DQN-IDS: A Deep Reinforcement Learning Approach for Open Set-Enabled Intrusion Detection

Shreyash Tiwari
Computer and Information Science
University of Massachusetts Dartmouth
Dartmouth, MA, USA
stiwari4@umassd.edu

Nathaniel D. Bastian
Electrical Engineering and Computer Science
United States Military Academy
West Point, NY, USA
nathaniel.bastian@westpoint.edu

Gokhan Kul
Computer and Information Science
University of Massachusetts Dartmouth
Dartmouth, MA, USA
gkul@umassd.edu

***Abstract*—Intrusion Detection Systems (IDS) remain vulnerable to zero-day attacks that manifest themselves as previously unseen traffic patterns. Traditional neural IDS models, constrained by closed-world assumptions, often misclassify such traffic as benign, leading to significant security risks. We present DQN-IDS, a deep reinforcement learning framework that integrates a Convolutional Neural Network (CNN) for feature extraction with a Deep Q-Network (DQN) for uncertainty-aware decision-making. Unlike threshold-based open-set methods, DQN-IDS dynamically learns to separate known and unknown traffic using softmax-derived confidence metrics maximum probability, probability gap, and entropy as its state representation. Evaluated on the CICIDS-2017 and UNSW2015 datasets, the proposed system achieves a binary F1-score of 97.8% (known vs. unknown) and reduces missed zero-day traffic compared to state-of-the-art threshold-based approaches. The DQN stage introduces negligible runtime overhead relative to CNN inference, yielding a deployable two-stage open-set NIDS suitable for IoT and other resource-constrained environments.**

I. Introduction

Zero-day attacks exploit new or previously unexplored vulnerabilities in a system, making them difficult to detect using traditional methods [1]. Deep learning–based Network Intrusion Detection Systems (NIDS) are inherently limited by the scope of their training datasets [2]. As a result, they may misclassify previously unseen attack patterns as benign traffic, creating significant security risks. To mitigate this limitation, prior work has explored threshold-based approaches such as OpenMax [3], varMax [1], open set recognition [4], and energy-based confidence calibration [5] to handle low-confidence predictions. However, these methods typically rely on manually tuned thresholds, which limits their adaptability in real-time or evolving network environments where confidence margins vary across traffic distributions [6], [7]. Open set recognition (OSR) is a machine learning paradigm that enables models to classify known classes while identifying samples that do not belong to any known category. Unlike closed-set recognition, which assumes that all test samples belong to predefined classes, OSR explicitly accounts for unknown or novel inputs, making it particularly suitable for intrusion detection in dynamic environments. In the context of NIDS, OSR helps detect previously unseen traffic that might otherwise be misclassified as benign or as a known attack class. This capability is critical in real-world networks, where new attack vectors frequently emerge and models must generalize beyond their training data. However, many OSR-enabled NIDS suffer from degraded performance on known classes [8]. Recent studies report that existing open-set NIDS methods fail to detect approximately 28% of zero-day traffic on average [1], [9], often due to their reliance on fixed thresholds and manual post-processing strategies. In this paper, we demonstrate that coupling uncertainty-aware features with reinforcement learning (RL) enables robust zero-day detection without relying on predefined thresholds. Our architecture integrates a Convolutional Neural Network (CNN) with a softmax output layer and a Deep Q-Network (DQN) that operates on the CNN's confidence outputs to determine whether a sample should be classified as known or unknown. The DQN learns a decision policy based on uncertainty patterns while preserving classification performance on known traffic. This approach avoids static decision rules and does not require labeled unknown samples during training. We present a hybrid framework that integrates a Deep Q-Network (DQN) with a deep neural network backbone. While a CNN is used in this work, the framework can be extended to other deep architectures. The CNN is trained on a subset of classes designated as the known set, while selected classes are withheld to simulate unknown attacks. During evaluation, the system is tested on the full class set, including unseen attack types. As expected, the supervised CNN exhibits lower confidence on unknown samples. To exploit this behavior, softmax-derived confidence features—maximum class probability (P1), probability gap (P1–P2), and Shannon entropy are passed to the DQN, which learns to distinguish known from unknown traffic based on these uncertainty signals. We evaluate the proposed system using the CICIDS-2017 dataset [10] as our main training data, which aggregates multiple types of network activities and exhibits significant class imbalance. To support controlled open-set evaluation, we use a balanced subset of the dataset and

Workshop on Security and Privacy in Standardized IoT (SDIoTSec) 2026
23 February 2026, San Diego, CA, USA
ISBN 978-1-970672-01-5
https://dx.doi.org/10.14722/sdiotsec.2026.23097
www.ndss-symposium.org

intentionally exclude selected classes from training to serve as unknown traffic during testing. We also include UNSW dataset [11] samples as additional unknown samples for our tests While the CNN occasionally assigns high confidence to certain unknown samples, such cases are relatively infrequent and reflect a known limitation of deep classifiers. These scenarios motivate the need for a decision-making module that reasons over confidence patterns rather than relying solely on fixed thresholds. Building on the limitations of existing approaches, most current zero-day detection methods rely on static confidence thresholds to identify unknown samples. In contrast, our proposed framework employs a dynamic, reinforcement learning–based decision process that adaptively distinguishes between high- and low-confidence outputs generated by the CNN, enabling threshold-free open-set intrusion detection. Our experimental results demonstrate the effectiveness of the proposed CNN+DQN framework for open-set intrusion detection. The CNN achieves over 97.38% accuracy on known classes, while the DQN significantly improves the detection of unknown attacks, achieving a binary F1-score of 97.83%. The hybrid system accurately classifies known traffic while effectively separating it from unknown patterns, demonstrating its suitability for zero-day attack detection.

