequent impacts on performance.

The varying performance of PIT entries under different attack levels is represented in Figure 8. A trade-off exists between the attacks' intensity and the performance of the PIT entry rate. As attacks increase gradually from 4x, 8x, and 16x from the original rate, the PIT entries rise sharply but decline early. More attacks appear to reach faster saturation in PIT entries, resulting in a quicker drop when the network can no longer support the load. The graph displays a steady trend across multiple attack levels, with increased PIT entries corresponding to the growth of attack values and a gradual reduction. Even though the overall pattern appears predictable, like similar attack levels, the precise timings for peak and saturation depend on the attack level. It shows that the performance of PIT (whatever that is) against the attacks varies widely. Researchers need to understand how the performance of the named data networking changes by attacks. It can help inform better security mechanisms that stop PIT entries from going too high. PIT congestion is crucial for reducing vulnerability to interest flooding attacks. Understanding these interactions can help network designers implement better adaptive defences to preserve system stability and efficiency against attacks.

The comparison of interest drops for various attack sizes (4x, 8x, 16x) over time is shown in Figure 9. The drop line's steepness corresponds to the attack level's magnitude. As the attack size increases, the number of interest drops also increases sharply. The attack with 16x makes the highest drop in interest, followed by 8x and 4x. It suggests that more aggressive attacks overload the system, resulting in a more significant loss of interest packets. The graph also shows that while all attack levels create interest drops precisely upon the attack, they all return to stability when the attack stops. Without an attack, the interest drop rate remains low and stable over the period examined. It shows that the number of dropped interests depends on the attack size, meaning that the network is prone to losing much interest during a heavy attack. Recognizing these trends will be helpful in the optimization of NDN security measures and in formulating solutions against excess interest drops during attacks.

Figure 10 demonstrates the number of satisfied interests for different attack values and how the performance of the attacked network decreases as the attack value increases. The attack value increases from 4x to 8x to 16x, and satisfied interests decrease significantly. It means that the higher the attack rate, the more clogged the network is, and legitimate interest requests do not get through. The bar graph shows that at the least attack level (4x), the system can still handle an enormous load of requests. However, with the increase in attack level (8x and 16x), the system is getting increasingly congested; thus, fewer interest requests are being satisfied. A lower number of satisfied interests does not mean better performance but rather shows that an interest flooding attack has severely affected the network. The attack traffic uses up the network's resources, and the ability to process legitimate requests is lessened, thereby reducing the performance of the whole network. The results highlight the importance of robust countermeasures to tackle excess interest flooding to sustain service availability.

The Interest Satisfaction Ratio (ISR) matched with the different attack intensities of 4x, 8x, and 16x over time is illustrated in Figure 11. The findings suggest that an initial interest request may still yield a high ISR. However, the satisfaction rate declines quickly after the attacks occur. At the beginning stages, show an ISR of approximately 100%, implying that the network can successfully satisfy interest requests. However, as the attack occurs and increases, the ISR with the 16x attack drops significantly, also quicker than the 8x or 4x attacks. It implies that higher attack levels can overwhelm a network earlier than lower intensity, resulting in a lower rate of satisfied interests.

On the other hand, both the 8x and 4x attacks have a much steadier decline and maintain a relatively high ISR throughout a longer time frame. This data suggests that the interest processing capability is retained during lower attack intensity. As a result, the network becomes utterly congested at the same attack intensity, resulting in nearly 0 ISR, much quicker than at a lower attack intensity of 4x. The periodic recovery in ISR in the graph signifies periods when the network can satisfy interests due to decreased attack intensity phases. A high ISR at all time frames means the overall network performance is good and resilient and provides stability to satisfy interested interest requests after the conditions of a significant interest flooding attack off.

The results of testing the classification algorithm on the reduced dataset are shown in Table 3. The table provides performance metrics for feature selection algorithms and machine learning classifiers. The first column lists various feature selection algorithms, including Full feature, Information Gain, Gain ratio, Wrapper NB, and CFS. These algorithms are used to select relevant features for the machine learning models. The subsequent columns represent different machine learning classifiers: Naive Bayes, J48, Random Forest, and Bayesian Network. Each classifier is evaluated based on several performance metrics.

The performance metrics in the table include False Positive (FP) rate, True Positive (TP) rate, Precision, and F-measure. These metrics measure the accuracy, reliability, and effectiveness of the classifiers. The values in the table correspond to the performance metrics for each combination of feature selection algorithm and classifier. The table also includes an "Average" row that calculates the average values of the performance metrics across all feature selection algorithms and classifiers. Table 4 supplies insights into implementing various classifiers and feature selection algorithms, allowing comparison and evaluation of their effectiveness in the given context.

