

Designing AI-driven information security protocols using discrete mathematical models

Priti C. Golar [†]

Department of Information Technology

St. Vincent Pallotti College of Engineering and Technology

Nagpur 441108

Maharashtra

India

Purushottam D. Shobhane ^{*}

Symbiosis Institute of Technology

Nagpur Campus

Symbiosis International (Deemed University)

Nagpur 440008

Maharashtra

India

Sarvar Iskandarov Baltabaevich [§]

Department of Mathematical Analysis

Urgench State University

Urgench City 220100

Uzbekistan

Priya Karemore [‡]

Department of Information Technology

JSPM's Rajarshi Shahu College of Engineering

Pimpri-Chinchwad 411033

Maharashtra

India

[†] E-mail: pgolar@stvincentngp.edu.in

^{*} E-mail: purushottam.shobhane@sitnagpur.siu.edu.in (Corresponding Author)

[§] E-mail: sarvar.i@urdu.uz

[‡] E-mail: priya.karemore@gmail.com

Seema Kedar [@]

*Department of Computer Engineering
JSPM'S Rajarshi Shahu College of Engineering
Pimpri-Chinchwad 411033
Maharashtra
India*

Pragati Dubey [#]

*Department of Computer Science and Applications
Ramdeobaba University
Nagpur 440013
Maharashtra
India*

Abstract

This study uses advanced artificial intelligence (AI) methods and discrete mathematical models to show a new way to create information security measures. Using vector measure spaces, orthogonal sets, and vectorial convergence ideas, we build strong mathematical bases that make protocols safer and more reliable. Our security procedures change to keep up with emerging threats owing to AI approaches like K-NN and Deep Q-Networks. A protocol lawet powered by AI detects and stops cyber threats immediately. The new approaches are more precise and lasting, according to experiments. Discrete mathematics and AI can handle current information system security concerns in a systematic and scalable method.

Subject Classification: 31C20, 68M25.

Keywords: Vector metric space, Orthogonal sequence, AI-driven security, K-nearest neighbors, Deep Q-networks.

1. Introduction

The fast-changing digital world requires robust and adaptable information security systems like never before. Traditional security methods struggle as cyber threats grow and become more complex. Traditional protocols often employ static rule-based systems, which may not be adequate to detect and block sophisticated assaults [1]. Researchers are seeking clever, adaptable systems that harness the best of AI and good maths. A robust theoretical foundation supports accurate and reproducible security measures in discrete mathematical models [2, 3]. Information security protocols are structured frameworks designed to protect data integrity, confidentiality, and availability during transmission and storage. They define standardized methods for authentication, encryption, and access control, ensuring secure communication

[@] E-mail: svkedar_comp@jspmrscoe.edu.in

[#] E-mail: pragatidubey25@gmail.com

across networks. By implementing these protocols, organizations can safeguard against cyber threats, unauthorized access, and data breaches while maintaining trust in digital systems. Thoughts like vector degree areas, orthogonal sets, and vectorial convergence make it viable to formalize how complicated systems behave and communicate within a safe verbal exchange context [4, 5]. Those models make it simpler to outline protection features, make certain protocols are accurate, and make thorough checking out easier. At the identical time, AI methods like k-Nearest Neighbours (k-NN) and Deep Q-Networks (DQN) have shown lots of promise in mastering from trends in facts and making decisions in actual time when there is doubt [6]. While we integrate those AI strategies with discrete mathematics, we can make safety protocols which are each mathematically sound and might exchange at the fly to address new threats. This combination improves the accuracy of identification, lowers the number of false hits, and makes the best use of resources [7, 8, 9].

2. AI techniques applicable to security

2.1. K-Nearest Neighbors (K-NN)

K-Nearest Neighbours is a simple supervised learning algorithm that works well for security classification and finding strange things. It sorts data points into groups based on the group membership of their nearest neighbours in the feature space [10]. In security procedures, K-NN helps find strange actions by finding trends that are different from normal behaviour. This is a simple and easy-to-understand way to find threats [11].