II. Related Works

We surveyed several open-set and NIDS methods and aimed to overcome their key limitations through our proposed approach.

A. Threshold-based Confidence

Threshold-based systems such as varMax [1], [8] use a three-step pipeline to improve confidence-based detection (i) the P1–P2 gap is used to assess prediction ambiguity, (ii) logit variance identifies unfamiliar or uncertain inputs, and (iii) energy-based scoring distinguishes in- versus out-of-distribution samples. Evaluations demonstrate strong performance in identifying zero-day attacks. However, varMax uses confidence-based multi-metric modeling, but its statistical boundaries can be sensitive to shifts, Which can fail to generalize when the underlying dataset distribution shifts or in live network traffic scenarios, limiting its applicability in real-world deployments.

Our CNN + DQN framework removes the dependency on fixed thresholds by transforming the CNN's confidence metrics (P1, P1–P2, entropy) into an adaptive, reinforcement-learning-based decision process. Using centroid-guided rewards and off-policy DQN training, the model dynamically adjusts to evolving traffic patterns, autonomously separating known and unknown attacks. This enables stable, scalable, and real-time open-set intrusion detection without manual calibration, making it significantly more robust and practical than threshold-based approaches.

B. Unsupervised Zero-Day Detection

Unsupervised zero-day detection methods typically focus on identifying deviations in incoming sample distributions. For instance, Fang and Xie [9] combine InfoGAN-based feature learning with OpenMax [3] to estimate the probability of an input belonging to an unknown class. Specifically, InfoGAN is used to learn rich latent feature representations of network traffic, while OpenMax adjusts the classifier's output to account for potential unknown classes. Their evaluation on CICIDS-2017, holding out one attack class at a time, achieves strong accuracy (above 88%). However, this approach has several limitations: it requires retraining for each withheld class, involves extensive parameter tuning (e.g., alpha rank, tail size), incurs high computational costs, and produces elevated false-alarm rates, which reduces its practicality for dynamic network environments.

Our CNN + DQN framework addresses these limitations by modeling multiple withheld attack classes as a single unknown category, reflecting the real-world need to detect previously unseen attacks. By using softmax-derived confidence metrics: $P1$, $P1 - P2$, and Shannon entropy as state features, the system autonomously separates known from unknown traffic without retraining or manually defined thresholds. Combined with centroid-guided rewards and off-policy DQN learning, this approach provides scalable, adaptive, and computationally efficient open-set detection, making it well-suited for real-time deployment in evolving network environments.

C. Reinforcement-Learning IDS

Reinforcement learning (RL) offers a natural framework for open-set recognition because it enables an agent to learn decision-making policies from feedback in the environment, rather than relying solely on labeled training data. In the context of intrusion detection, the RL agent can evaluate the uncertainty or novelty of incoming traffic and adapt its actions to maximize long-term detection performance, which is particularly useful for identifying unknown or zero-day attacks. Prior work illustrates different applications of RL in IDS. For example, Alavizadeh et al. [12] use fixed reward functions with full supervision to classify network traffic in a closed-world setting, while DeROL [13] leverages human-in-the-loop reinforcement learning on NSL-KDD [14] to improve decision reliability. However, these approaches have clear limitations: they either require predefined reward structures and labels or depend on manual human input, which introduces latency and reduces scalability in dynamic network environments. To overcome these limitations, our CNN + DQN framework employs unsupervised, centroid-guided rewards within an off-policy DQN setup. This allows the system to autonomously learn from softmax-derived confidence metrics (P1, P1–P2, entropy) and dynamically distinguish between known and unknown traffic. By removing reliance on fixed thresholds, labels, or human feedback, the approach provides adaptive, real-time zero-day attack recognition, making it both efficient and practical for deployment in evolving network environments.

We have developed **DQN-IDS** over policy-gradient or actor-critic methods because:

- Experience replay stabilizes training in a 3D state space (P1, P1–P2, entropy) which adds sample efficiency

- The binary decision on known/unknown outcomes fits Q-learning natively.
- Fixed targets and ϵ-greedy exploration avoid high-variance updates common in on-policy methods.

TABLE I
OPEN-SET NIDS COMPARISON

Approach	Input	Adapt.	Supervision
Threshold	Logits, P1–P2	Low	None
Unsup.	Feat.+EVT	Med.	None
RL	Raw	High	Full/Human
Ours	P1, P1–P2, H	**High**	**None**

This design combines the strengths of multiple paradigms, it uses confidence-based metrics inspired by thresholding methods, adds unsupervised learning by avoiding the need for labeled unknown samples, and integrates reinforcement learning to adaptively refine decision-making. The approach is also well suited for live-traffic deployment. The CNN can be trained entirely on known traffic, while a small portion of unlabeled real-world traffic can be used to tune and validate the DQN before deployment, yielding a practical zero-day detection model. Because the DQN learns boundaries from uncertainty patterns rather than relying on manually selected thresholds, it can be periodically updated with new unlabeled traffic without retraining the CNN, making the system flexible for continuously evolving network environments.