Table 4. Comparison of Metrics Evaluation Tree Topology

No.	Feature selection algorithm	Naive bayes				J48				Random forest				Bayesian network			
		TP rate	FP rate	Precision	F-measure	TP rate	FP rate	Precision	F-measure	TP rate	FP rate	Precision	F-measure	TP rate	FP rate	Precision	F-measure
1	Full Fitur [20]	0.931	0.077	0.931	0.931	0.997	0.003	0.997	0.997	0.995	0.006	0.995	0.995	0.982	0.018	0.982	0.982
2	Information Gain [29]	0.926	0.081	0.926	0.926	0.998	0.002	0.998	0.998	0.998	0.002	0.998	0.998	0.984	0.017	0.984	0.984
3	Gain ratio [29]	0.924	0.085	0.925	0.924	0.997	0.003	0.997	0.997	0.998	0.003	0.998	0.998	0.983	0.017	0.983	0.983
4	Wrapper NB	0.924	0.063	0.925	0.923	0.977	0.023	0.977	0.977	0.975	0.026	0.975	0.975	0.970	0.033	0.970	0.970
5	**CFS**	**0.883**	**0.127**	**0.883**	**0.882**	**0.999**	**0.001**	**0.999**	**0.999**	**1.000**	**0.000**	**1.000**	**1.000**	**0.999**	**0.001**	**0.999**	**0.999**
	Average	0.918	0.086	0.918	0.917	0.993	0.007	0.993	0.993	0.993	0.007	0.998	0.993	0.984	0.017	0.984	0.984

The Table shows that both J48 and Random Forest yield high True Positives (TP) rates and CFS plots well under these classifiers. Furthermore, Random Forest returns the highest TP rate (1.000) with CFS, which indicates that Random Forest ultimately classified the characteristics of those cases' combination variables correctly. The Bayesian Network also provides strong performance, particularly with CFS, and generated the lowest rate of False Positives (0.001) and the highest rate of precision (0.999). In comparison, Naïve Bayes, while a popular classification option, generally produced lower TP rates and higher FP rates than the other classifiers tested. Lastly, the 'Average' row at the base of the table provides a general overview of the effectiveness of the classifiers. J48 and Random Forest produced the most consistent and reliable outcomes given the average TP rate of 0.993, precision of 0.993, and F-measure of 0.993. Overall, the information in the table exponentially aided the decision-making process related to which classifier-feature selection combination to optimize detection accuracy and reduce false positives, which ultimately fails or improves the system's performance in the given scenario.

The following is a table of the results of the features chosen by the feature selection algorithm in several attack scenarios. The attacker sends 4x, 8x, and 16x the number of interest packets transmitted by legitimate consumers. The simulation is carried out using tree topologies. The number of full features is 18 (eighteen), and the number of selected features based on the feature selection algorithm can be noticed in Table 4. Additionally, the accuracy of the classification algorithm is shown in Table 5.

Table 5. Number of Selected Feature

No.	Feature selection algorithm	Number of selected feature		
		4x	8x	16x
1	Full Fitur [20]	18	18	18
2	Information Gain [29]	9	9	9
3	Gain Ratio [29]	9	9	9
4	Wrapper NB	5	2	5
5	**CFS**	**3**	**3**	**3**

The number of features chosen by the feature selection methods for each attack intensity (e.g., 4x, 8x, and 16x) is displayed in Table 5. As observed, the Full Feature selection method maintains all features (i.e., 18) through each attack intensity, meaning that dimensionality is not reduced. The Information Gain and Gain Ratio selections maintain nine features, indicating that it uses the criteria from previous studies to identify the most relevant features. The Wrapping Naïve Bayes (NB) selection algorithm is not consistent as it occurs by identifying five features for the 4x and 16x attacks. In comparison, it only uses two features for the 8x attack. It indicates that the Wrapping NB selection method reacts independently of the attack intensity and determines the most important features to use for each attack separately. CFS identifies the least number of features because it consistently identifies only three features independent of attack intensity, suggesting it is effective at filtering redundant or less significant features while maintaining a strong performance at each attack intensity. Reducing the number of features through the feature selection process will help the various algorithms in computational efficiency and classification performance. Overall, the results in Table 4 suggest that CFS could achieve the best efficiency.

Table 6 shows classification accuracy results for various classification algorithms used for different feature selection methods, with a collection of instances subjected to a 16x attack. The results show a clear relationship between feature selection and classification accuracy: fewer or fewer selected features can yield greater classification accuracy, as in the Correlation-Based Feature Selection (CFS) algorithm, which generates the most excellent accuracy. Although it only selects three features, the CFS method has corresponding performance (99.79% accuracy for J48) and outstanding performance (99.98% accuracy for Random Forest and 99.88% for Bayesian Network) across several classifiers. It provides evidence that a few non-less relevant features can increase classification results. As a comparison, all classifiers yield good accuracy using Random Forest, over 99% accuracy across all feature selection methods implemented, and J48 accuracy appears similar for CFS and Information Gain methods, all above 99.7%. Using CFS can significantly enhance classification accuracy while reducing computational complexity. The classification algorithm comparison under the 16x attack scenario is further illustrated in Figure 12.