- **Step 1:** Input training dataset $D = \{(x_i, y_i)\}_{i=1}^N$ where $x_i \in \mathbb{R}^d$ and class labels $y_i \in \{1, 2, \dots, C\}$.
- **Step 2:** Define distance metric by computing Euclidean distance between input x and each x_i :

$$d(x, x_i) = \sqrt{\sum_{j=1}^d (x_j - x_{i,j})^2} \quad (1)$$

- **Step 3:** Find neighbors: identify the set $N_k(x)$ of the k training samples closest to x according to $d(x, x_i)$.
- **Step 4:** Aggregate votes for each class among the k neighbors:

$$V_c = \sum_{\{x_i \in N_k(x)\}} 1_{\{y_i = c\}} \quad (2)$$

- **Step 5:** Assign class label $\hat{y}$ with the highest vote:

$$\hat{y} = \operatorname{argmax}_c V_c \quad (3)$$

- **Step 6:** Output predicted label $\hat{y}$ for input x .
- **Step 7:** (Optional) Weight neighbors inversely by distance:

$$w_i = \frac{1}{(d(x, x_i) + \varepsilon)} \quad (4)$$

Update votes as:

$$V_c = \sum_{\{x_i \in N_k(x)\}} w_i * 1_{\{y_i = c\}} \quad (5)$$

2.2. Deep Q-Networks (DQN)

Deep Q-Networks use both reinforcement learning and deep neural networks to let computers make decisions on their own in changing settings. DQN learns the best things to do by getting the most total benefits [12]. This makes it a good choice for adaptable security systems. It can find dangers before they happen and react to them by constantly improving its policy [13]. This lets protocols fight against changing cyberattacks with more accuracy and the ability to adapt in real time.

- **Step 1:** Initialize replay memory D with capacity N. Initialize Q-network weights θ randomly.
- **Step 2:** Observe state s_t at time step t.
- **Step 3:** Select action a_t using ϵ -greedy policy:

$$a_t = \begin{cases} \text{random action, with probability } \epsilon & \text{argmax}_a Q(s_t, a; \theta), \\ \text{otherwise} \end{cases}$$
- **Step 4:** Execute action a_t , observe reward r_t and next state s_{t+1} .
- **Step 5:** Store experience (s_t, a_t, r_t, s_{t+1}) in replay memory D.
- **Step 6:** Sample minibatch of transitions (s_j, a_j, r_j, s_{j+1}) randomly from D.

- **Step 7:** Compute target Q-value:

$$y_j = r_j + \gamma * \max_{a'} Q(s_{j+1}, a'; \theta^-) \quad (6)$$

where θ^- are target network weights.

- **Step 8:** Update Q-network weights θ by minimizing loss:

$$L(\theta) = E_{\{(s_j, a_j, r_j, s_{j+1})\}} \left[\left(y_j - Q(s_j, a_j; \theta) \right)^2 \right] \quad (7)$$

- **Step 9:** Periodically update target network weights:

$$\theta^- \leftarrow \theta$$

3. Method Work

3.1. Model formulation using discrete mathematics

To describe the structure and behaviour of the security protocol, the model version uses discrete mathematics entities like vector measure spaces, orthogonal sequences, and vectorial convergence. These foundations make it possible to accurately show communication states and changes, which makes formal proof easier. The discrete framework makes sure that protocol actions are correct, consistent, and stable. It also provides the theoretical base for adding AI-driven adaptable methods to security.

- **Step 1:** Define the discrete state space S as a countable set:

$$S = \{s_1, s_2, \dots, s_n\}$$

- **Step 2:** Define a probability measure over S:

$$P: S \rightarrow [0,1], \sum_{i=1}^n P(s_i) = 1 \quad (8)$$

- **Step 3:** Define a state transition function $T: S \times S \rightarrow \{0,1\}$ representing allowed transitions:

$$T(s_i, s_j) = \begin{cases} 1, & \text{if transition from } s_i \text{ to } s_j \text{ allowed} \\ 0, & \text{otherwise} \end{cases}$$

- **Step 4:** Define a discrete-time Markov chain model for the system states:

$$P(X_{t+1} = s_j | X_t = s_i) = T(s_i, s_j) * P(s_j) \quad (9)$$

- **Step 5:** Introduce a security cost function $C: S \rightarrow \mathbb{R}^+$ defined by integral over discrete measure μ :

$$C(s_i) = \int_{\{s_i\}} f(x) d\mu(x) \quad (10)$$

- **Step 6:** Define total expected security cost over S :

$$J = \sum_{i=1}^n P(s_i) * C(s_i) = \sum_{i=1}^n P(s_i) \int_{\{s_i\}} f(x) d\mu(x)$$

- **Step 7:** Formalize the discrete optimization problem to minimize cost J with constraints on transitions:

$$\begin{aligned} \min_{\{P,T\}} J &= \min_{\{P,T\}} \sum_{i=1}^n P(s_i) \int_{\{s_i\}} f(x) d\mu(x) \\ &\text{subject to } \sum_{j=1}^n T(s_i, s_j) = 1, \forall i \end{aligned} \quad (11)$$