In summary, our CNN + DQN framework overcomes key limitations of existing IDS approaches by learning from softmax-derived uncertainty features in an unsupervised manner to identify unknown attacks. Unlike traditional threshold-based or closed-set methods, it removes dependence on manual tuning and fixed decision rules. By combining CNN feature extraction with off-policy, centroid-guided DQN learning, the system remains efficient, scalable, and adaptive, enabling real-time open-set intrusion detection in complex, shifting network conditions. This hybrid design integrates unsupervised inference, reinforcement learning, and confidence-based reasoning, making it robust to dynamic behavior and previously unseen attacks while remaining practical for continuous, real-world deployment.

III. METHODOLOGY

The methodology is divided into four parts: (1) threat model, (2) network architecture and design, (3) hyperparameter selection, and (4) the algorithmic design of the confidence-aware DQN used to detect unknown traffic.

A. Threat Model

An effective intrusion detection framework must specify the assumptions under which it operates and the adversarial capabilities it defends against. Our threat model defines the attacker's goals, actions, knowledge exposure, and system boundaries while clarifying what DQN-IDS is and is not designed to resist.

System Assumptions: We assume the NIDS is deployed at an ingress point within an enterprise or IoT network, where it receives raw network-flow features extracted from live traffic. The upstream feature extractor and network monitoring infrastructure are considered trusted. Namely,

- Traffic features cannot be altered post-capture without detection.
- Attackers cannot tamper with flow timestamps, packet statistics, or metadata.
- The CNN and DQN weights are protected against modification at runtime.

Traffic can be malicious, novel, or adversarial, but the model input pipeline (feature sampling, normalization, batching) is assumed uncompromised.

Adversary Goals:

The adversary seeks one or more of the following:

- Evasion attack through crafting traffic resembling benign flows such that it is classified as known benign rather than unknown/zero-day.
- Introduce previously unseen attack vectors with minimal statistical deviation to avoid detection.
- Force the CNN to output high-confidence predictions on malicious flows, bypassing the DQN open-set classifier.
- Gradually shift traffic distribution to distort centroid evolution, weakening unknown detection boundaries over time.

B. Network Architecture and Design

We built a 1D Convolutional Neural Network (CNN) on the CICIDS-2017 dataset using the first 10 out of 15 classes for training, while the remaining 5 minor classes were withheld and introduced only during testing to simulate unknown or zero-day attacks. The withheld classes were combined into a single "unknown" category (label 10), and the CNN is evaluated in a closed-set scenario that includes these samples.

To distinguish known from unknown traffic, we analyze the CNN's softmax outputs using three confidence-based metrics:

- $P1$: Maximum class probability
- $P1 - P2$: Difference between the top two class probabilities
- H: Shannon entropy of the softmax distribution

Shannon entropy (H) effectively captures the overall distribution of predicted probabilities, providing a measure of uncertainty beyond raw logits or individual class scores. While energy-based measures could also be used, we found that softmax outputs alone provide sufficient discriminative information. Entropy complements $P1$ and $P1 - P2$ by reflecting the spread and ambiguity of predictions: known samples typically yield high-confidence, low-entropy distributions, whereas unknown samples produce low-confidence, high-entropy outputs.

Using these three parameters together provides a robust, interpretable, and computationally efficient method for open-set classification. Unlike prior works [1], [8] that combined probabilities with energy-based measures, we focus only on the metrics that offer sufficient discriminative power, avoiding unnecessary calculations. Fig. 1 shows the flow chart and

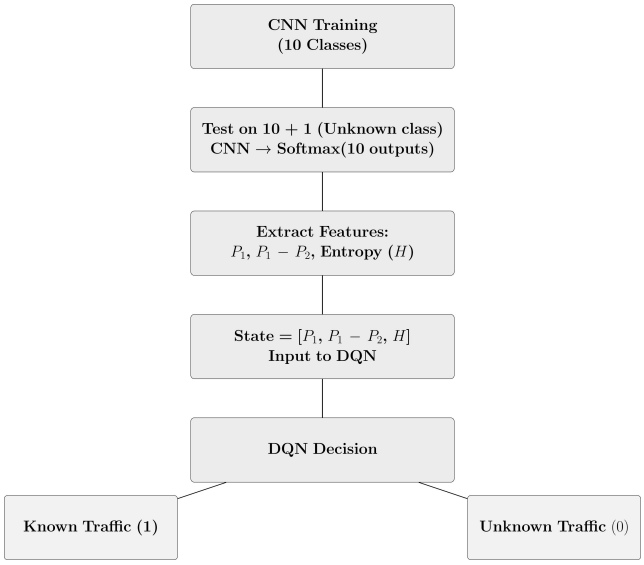

Fig. 1. DQN-IDS Flowchart

Figs. 2, 3 and 4 illustrate the distributions of these metrics for known versus unknown traffic in the test set, confirming the effectiveness of this streamlined approach for distinguishing zero-day attacks.