Table 6 showcases the accuracy outputs of different classification algorithms under an 8x attack in a tree topology. From the results, it is noticeable that the algorithms for feature selection influence the classifier's performance, with the CFS algorithm being the most accurate across classifiers. The best classifier is the Bayesian Network classifier, followed closely by CFS, with accuracy reaching 99.98%, showing that this classifier is highly adept in working with a dataset with a limited number of features with high relevancy. Using the CFS method, the Random Forest classifier takes a slight step down in accuracy, reaching 99.97%, indicating that this classifier also performs well in utilizing the CFS method. The J48 classifier also produced a high level of accuracy as a whole, with the highest accuracy being 99.96% when CFS was used as the feature selection method; this only shows that a decision tree-based classifier can classify successfully based on means of optimized features, as is the case in this scenario. Differently, Naïve Bayes was shown to yield the lowest amount of accuracy in all cases, with the highest amount of accuracy coming in at 94.81%, due to the entirety of the feature set being utilized. It implies that as features are eliminated, Naïve Bayes would not perform similarly to other classifiers. Wrapper NB also yielded the lowest amount of accuracy through a comparison with different classifiers, notably the Bayesian Network classifier (95.05%) and Random Forest classifier (95.39%), as both other classifiers demonstrated their proficiency despite feature selection in feature selection at an 8x attack.

Table 6. The Results of Accuracy Algorithm in Tree Topology 16x

No.	Feature selection algorithm	Accuracy (%)			
		Naive Bayes	J48	Random Forest	Bayesian Network
1	Full feature [20]	92.63	99.61	99.42	98.13
2	Information Gain [29]	92.73	99.70	99.77	98.35
3	Gain ratio [29]	92.73	99.70	99.77	98.35
4	Wrapper NB	94.75	97.21	96.98	96.06
5	**CFS**	**88.35**	**99.79**	**99.98**	**99.88**

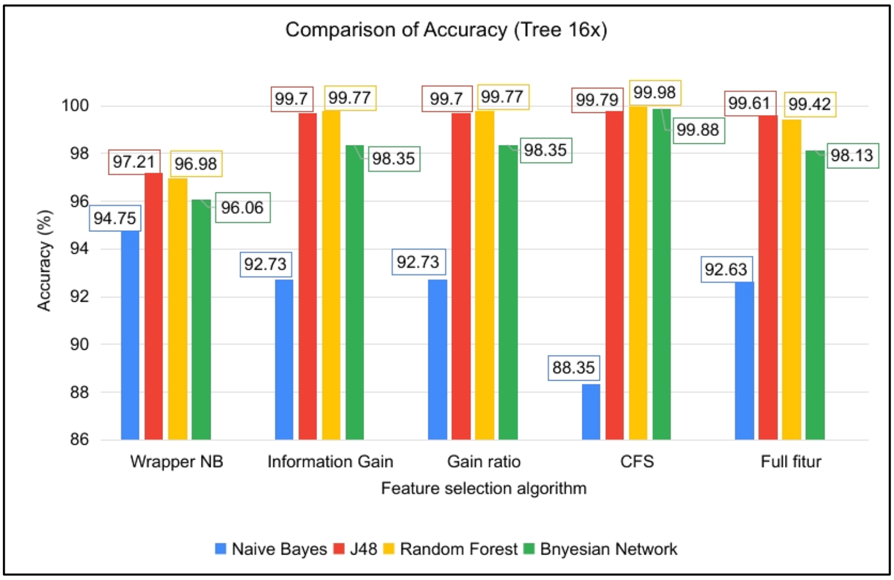

Figure 12. Comparison of accuracy on tree topology 16x

Figure 13 displays a comparative assessment of the classification accuracy of various classification algorithms operating under the 8x attack scenario in a tree topology. The analysis shows that the Random Forest classifier has the highest accuracy in all algorithms tested in Graph 1 and Graph 2 for feature selection; this further supports the utility of Random Forest classification when optimal features are selected. Across each of the algorithms used for feature selection, the CFS algorithm selected the least number of features compared to the other feature selection methods while maximizing accuracy, with the Bayesian Network reaching 99.98% accuracy and Random Forest reaching 99.97% accuracy. It demonstrates the value of selecting the most relevant features for increased classification accuracy. The J48 classifier showed additional increases in accuracy across feature selections, with its highest perceived accuracy reaching 99.96% using CFS feature selection. Therefore, decision-tree-based classifiers have a significantly positive influence after optimum feature selection.