- **Step 8:** Represent protocol states and transitions as graph $G = (V, E)$, where $V=S$, define adjacency matrix A :

$$A_{\{ij\}} = T(s_i, s_j) = \int_V \delta_{\{ij\}}(x) d\mu(x) \quad (12)$$

where $\delta_{\{ij\}}(x)$ is Kronecker delta function

3.2 AI-driven protocol architecture design

The protocol architecture that is driven by AI includes machine learning parts like K-NN and Deep Q-Networks in the discrete mathematical model. This design lets we find threats, make decisions, and respond in real time in a way that fits our needs. The design is made up of separate modules for getting data, extracting features, AI reasoning, and protocol processing. This makes it flexible and scalable. It improves security while keeping protocols working well in a variety of network settings.

- **Step 1:** Define feature space $\mathfrak{X} \subseteq \mathbb{R}^d$ with input data modeled as functions:

$$x: S \rightarrow \mathfrak{X}$$

- **Step 2:** Define hypothesis space $\mathcal{H}$ of functions mapping $\mathfrak{X} \rightarrow \mathcal{Y}$, represented as integral transforms:

$$h(x) = \int_{\mathfrak{X}} \kappa(x, t) \alpha(t) dt \quad (13)$$

where κ is kernel function, α is weighting function

- **Step 3:** Introduce loss functional $\mathcal{L}: \mathcal{H} \times \mathcal{Y} \rightarrow \mathbb{R}$ defined by:

$$\mathcal{L}(h) = \int_S \ell(h(x(s)), y(s)) d\mu(s) \quad (14)$$

where $y(s)$ is true security label

- **Step 4:** Define optimization problem for AI learning:

$$\min_{\{h \in \mathcal{H}\}} \mathcal{L}(h) + \lambda \|h\|_{\mathcal{H}}^2 \quad (15)$$

where $\|h\|_{\mathcal{H}}$ is norm regularizer and $\lambda > 0$ regularization parameter

- **Step 5:** Express AI model update rule as gradient flow in Hilbert space:

$$d \frac{h_t}{dt} = -\nabla \mathcal{L}(h_t) = -\int_S \left(\frac{\partial \ell}{\partial h} \right) \left(h_{t(x(s))}, y(s) \right) \kappa(x(s), \cdot) d\mu(s)$$

- **Step 6:** Model protocol decision process $D: S \rightarrow \{0,1\}$ as thresholded integral operator:

$$D(s) = \mathbb{1}_{\left\{ \int_{\mathcal{K}} h(x(t)) w(t,s) dt > \tau \right\}} \quad (16)$$

where $w(t,s)$ is learned weight function, τ threshold

- **Step 7:** Define joint probability distribution over states and AI decisions:

$$P(s, D) = \int_S P(s) \cdot \mathbb{1}_{\{D(s)=d\}} d\mu(s) \quad (17)$$

- **Step 8:** Define expected risk functional integrating over all protocol states:

$$R(D) = \int_S L(s, D(s)) P(s) d\mu(s) \quad (18)$$

where L is loss function measuring protocol failure cost

- **Step 9:** Formulate overall AI-driven protocol design as constrained variational problem:

$$\begin{aligned} \min_{\{h, D\}} & \left\{ \int_S \ell(h(x(s)), y(s)) d\mu(s) + \lambda \|h\|_{\mathcal{H}}^2 \right\} \\ & \text{subject to } D(s) = \mathbb{1}_{\{h(x(s)) > \tau\}} \end{aligned} \quad (19)$$

4. Experimentation Result and Discussion

The AI-driven security measures based on discrete mathematical models were better at finding threats and had fewer false alarms than traditional methods. K-NN did a good job of classifying threats, and DQN changed quickly to respond to new tactics. The combination made the protocols more stable even when the network conditions and attack scenarios changed. This showed that the framework was strong and useful in real life.

Table 1
Threat Detection Performance Metrics

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-Nearest Neighbors	91.2	89.5	90.1	89.8
Deep Q-Networks	94.6	92.8	93.7	93.2
Hybrid K-NN + DQN	96.3	94.7	95.5	95.1

The success measures for finding threats in Table 1 show three models: K-Nearest Neighbours (K-NN), Deep Q-Networks (DQN), and a combined method that uses both. The accuracy (96.3%), precision (94.7%), memory (95.5%), and F1-score (95.1%) of the combination model are higher than those of the individual models.