The CNN architecture is detailed in Table II. It includes three Conv1D layers with increasing filter sizes (8, 24, 32), each followed by ReLU activation, batch normalization, and max pooling. A global average pooling layer and a 48-unit dense layer follow before producing raw logits over the 10 training classes. The output layer uses softmax activation.

TABLE II
CNN ARCHITECTURE (CONV AND DENSE LAYERS)

Convolutional Layers		
Layer	**Params**	**Details**
Input	Shape	$(78, 1)$
Conv1D	$f = 8$	$k = 3$, L2=0.005, pad=same
ReLU+BN+Pool	$p = 2$	—
Conv1D	$f = 24$	$k = 3$, L2=0.005, pad=same
ReLU+BN+Pool	$p = 2$	—
Conv1D	$f = 32$	$k = 3$, L2=0.005, pad=same
ReLU+BN+Pool	$p = 2$	—
GlobalAvgPool	—	—
Dense Layers		
Layer	**Params**	**Details**
Dense	$u = 48$	L2=0.005, ReLU
Dropout	$r = 0.5$	—
Dense (Out)	$u = 10$	Softmax

C. Hyperparameter Selection

The CNN uses L2 regularization (0.005) on convolutional and dense layers to prevent overfitting, and a dropout layer (rate = 0.5) after the dense layer to promote generalization. The model is compiled with Adam optimizer (learning rate = 10^{-5}, clipnorm = 1.0) and a custom loss function:

$$\mathcal{L} = \text{CE}(y, \hat{y}) + 1.0 \cdot H(\hat{y})$$

where CE is categorical cross-entropy and $H(\hat{y}) = -\sum \hat{y}_i \log \hat{y}_i$ is Shannon entropy, encouraging confident predictions.

The training set was balanced across the 10 selected known classes. The *Bot* class, whose traffic patterns closely resemble DoS attacks, is included in the known set to ensure balanced learning.

TABLE III
DATASET SPLIT AND CLASS BALANCE

Set	**Total**	**Known / Unknown**
Validation	1,250	625 / 625
Test	11,269	9,667 / 1,602

D. Algorithmic Design: Confidence-Aware DQN

We split the dataset into a balanced validation set (10% of total samples) and a test set. The validation set contains 1,250 samples (625 known and 625 unknown), while the test set contains 11,269 samples (9,667 known and 1,602 unknown), as shown in Table III. A validation set was used to calibrate the DQN model for the centroid-guided reward, ensuring that low-confidence and high-confidence samples could be effectively separated. We use cosine similarity rather than Gaussian distance because the three confidence features $P1$, $(P1-P2)$, and (H) entropy, operate on different numerical scales and magnitudes. Gaussian or Euclidean metrics would impose a magnitude bias, causing features with larger numeric ranges to dominate the distance function. Cosine similarity is the perfect option that we have to avoid this issue by comparing only the direction of the confidence vectors, making it well-suited for a mixed-scale 3D state space. We set a similarity threshold of 0.75 for updating the centroids, preventing high-confidence states from being distorted by low-confidence samples. This constraint stabilizes learning and preserves a clean separation between the evolving representations of known and unknown traffic.

From validation, we select top/bottom 5% by $P1$ as initial centroids c_k (known) and c_u (unknown), yielding 62 high-confidence and 62 low-confidence anchor samples.

The DQN state is $s = (P1, P1 - P2, H)$. Actions are $\{0 = \text{unknown}, 1 = \text{known}\}$.

1) Unsupervised Reward: Cosine similarity:

$$\text{sim}_k = \frac{s \cdot c_k}{\|s\|\|c_k\|}, \quad \text{sim}_u = \frac{s \cdot c_u}{\|s\|\|c_u\|}$$

Reward:

$$r = \begin{cases} \max(\text{sim}_k, \text{sim}_u) & \text{if } a = \arg\max(\text{sim}_k, \text{sim}_u) \\ -\max(\text{sim}_k, \text{sim}_u) & \text{otherwise} \end{cases}$$

$$r \in [-1, 1]$$

Centroids update only if similarity > 0.75:

$$c \leftarrow \frac{c \cdot n + s}{n + 1}$$

Algorithm 1 DQN Training for Open-Set Detection

Initialize $Q(s, a; \theta) : 3 \to 64 \to 64 \to 2$
Initialize replay $\mathcal{D}$, centroids c_k, c_u from top/bottom 5% $P1$
for episode = 1 to 30 **do**
  for each sample in 1,250 subsample **do**
    $s = (P1, P1 - P2, H)$
    $a \sim \epsilon$-greedy$(Q(s))$
    $\text{sim}_k = \cos(s, c_k)$, $\text{sim}_u = \cos(s, c_u)$
    $r = \max(\text{sim}_k, \text{sim}_u)$ if correct else $-\max$
    Store $(s, a, r, s', \text{done})$ in $\mathcal{D}$
    if sim > 0.75 **then** update centroid $c \leftarrow \frac{c \cdot n + s}{n+1}$
    end if
    Sample minibatch (batch=32), update Q via TD error
  end for
  Decay ϵ to min 0.05
end for

2) DQN Training: We subsample 1,250 validation samples (excluding anchors) and select the top and bottom 5% based on confidence metrics, resulting in 62 high-confidence and 62 low-confidence samples used for DQN training. Training is performed for 30 episodes using experience replay (batch=32), ϵ-greedy exploration (decaying from 1.0 to 0.05), discount γ=0.95, and the Adam optimizer.