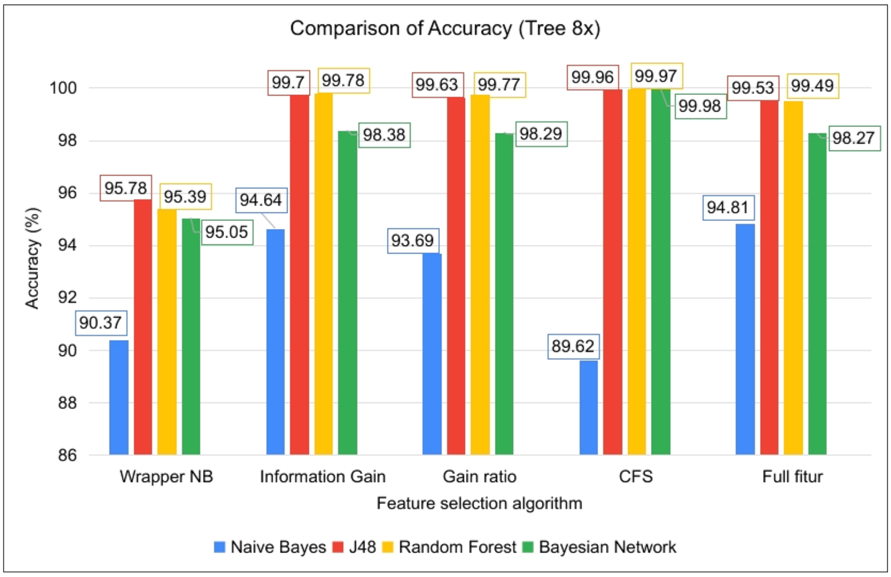

Figure 13. Comparison of accuracy on tree topology 8x

Table 7 displays the accuracy results of different classification algorithms within a tree topology during a 4x attack. As observed, the Random Forest classifier with the CFS feature selection algorithm achieves the best accuracy of 100%. It shows that selecting relevant features improves classification performance and allows an algorithm to make accurate classifications. The J48 classifier, on the other hand, shows a substantial accuracy across all feature selection strategies, achieving the best accuracy of 99.98% when the wrapper NB algorithm is used. It indicates that J48 takes advantage of optimal feature selection strategies. Similarly, the Bayesian Network classifier achieves the best accuracy of 99.86% with the CFS feature selection method, which proves that the algorithm can handle a refined feature set.

Table 7. The Results of Accuracy Algorithm in Tree Topology 8x

No.	Feature selection algorithm	Accuracy (%)			
		Naive Bayes	J48	Random Forest	Bayesian Network
1	Full Fitur [20]	94.81	99.53	99.49	98.27
2	Information Gain [29]	94.64	99.70	99.78	98.38
3	Gain ratio [29]	93.69	99.63	99.77	98.29
4	Wrapper NB	90.37	95.78	95.39	95.05
5	CFS	89.62	**99.96**	**99.97**	**99.98**

In contrast, Naïve Bayes continues to achieve the worst accuracy of all the classifiers. Using the CFS feature selection method, Naïve Bayes achieves an accuracy of 86.75%, indicating its sensitivity in reduced feature set methodologies. However, it achieves an improved accuracy of 91.88% using the full feature selection methodology. Overall, the best results for the 4x attack are combined with the Random Forest classifier and CFS feature selection algorithm. Figure 14 compares classification accuracy results under the 4x attack conditions.

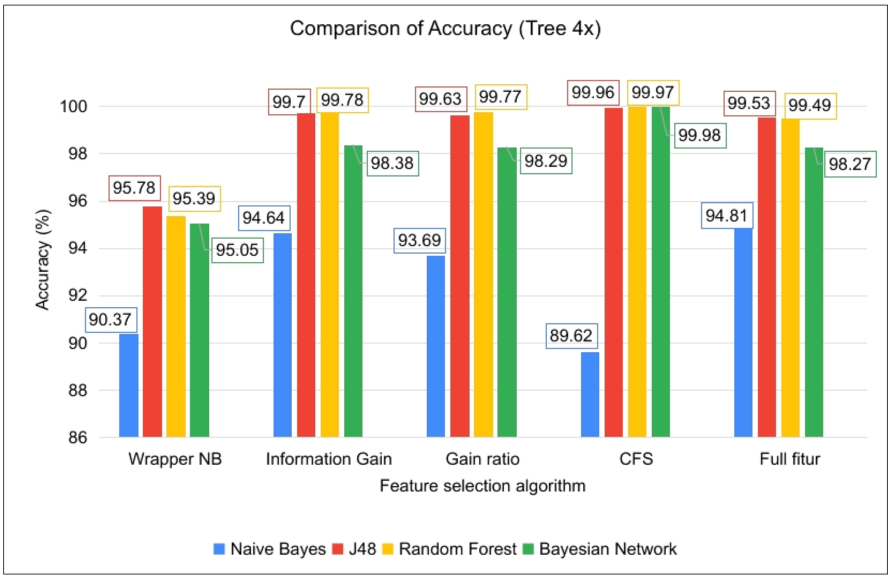

Figure 14. Comparison of accuracy on tree topology 4x

The model development time in this study is critical; removing tested and noted unnecessary features improves computation time while enhancing classification efficiency. Only choosing the features the model developed finds practical and necessary will significantly improve classification efficiency after development, as the developed model can operate on the data more quickly. It enhances the model's overall performance and reduces resources consumed to maintain the capabilities following computational processing, making it useful for practical catastrophes in the field. A tabular comparison in Table 8 demonstrates the relative computational time for the reduced dataset. Further, it shows the results and impacts of feature selection on efficiency and processing.