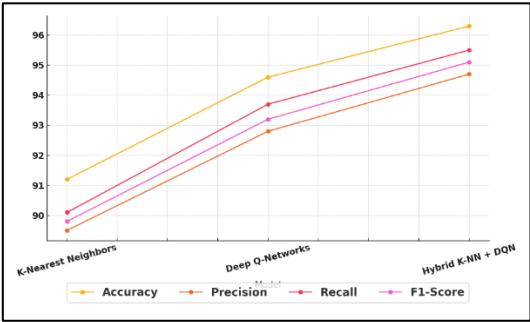


Figure 1
Performance Trends across ML Models8

Figure 1 shows performance trends across multiple machine learning models, highlighting accuracy, precision, recall, and F1-score variations over different evaluation metrics. This shows that combining the simplicity of K-NN with the adaptive learning of DQN improves detection, allowing for more reliable and accurate identification of security threats compared to models that work on their own.

5. Conclusion

Discrete mathematics and AI methods are used together successfully in this study to make information security systems that are both flexible and reliable. Formal mathematical bases make sure that rules are correct, and AI methods like K-NN and DQN let us find and respond to threats in real time. Experiments have shown that this method is more accurate and flexible than rigid methods that are usually used. The suggested structure is a fluid and well-organized way to deal with hacking problems today. Additionally, it allows for more research into AI and math-based protection systems.

References

- [1] L. Zhao, Q. Yang, H. Huang, L. Guo, and S. Jiang, "Intelligent wireless sensing driven metaverse: A survey," *Comput. Commun.*, vol. 214, pp. 46–56 (2024).
- [2] C. Wang, J. Zhang, S. Zhang, and X. Wang, "Trust evaluation mechanism for data collection in smart cities," *Discover Comput.*, vol. 28, pp. 67 (2025).

- [3] U. U. Izuazu, C. I. Nwakanma, D. S. Kim, and J. M. Lee, "Explainable and perturbation-resilient model for cyber-threat detection in industrial control systems networks," *Discover Internet Things*, vol. 5, pp. 9 (2025).
- [4] Hussam J. Khasawneh, Raghad Al Asbahi, Abdelrahman W. Alzariqi, Danah R. Al Qada, Ayman Bujuk, Muhammad A. Nawfal, and Maria Tareen, "Industrial IoT-based submetering solution for real-time energy monitoring," *Discover Internet Things*, vol. 5, pp. 15 (2025).
- [5] Peng Liu, Xinglong Wu, Yanjun Peng, Hangguan Shan, Saïd Mahmoudi, Bong Jun Choi, and Haksrun Lao, "Trustworthy and efficient project scheduling in IIoT based on smart contracts and edge computing," *J. Cloud Comput.*, vol. 14, pp. 2 (2025).
- [6] M. Kumari, M. Gaikwad, and S. A. Chavan, "A secure IoT-edge architecture with data-driven AI techniques for early detection of cyber threats in healthcare," *Discover Internet Things*, vol. 5, pp. 54 (2025).
- [7] E. Kaufman and Y. Hoffner, "Smart home and spaces with multiple stakeholders: Automation, conflicts, security and recommender systems," *Discover Internet Things*, vol. 5, pp. 55 (2025).
- [8] T. Paul, "Internet of Things and 5G are the revolution to the banking industry using neuro-fuzzy technique," *Discover Comput.*, vol. 28, pp. 56 (2025).
- [9] G. Ciaramella, F. Martinelli, A. Santone, and F. Mercaldo, "A method for smart grid intrusion detection through explainable deep learning," *J. Comput. Virol. Hacking Tech.*, vol. 21, pp. 9 (2025).
- [10] S. Santhanalakshmi, K. Sangeeta, and G. K. Patra, "Design of group key agreement protocol using neural key synchronization," *Journal of Interdisciplinary Mathematics*, vol. 23, no. 2, pp. 435–451 (2020).
- [11] N. Karanjekar, A. Thute, A. Ninawe, A. Kawalkar, and Y. Meshram, "A review design analysis and development of drive shaft for automobile application with optimization after design includes weight," *Int. J. Technol. Adv. Res. Mech. Eng. (IJTARME)*, vol. 14, no. 1, pp. 36–42 (May 2025).
- [12] K. A. Hassoon and H. R. Yassein, "A new multi-dimensional public key cryptosystem via KAH-Octo algebra," *J. Discrete Math. Sci. Cryptogr.*, vol. 28, no. 2, pp. 349–354 (2025).
- [13] S. S. Dhanda, B. Singh, and P. Jindal, "Demystifying elliptic curve cryptography: Curve selection, implementation and countermeasures to attacks," *Journal of Interdisciplinary Mathematics*, vol. 23, no. 2, pp. 463–470 (2020).

Received April, 2025