3) Why DQN (off-policy)?: We chose DQN over actor-critic or policy gradients because:

- **Sample efficiency**: Experience replay stabilizes learning in a low-dimensional state space (3D).
- **Discrete actions**: Binary decision (known/unknown) fits Q-learning perfectly.
- **Proven stability**: Actor-critic methods suffer high variance in sparse-reward settings; DQN converges faster with ϵ-greedy exploration.

IV. Experiments

A. Confidence Metric Distributions

The purpose of analyzing the distributions of the three softmax-derived confidence metrics is to verify whether known and unknown traffic naturally separate in feature space before any reinforcement learning is applied. Since our DQN operates purely on these confidence features and not on raw logits or class labels examining their statistical behavior is essential for confirming that they provide a meaningful signal for distinguishing low-confidence (unknown) from high-confidence (known) events. If the metrics exhibit clear separability, the DQN can exploit this structure during training without the need for supervision, manual thresholds, or human feedback. Thus, this analysis validates that the inputs to the RL agent contain sufficient discriminative information for open-set intrusion detection.

We first analyze the three confidence metrics extracted from the CNN softmax outputs on the test set: maximum probability ($P1$), probability gap ($P1 - P2$), and Shannon entropy (H). The test set contains 10,292 known and 2,227 unknown samples. As shown in Fig. 2, known samples exhibit high $P1$ values (0.9860 ± 0.0663), while unknown samples are significantly lower (0.6697 ± 0.1280). Fig. 3 demonstrates that $P1 - P2$ is large for known samples (mean = 0.9755, std = 0.1142) and near zero for unknowns (mean = 0.5072, std = 0.1890).

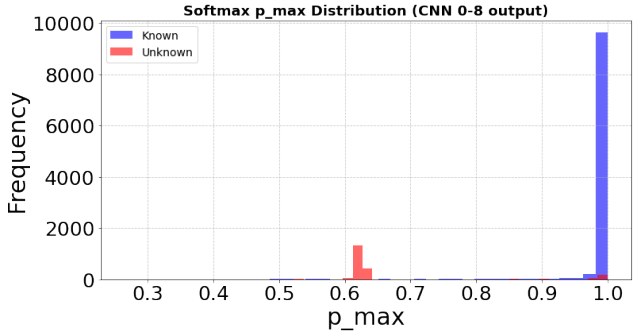

Fig. 2. $P1$: Maximum class probability of samples

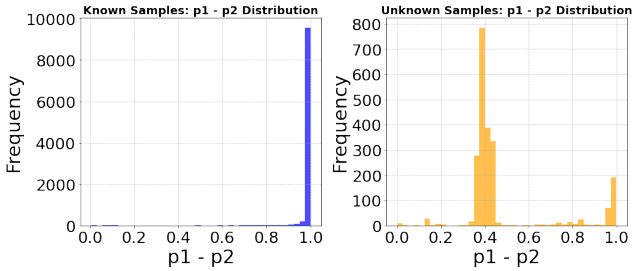

Fig. 3. $(P1 - P2)$: Distribution of known and unknown samples

These distributions validate our hypothesis that unknown traffic produces low-confidence, high-entropy predictions, providing a strong signal for open-set detection.

B. DQN-IDS Performance

The goal of this section is to demonstrate how effectively the proposed DQN-based intrusion detection system converts softmax derived confidence features into accurate open-set decisions. Since The DQN operates without access to labels, logits, or explicit thresholds, its performance must empirically

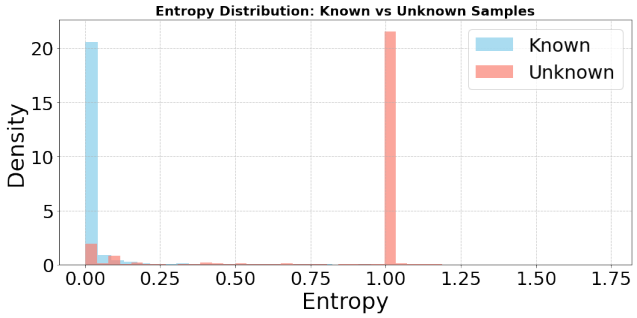

Fig. 4. H: Shannon entropy of the softmax distribution

validate that reinforcement learning can reliably distinguish known high-confidence traffic from unknown low-confidence traffic in a fully unsupervised manner. The presentation of these results confirms that the learned policy captures the underlying structure in the confidence space and generalizes well to unseen attacks. The full CNN+DQN pipeline is evaluated in binary open-set mode (known vs. unknown). The DQN learns to classify samples based on the 3D confidence state. Results are shown in Table IV. DQN-IDS achieves a binary F1-score of 97.83%, correctly identifying 9,465 out of 9,667 known samples and 1,384 out of 1,602 unknown samples.