Table 8. The Results of Accuracy Algorithm in Tree Topology 4x

No.	Feature selection algorithm	Accuracy (%)			
		Naive Bayes	J48	Random Forest	Bayesian Network
1	Full Fitur [20]	91.88	99.89	99.51	98.19
2	Information Gain [29]	90.53	99.93	99.83	98.27
3	Gain ratio [29]	90.92	99.85	99.8	98.28
4	Wrapper NB	92.15	99.98	99.97	99.92
5	**CFS**	**86.75**	**99.92**	**100**	**99.86**

Based on Table 8, building a classification model improves computational efficiency by removing irrelevant features from the dataset. This feature reduction allows the model to be trained faster while maintaining or improving accuracy. Among the various classification algorithms tested, the Naïve Bayes algorithm is the fastest in computation time, making it an ideal choice for situations where rapid processing is required. However, its drawback lies in its lower accuracy compared to other algorithms. In contrast, the Random Forest algorithm takes the longest to build the model, mainly when using the complete set of features. Despite its high computational cost, Random Forest consistently delivers the highest accuracy, demonstrating a trade-off between processing time and prediction performance.

Furthermore, all classification algorithms, except for Naïve Bayes, improve speed and accuracy when applying feature selection techniques. The table also highlights that as the number of features decreases, the computational time for all algorithms is significantly reduced. The CFS feature selection algorithm has the fewest features, resulting in the shortest model-building time across all classification methods. Feature selection plays a crucial role in optimizing model efficiency. Figure 15 illustrates a more detailed comparison of the average time consumed constructing the model, further emphasizing the impact of feature selection on computation time and classification performance (see Table 9).

Table 9. The Comparison of Time Consumption Algorithm to Build Model on Tree Topology

No.	Feature selection algorithm	Time consumption 16x (second)				Time consumption 8x (second)				Time consumption 4x (second)			
		Naive Bayes	J48	Random Forest	Bayesian Network	Naive Bayes	J48	Random Forest	Bayesian Network	Naive Bayes	J48	Random Forest	Bayesian Network
1	Full Fitur [20]	0.04	0.27	6.19	0.13	0.04	0.72	5.43	0.22	0.17	0.59	4.98	0.37
2	Gain ratio [29]	0.04	0.16	4.84	0.07	0.02	0.13	4.43	0.06	0.02	0.09	4.62	0.05
3	Wrapper NB	0.02	0.43	4.27	0.14	0.01	0.05	2.38	0.02	0.01	0.06	3.03	0.04
4	Information Gain [29]	0.02	0.16	4.06	0.06	0.03	0.12	4.38	0.05	0.02	0.13	3.84	0.05
5	**CFS**	**0.01**	**0.07**	**1.96**	**0.02**	**0.01**	**0.07**	**1.92**	**0.03**	**0.01**	**0.06**	**2.6**	**0.03**
	Average	0.026	0.218	4.264	0.084	0.022	0.218	3.708	0.076	0.046	0.186	3.814	0.108

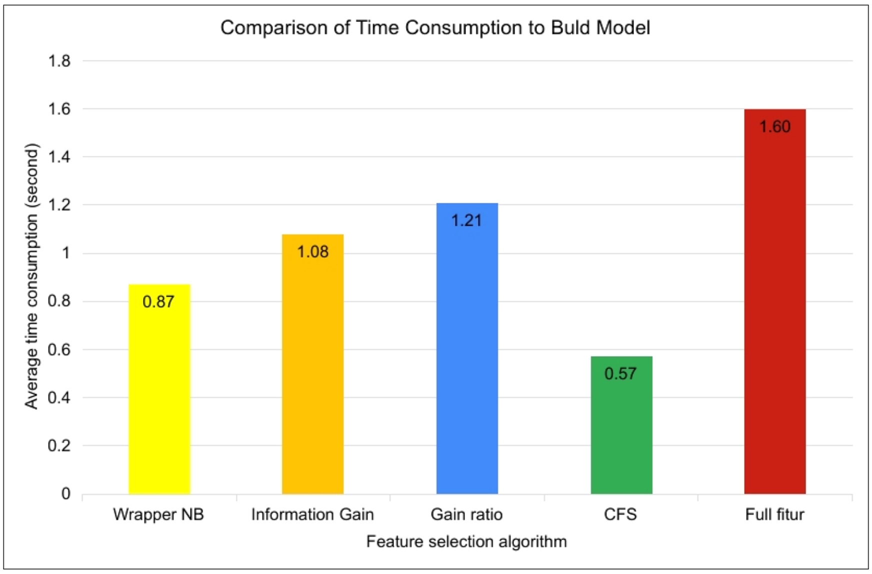

Figure 15. Comparison of time consumption to build model

The bar graph compares the average time to develop a model using distinct feature selection algorithms. These findings show that feature selection significantly decreases computation time compared to all feature methods, as indicated by implementing various methods. In the feature selection, CFS is the fastest algorithm on average, anywhere close to 0.57 seconds to do it perfectly as the most computationally effective one. The Wrapper NB method takes seconds, an average of 0.87 seconds. It shows a lot of speed improvement compared to other methodologies. Overall, the Information Gain and Gain Ratio are slower on average, taking 1.08 seconds for each to execute while performing better in increased efficiency (not the point with the most optimal time reduction). The complete feature set gives the highest time computation (1.60 sec on average), significantly longer than every other method. It exposes the limitation of training a model even with all the available features, which drags the computational requirement absurdly even if we do not notice any increase in performance. Overall, the chart stresses the impact of feature selection on efficiency improvement in machine learning models. Results validate that CFS has the least time consumption, followed by the Wrapper NB methods and then the Information Gain, Gain Ratio. Although complete, the whole feature approach is the least efficient, proving that feature redundancy/irrelevance should be cleaned and cut when faster model building is desired.

4- Conclusion

It is essential to detect Interest Flooding Attacks (IFA) in networks based on Named Data Networking (NDN) to ensure network performance is not hampered. Detection should be fast and accurate to minimize the effects of the attack. Based on our experiments in the present work, not all features can be reliably used for an attack presence indication. Hence, the feature selection process is crucial, dramatically affecting the detection system's accuracy and speed. This study used four feature selection algorithms: wrapper Naive Bayes, gain ratio, information gain, and correlation-based feature selection (CFS) to identify significant features. The CFS algorithm was the most efficient among these in determining the fewest features compared to the other methods. The research paper validated the chosen features using four classification algorithms: Naive Bayes, J48, Random Forest, and Bayesian Network. The best results were obtained when combining CFS and Random Forest, which achieved a fantastic 100% detection rate for three essential features extracted from 18 total features. This method offers excellent efficiency, as it reduces the time to build the model by 47.8% and increases the detection accuracy. The following study will test the CIFA and ICIFA attacks, the extended version of the original IFA attack. This feature selection process will be modified to suit advanced detections. The research will evaluate the CIFA and ICIFA attack types to develop further methodologies to provide the required capabilities. The study examines advanced types of attacks so that the network security offered by NDN does not go down; thus, the reliability and resilience of NDN are maintained.

5- Declarations

5-1- Author Contributions

Conceptualization, J. and N.R.S.; methodology, E.M.; software, J.; validation, J., E.M., and N.R.S.; formal analysis, J.; investigation, E.M.; resources, E.M.; data curation, J.; writing—original draft preparation, J.; writing—review and editing, E.M.; visualization, J.; supervision, N.R.S.; project administration, E.M. All authors have read and agreed to the published version of the manuscript

5-2- Data Availability Statement

Data was obtained from Naveen Kumar (nk10121989) and are available “https://github.com/nk10121989/NDN-IFA-Feature-Selection” with the permission of nk10121989 by request.

5-3- Funding

This work was supported by the Indonesian Education Scholarship (BPI) program, administered by the Center for Higher Education Funding and Assessment (PPAPT) Ministry of Higher Education, Science, and Technology of Republic Indonesia, and the Indonesian Endowment Fund for Education (LPDP) of the Republic of Indonesia.

5-4- Institutional Review Board Statement

Not applicable.

5-5- Informed Consent Statement

Not applicable.

5-6- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

6- References

[1] Jaffri, Z. ul A., Ahmad, Z., & Tahir, M. (2013). Named Data Networking (NDN), New Approach to Future Internet Architecture Design: A Survey. International Journal of Informatics and Communication Technology (IJ-ICT), 2(3), 155–164. doi:10.11591/ij-ict.v2i3.5122.

[2] NDN. (2025) A Future Internet Architecture, Named Data Networking (NDN), Palo Alto, United States. Available online: https://www.named-data.net (accessed on November 2025)

[3] Afanasyev, A., Burke, J., Refaei, T., Wang, L., Zhang, B., & Zhang, L. (2018). A Brief Introduction to Named Data Networking. MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM), 8599682. doi:10.1109/milcom.2018.8599682.

[4] Saxena, D., Raychoudhury, V., Suri, N., Becker, C., & Cao, J. (2016). Named Data Networking: A survey. Computer Science Review, 19, 15–55. doi:10.1016/j.cosrev.2016.01.001.

[5] Aggarwal, M., Nilay, K., & Yadav, K. (2017). Survey of named data networks: future of internet. International Journal of Information Technology, 9(2), 197–207. doi:10.1007/s41870-017-0014-y.

[6] Kumar, N., Singh, A. K., Aleem, A., & Srivastava, S. (2019). Security Attacks in Named Data Networking: A Review and Research Directions. Journal of Computer Science and Technology, 34(6), 1319–1350. doi:10.1007/s11390-019-1978-9.

[7] Kumar, N., Singh, A. K., & Srivastava, S. (2017). Evaluating machine learning algorithms for detection of interest flooding attack in named data networking. Proceedings of the 10th International Conference on Security of Information and Networks, 299–302. doi:10.1145/3136825.3136864.

[8] Al-Share, R. A., Shatnawi, A. S., & Al-Duwairi, B. (2022). Detecting and Mitigating Collusive Interest Flooding Attacks in Named Data Networking. IEEE Access, 10, 65996–66017. doi:10.1109/access.2022.3184304.