This demonstrates robust separation even under class imbalance, with the DQN effectively leveraging the uncertainty signals from the CNN to maintain high performance on both known and unknown classes.

TABLE IV
DQN-IDS PERFORMANCE ON TEST SET

Metric	Value
Binary Accuracy	96.27%
Binary F1-score	97.83%
Correct Known	9,465 / 9,667
Correct Unknown	1,384 / 1,602

C. Ablation Study

We ablate the DQN module in two ways: (1) replacing it with fixed thresholds on individual confidence metrics, and (2) varying the number of withheld unknown classes.

1) DQN vs. Fixed Thresholds: We compare DQN with static thresholds on $P1$, $P1 - P2$, and H. Results are shown in Table V. DQN significantly outperforms all thresholds, achieving 97.83% F1. Threshold-based methods suffer from poor adaptability, especially on minority classes, with F1 scores dropping below 92%. This highlights the advantage of DQN's learned decision boundary over rigid rules.

TABLE V
PERFORMANCE COMPARISON ON KNOWN AND UNKNOWN ATTACK DETECTION

Metric	Ours	VarMax
Dataset	CICIDS 2017	CICIDS, UNSW
Test Size	11,269	∼12k–15k
F1 Score (Known)	98.05%	∼74%
F1 Score (Unknown)	86.39%	∼71%
F1 Score (Binary Total)	97.83%	∼72–74%
Zero-Day Evaluation	Yes	Yes

D. Runtime Summary

To address concerns about the two-stage design, we evaluated the runtime of both components and clarified their roles during deployment. Each incoming flow is first processed by CNN to produce confidence metrics P1, P1-P2, and H. Only these three scalars are forwarded to the DQN, meaning that CNN sets the upper bound on runtime while the DQN adds a negligible decision step. CNN achieves an inference time of 0.5215 ms per sample (1.9k samples/s), which comfortably meets the throughput requirements of IDS deployments. The DQN, which operates on only three scalar inputs and is implemented as a lightweight fully connected network, runs at 0.0241 ms per sample (41.5k samples/s). Its forward pass is over 20 times faster than the CNN and is effectively cost-free relative to the pipeline. This makes the two-stage architecture as practical as one-stage close-set CNN IDS models with added advantages. If we Compare it to other open-set IDS like threshold-based systems (e.g., varMax), it removes manual tuning overhead; compared to unsupervised GAN/OpenMax pipelines, it avoids retraining and heavy parameterization; and unlike prior RL-based IDS, it requires no labels or human-in-the-loop feedback. The result is a more flexible, scalable and low-latency alternative for real-world intrusion detection where both accuracy and runtime constraints matter. A deployable 2-stage Zero-day detection IDS.

TABLE VI
INFERENCE RUNTIME COMPARISON OF CNN CLASSIFIER AND DQN CONFIDENCE MODULE

Module	Inf. per Sample (ms)	Throughput (samples/s)
CNN Classifier	0.5215	1,917
DQN Confidence Module	0.0241	41,500

E. Generalization to another dataset

We evaluated our approach on the UNSW-NB15 dataset [11], which is more volatile comapred to CICIDS-2017. Two experimental strategies were explored: (1) treating selected UNSW attack classes as zero-day inputs for our CICIDS-trained IDS, and (2) applying the entire CNN+DQN framework directly on the UNSW dataset to assess its standalone performance.

1) Unique classes in UNSW as zero-day attack for CICIDS-Based IDS: .

To evaluate the generalization capability of our CNN+DQN framework beyond CICIDS-2017, we first identified 15 common features shared between the CICIDS-2017 and UNSW-NB15 datasets. These features capture fundamental flow-level properties such as packet counts (forward/backward), packet lengths, inter-arrival time statistics, and flow throughput making them consistent indicators of traffic behavior across datasets. Because they represent low-level transport and timing characteristics rather than dataset-specific metadata, they provide a reliable basis for cross-dataset inference. CNN was trained on CICIDS-2017 using only these 15 features to ensure full compatibility with UNSW-NB15 during evaluation.

To assess the model's ability to detect previously unseen attacks, we conducted experiments on a subset of the UNSW-NB15 dataset. In this evaluation, four attack categories *Fuzzers*, *Backdoor*, *Shellcode*, and *Worms* were withheld from training and relabeled as *Unknown* to simulate zero-day threats. These categories were selected because their traffic signatures differ markedly from the known classes in CICIDS-2017, categories include: *Backdoor* traffic often involves stealthy periodic beaconing, *Shellcode* produces short

exploit triggered bursts, *Worms* generate self-propagating scan patterns, and *Fuzzers* exhibit highly variable probing behavior.