[9] Benmoussa, A., Kerrache, C. A., Lagraa, N., Mastorakis, S., Lakas, A., & Tahari, A. E. K. (2022). Interest Flooding Attacks in Named Data Networking: Survey of Existing Solutions, Open Issues, Requirements, and Future Directions. ACM Computing Surveys, 55(7), 1–37. doi:10.1145/3539730.

[10] Gasti, P., Tsudik, G., Uzun, E., & Zhang, L. (2013). DoS and DDoS in Named Data Networking. 2013 22nd International Conference on Computer Communication and Networks (ICCCN), 1–7. doi:10.1109/icccn.2013.6614127.

[11] Xin, Y., Li, Y., Wang, W., Li, W., & Chen, X. (2016). A Novel Interest Flooding Attacks Detection and Countermeasure Scheme in NDN. 2016 IEEE Global Communications Conference (GLOBECOM), 1–7. doi:10.1109/glocom.2016.7841526.

[12] Xing, G., Li, X., & Hou, R. (2024). A Reconstruction Forest-Based Interest Flooding Attack Detection Method in Named Data Networking. 2024 International Conference on Computing, Networking and Communications (ICNC), 823–829. doi:10.1109/icnc59896.2024.10556003.

[13] Kai Wang, Huachun Zhou, Yajuan Qin, Jia Chen, & Hongke Zhang. (2013). Decoupling malicious Interests from Pending Interest Table to mitigate Interest Flooding Attacks. 2013 IEEE Globecom Workshops (GC Wkshps), 963–968. doi:10.1109/glocomw.2013.6825115.

[14] Tang, J., Zhang, Z., Liu, Y., & Zhang, H. (2013). Identifying Interest Flooding in Named Data Networking. 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, 306–310. doi:10.1109/greencom-ithings-cpscom.2013.68.

[15] Zhi, T., Luo, H., & Liu, Y. (2018). A gini impurity-based interest flooding attack defence mechanism in NDN. IEEE Communications Letters, 22(3), 538–541. doi:10.1109/LCOMM.2018.2789896.

[16] Hou, R., Han, M., Chen, J., Hu, W., Tan, X., Luo, J., & Ma, M. (2019). Theil-Based Countermeasure against Interest Flooding Attacks for Named Data Networks. IEEE Network, 33(3), 116–121. doi:10.1109/MNET.2019.1800350.

[17] Dhiwar, C., Kumari, M. K., Tripathi, N., & Joshi, P. (2025). A Statistical Anomaly Detection Approach to Detect Induced Interest Flooding Attacks in NDN. 2025 17th International Conference on COMmunication Systems and NETworks (COMSNETS), 1305–1310. doi:10.1109/comsnets63942.2025.10885725.

[18] Karami, A., & Guerrero-Zapata, M. (2015). A hybrid multiobjective RBF-PSO method for mitigating DoS attacks in Named Data Networking. Neurocomputing, 151, 1262–1282. doi:10.1016/j.neucom.2014.11.003.

[19] Zhi, T., Liu, Y., Wang, J., & Zhang, H. (2020). Resist Interest Flooding Attacks via Entropy-SVM and Jensen-Shannon Divergence in Information-Centric Networking. IEEE Systems Journal, 14(2), 1776–1787. doi:10.1109/JSYST.2019.2939371.

[20] Afanasyev, A., Mahadevan, P., Moiseenko, I., Uzun, E., & Zhang, L. (2013). Interest flooding attack and countermeasures in named data networking. 2013 IFIP Networking Conference, IEEE, 22-24 May, 2013, Brooklyn, United States.

[21] Mandapati, S. G., Ranaweera, C., & Doss, R. (2024). Advancing NDN Security for IoT: Harnessing Machine Learning to Detect Attacks. 2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks - Supplemental Volume (DSN-S), 53–55. doi:10.1109/dsn-s60304.2024.00023.

[22] Benmoussa, A., Tahari, A. el K., Kerrache, C. A., Lagraa, N., Lakas, A., Hussain, R., & Ahmad, F. (2020). MSIDN: Mitigation of Sophisticated Interest flooding-based DDoS attacks in Named Data Networking. Future Generation Computer Systems, 107, 293–306. doi:10.1016/j.future.2020.01.043.

[23] Wu, Z., Feng, W., Yue, M., Xu, X., & Liu, L. (2020). Mitigation measures of collusive interest flooding attacks in named data networking. Computers & Security, 97, 101971. doi:10.1016/j.cose.2020.101971.

[24] Benarfa, A., Hassan, M., Compagno, A., Losiouk, E., Yagoubi, M.B., & Conti, M. (2019). ChoKIFA: A New Detection and Mitigation Approach Against Interest Flooding Attacks in NDN. Wired/Wireless Internet Communications. WWIC 2019. Lecture Notes in Computer Science, vol 11618. Springer, Cham, Switzerland. doi:10.1007/978-3-030-30523-9_5.