The combined evaluation dataset consists of 12,903 samples with 15 features, including 10,292 known and 1,611 unknown instances, as summarized in Table VIII. The per-class distribution, including the aggregated unknown category, is shown in Table IX. The confidence-based prediction performance of the DQN-IDS on the UNSW test set is reported in Table VII, demonstrating strong cross-dataset robustness with an overall accuracy of 92.07

TABLE VII
PERFORMANCE ON UNSW DATASET AS UNKNOWN

Metric	Value
Total Samples	10,713
Known Samples	9,697
Unknown Samples	1,016
Overall Accuracy	92.07%
Overall F1 Score	95.46%
Correct Known Predictions	8,928 / 9,697
Correct Unknown Predictions	935 / 1,016

TABLE VIII
COMBINED DATASET SUMMARY

Metric	Value
Total Samples	12,903
Known Samples	10,292
Unknown Samples (UNSW)	1,611
Number of Features	15

TABLE IX
LABEL COUNTS IN COMBINED CICIDS + UNSW DATASET

Label	Count
BENIGN	5,499
DoS Hulk	5,499
PortScan	5,499
DDoS	5,499
DoS GoldenEye	5,499
FTP-Patator	5,499
SSH-Patator	5,499
DoS slowloris	5,499
DoS Slowhttptest	5,499
Bot	1,966
Unknown (UNSW)	1,611

2) UNSW Based DQN-IDS Results: .

To further examine the flexibility of our framework, We applied our complete CNN+DQN intrusion detection framework to the UNSW-NB15 dataset to assess its adaptability and generalization beyond CICIDS-2017. After tuning the hyperparameters across multiple configurations, the F1-scores consistently fell between 0.75 and 0.85. The representative configuration reported here illustrates that the model converges reliably and remains stable under multiple training setups. The performance advantage primarily stems from the DQN's ability to learn a non-linear, adaptive decision boundary in the 3-D confidence space, whereas fixed thresholds impose rigid, axis-aligned separations that fail to capture the curved geometry between known and unknown traffic. Our lightweight DQN adds negligible inference overhead (<1 ms, <100 KB) and requires no additional labels or threshold adjustments, making it both practical and more effective than static baselines. Unlike the cross-dataset experiment (CICIDS $\rightarrow$ UNSW), this setup uses UNSW-NB15 exclusively for both training and testing, providing a direct assessment on a more heterogeneous dataset. Six classes: Normal, Fuzzers, Generic, Exploits, DoS, and Reconnaissance were treated as known categories. Their training distribution was balanced through equal sampling, yielding 5,999 samples per class (with DoS and Reconnaissance naturally smaller after filtering). Additionally, 1,682 samples from the remaining attack types were grouped into a single Unknown class to simulate zero-day behavior. The balanced known-class counts used for training are shown in Table X.

The CNN was trained on 41 statistical flow features reshaped for convolutional processing. The final dataset consisted of 25,264 training samples and 7,999 testing samples, with all NaN or infinite values removed. The resulting confidence distributions showed clear separation: known samples exhibited consistently higher softmax confidence, while unknown traffic showed lower (P max) and reduced (P1-P2) aligning with typical out-of-distribution characteristics.

Performance on the UNSW test set demonstrates that our DQN-IDS successfully captures these patterns. The CNN alone achieved 80.78% accuracy and 80.66% F1 on the known classes. In contrast, the DQN-based classifier, operating purely on confidence features, reached 74.89% overall accuracy and 82.47% F1, correctly identifying 1,140 out of 1,283 unknown samples. These test-set results are presented in Table XI.

Overall, UNSW-NB15 remains challenging due to its high variability and noise, which often stress test statistical feature based IDS models. Despite this, our CNN+DQN framework adapts effectively, learns meaningful decision boundaries, and demonstrates strong capability to detect zero-day traffic patterns.

TABLE X
KNOWN-CLASS AND UNKNOWN COMBINED SET COUNTS FOR UNSW-BASED DQN-IDS

Label	Count
Normal	5,999
Fuzzers	5,999
Generic	5,999
Exploits	5,999
DoS	4,089
Reconnaissance	3,496
Unknown	1,682

V. CONCLUSION

Our results highlight the effectiveness of combining CNN-based feature extraction with DQN-driven confidence analysis for open-set intrusion detection. The CNN model achieves high accuracy and confidence in known traffic, while the integration of a DQN trained in softmax-derived uncertainty features enables the hybrid system to generalize effectively to

TABLE XI
UNSW-ONLY CNN+DQN TEST SET CONFIDENCE-BASED PERFORMANCE

Metric	Value
Total Samples	7,201
Known Samples	5,918
Unknown Samples	1,283
Overall Accuracy	74.89%
Overall F1 Score	82.47%
Correct Known Predictions	4,253 / 5,918
Correct Unknown Predictions	1,140 / 1,283

previously unseen traffic. On a test set of 10,016 samples held-out, the CNN+DQN pipeline achieved an overall precision of 96.27% and a binary F1-score of 97.83%. The system correctly identified 9,465 out of 9,667 known samples and 1,384 out of 1,602 unknown samples. Most known attack classes were recognized with 98% accuracy. In contrast, benign traffic exhibited lower accuracy (77.51%), which may be attributed to similarities between certain benign patterns and low-confidence attack traffic, leading to conservative unknown classifications. By learning to distinguish known and unknown traffic without relying on hard-coded thresholds, the DQN introduces adaptability and robustness into the decision process. This flexibility is particularly valuable in real-world environments where network conditions and attack behaviors evolve over time. In general, the proposed framework based on confidence-driven reinforcement learning represents a practical step towards more resilient and deployable intrusion detection systems capable of addressing zero-day threats.