[25] Benarfa, A., Hassan, M., Losiouk, E., Compagno, A., Yagoubi, M. B., & Conti, M. (2021). ChoKIFA+: an early detection and mitigation approach against interest flooding attacks in NDN. International Journal of Information Security, 20(3), 269–285. doi:10.1007/s10207-020-00500-z.

[26] Yang, J., Ding, K., Xue, K., Han, J., Wei, D. S. L., Sun, Q., & Lu, J. (2025). HPR-DS: A Hybrid Proactive Reactive Defense Scheme Against Interest Flooding Attack in Named Data Networking. IEEE Transactions on Networking, 33(4), 1854–1869. doi:10.1109/ton.2025.3546547.

[27] Xing, G., Chen, J., Hou, R., Zhou, L., Dong, M., Zeng, D., Luo, J., & Ma, M. (2021). Isolation Forest-Based Mechanism to Defend against Interest Flooding Attacks in Named Data Networking. IEEE Communications Magazine, 59(3), 98–103. doi:10.1109/MCOM.001.2000368.

[28] Joseph K, S. (2024). Multi-classifier and meta-heuristic based cache pollution attacks and interest flooding attacks detection and mitigation model for named data networking. Journal of Experimental & Theoretical Artificial Intelligence, 36(6), 839–864. doi:10.1080/0952813x.2022.2115141.

[29] Kumar, N., Singh, A. K., & Srivastava, S. (2021). Feature selection for interest flooding attack in named data networking. International Journal of Computers and Applications, 43(6), 537–546. doi:10.1080/1206212X.2019.1583820.

[30] Mandala, S., Ramadhan, A. I., Rosalinda, M., Zaki, S. M., & Weippl, E. (2022). DDoS Detection Using Information Gain Feature Selection and Random Forest Classifier. 2022 2nd International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA), 294–299. doi:10.1109/icicyta57421.2022.10038126.

[31] Jupriyadi, & Kistijantoro, A. I. (2014). Vitality based feature selection for intrusion detection. 2014 International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA), 93–96. doi:10.1109/icaicta.2014.7005921.

[32] Miguel-Andrés, I., Ramos-Frutos, J., Sharawi, M., Oliva, D., Reyes-Dávila, E., Casas-Ordaz, Á., Pérez-Cisneros, M., & Zapotecas-Martínez, S. (2024). Wrapper-Based Feature Selection to Classify Flatfoot Disease. IEEE Access, 12, 22433–22447. doi:10.1109/access.2024.3361936.

[33] Naiem, S., Khedr, A. E., Idrees, A. M., & Marie, M. I. (2023). Enhancing the Efficiency of Gaussian Naïve Bayes Machine Learning Classifier in the Detection of DDOS in Cloud Computing. IEEE Access, 11, 124597–124608. doi:10.1109/access.2023.3328951.

[34] Ferdousi, T., Cohnstaedt, L. W., & Scoglio, C. M. (2021). A Windowed Correlation-Based Feature Selection Method to Improve Time Series Prediction of Dengue Fever Cases. IEEE Access, 9, 141210–141222. doi:10.1109/access.2021.3120309.

[35] Garrett, D., Peterson, D. A., Anderson, C. W., & Thaut, M. H. (2003). Comparison of linear, nonlinear, and feature selection methods for EEG signal classification. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 11(2), 141–144. doi:10.1109/tnsre.2003.814441.

[36] Wisanwanichthan, T., & Thammawichai, M. (2021). A Double-Layered Hybrid Approach for Network Intrusion Detection System Using Combined Naive Bayes and SVM. IEEE Access, 9, 138432–138450. doi:10.1109/access.2021.3118573.

[37] Perez-Diaz, J. A., Valdovinos, I. A., Choo, K. K. R., & Zhu, D. (2020). A Flexible SDN-Based Architecture for Identifying and Mitigating Low-Rate DDoS Attacks Using Machine Learning. IEEE Access, 8, 155859–155872. doi:10.1109/ACCESS.2020.3019330.

[38] Ismail, Mohmand, M. I., Hussain, H., Khan, A. A., Ullah, U., Zakarya, M., Ahmed, A., Raza, M., Rahman, I. U., & Haleem, M. (2022). A Machine Learning-Based Classification and Prediction Technique for DDoS Attacks. IEEE Access, 10, 21443–21454. doi:10.1109/access.2022.3152577.

[39] Faruq, Z. F., Mantoro, T., Catur Bhakti, M. A., & Wandy. (2022). Random Forest Classifier Evaluation in DDoS Detection System for Cyber Defence Preparation. 2022 IEEE 8th International Conference on Computing, Engineering and Design, ICCED, 1–5,. doi:10.1109/ICCED56140.2022.10010341.

[40] Yin, H., Xue, M., Xiao, Y., Xia, K., & Yu, G. (2019). Intrusion Detection Classification Model on an Improved k-Dependence Bayesian Network. IEEE Access, 7, 157555–157563. doi:10.1109/access.2019.2949890.