TABLE XII
CNN + DQN ACCURACY ON KNOWN TEST SAMPLES CICIDS-2017

Class ID	Samples	Correct	Accuracy (%)
0	867	672	77.51
1	882	880	99.77
2	880	864	98.18
3	898	882	98.22
4	868	846	97.47
5	884	870	98.42
6	876	866	98.86
7	880	872	99.09
8	871	859	98.62
9	328	326	99.39

A. Possible Limitations

The proposed framework relies on the quality of the confidence estimates produced by the underlying CNN. Although unknown or zero-day samples generally exhibit lower confidence or higher uncertainty, deep neural networks may occasionally assign high confidence to unseen inputs. In such cases, the ability of the DQN to distinguish unknown traffic can be reduced, reflecting a known limitation of confidence-based detection approaches.

Additionally, the tradeoff between detecting unknown attacks and maintaining high accuracy in benign traffic may lead to increased false positives in certain scenarios. In practical deployments, this can be mitigated by prioritizing alerts, secondary verification mechanisms. Despite these limitations, the proposed approach provides a flexible and threshold-free mechanism for open-set intrusion detection.

B. Future Work

The proposed framework can be extended in several promising directions. First, evaluating the CNN-DQN pipeline across diverse intrusion detection datasets would help assess its generalizability. Further, analyzing and categorizing unknown traffic into groups of similar values could reveal whether it represents new type novel attacks or benign variations, enabling dynamic updates to the model.

To enhance the decision making process, the DQN component could work with richer input features such as raw CNN logits or components based on logits like energy-based confidence scores, offering a more nuanced understanding of uncertainty than softmax alone. Lastly, real-time deployment should be explored by integrating the system with live traffic monitoring tools, to detect the traffic in live environment.

ACKNOWLEDGMENT

This work was supported in part by the U.S. Military Academy (USMA) under Cooperative Agreement No. W911NF-22-2-0160 and in part by UMass Dartmouth's Marine and Undersea Technology (MUST) Research Program funded by the Office of Naval Research (ONR) under Grant No. N00014-23-1-2141. The views and conclusions expressed in this paper are those of the authors and do not reflect the official policy or position of the U.S. Military Academy, University of Massachusetts Dartmouth, the Office of Naval Research, U.S. Army, U.S. Navy, U.S. Department of Defense, or U.S. Government.

REFERENCES

[1] G. Baye, P. Silva, A. Broggi, N. D. Bastian, L. Fiondella, and G. Kul, "varmax: Towards confidence-based zero-day attack recognition," in *MILCOM 2024 - 2024 IEEE Military Communications Conference (MILCOM)*, pp. 863–868, IEEE, 2024.

[2] P. Silva, G. Baye, N. Costagliola, N. D. Bastian, G. Kul, and L. Fiondella, "A repair-time trigger for cyberattack classifiers," in *MILCOM 2025 - 2025 IEEE Military Communications Conference (MILCOM)*, pp. 1–6, 2025.

[3] A. Bendale and T. E. Boult, "Towards open set deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1563–1572, IEEE, 2016.

[4] W. J. Scheirer, A. d. R. Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1757–1772, 2012.

[5] Y. Wang, B. Li, T. Che, K. Zhou, D. Li, and Z. Liu, "Energy-based open-world uncertainty modeling for confidence calibration," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9302–9311, IEEE, 2021.

[6] G. Ekinci, A. Broggi, L. Fiondella, N. D. Bastian, and G. Kul, "Adaptive network intrusion detection systems against performance degradation via model agnostic meta-learning," in *Proceedings of the 11th ACM Workshop on Adaptive and Autonomous Cyber Defense*, AACD '24, (New York, NY, USA), p. 23–26, Association for Computing Machinery, 2024.

[7] N. Costagliola, G. Ekinci, N. D. Bastian, L. Fiondella, and G. Kul, "Replay or regret: Evaluating continual learning methods for robust intrusion detection," in *MILCOM 2025 - 2025 IEEE Military Communications Conference (MILCOM)*, pp. 1–6, 2025.

[8] A. Broggi, G. Baye, P. Silva, N. Costagliola, N. Bastian, L. Fiondella, and G. Kul, "varmax: Uncertainty and novelty management in deep neural networks," in *Hawaii International Conference on System Sciences (HICSS)*, 2025.

[9] Y. Fang and X. Xie, "Unknown intrusion traffic detection method based on unsupervised learning and open-set recognition," *Sci. Rep.*, vol. 15, no. 1084, 2025.

[10] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *2018 International Conference on Communications and Network Security (CNS)*, pp. 1–6, IEEE, 2018.

[11] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems (UNSW-NB15 Network Data Set)," in *Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS)*, pp. 1–6, IEEE, 2015.

[12] S. H. Alavizadeh, M. H. Anisi, A. H. Abdullah, and A. Gani, "A deep reinforcement learning approach for intrusion detection in software defined networks," *Computers & Security*, vol. 102, p. 102118, 2021.

[13] A. Puzanov and K. Cohen, "Deep reinforcement one-shot learning for change point detection," in *Proceedings of the 56th Annual Allerton Conference on Communication, Control, and Computing*, 2018.

[14] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp. 1–6, IEEE, 2009.