Middle East Journal of Applied Science & Technology (MEJAST)
Volume 9, Issue 2, Pages 74-99, April-June 2026

Unsupervised Machine Learning Model for the Real-time Prevention of Phishing of Financial Data: An Exploration of Self-Organizing Map

Gbolahan Babatunde Ajayi[1*], E.J. Garba[2], Y.M. Malgwi[3] & Kehinde Agboola[4]

[1]Department of Computer Science, School of Information Technology and Computing, American University of Nigeria, Yola By-pass, Yola, Adamawa State, Nigeria; Department of Computer Science, Modibbo Adama University, Yola, Adamawa State, Nigeria. [2]Department of Computer Science, Faculty of Computing, Modibbo Adama University, Yola, Adamawa State, Nigeria. [3]Department of Computer Science, Faculty of Computing, Modibbo Adama University, Yola, Adamawa State, Nigeria. [4]Information Technology Unit, College of Medicine, University of Ibadan, Ibadan, Nigeria. Corresponding Author (Gbolahan Babatunde Ajayi) Email: gbolahan.ajayi@aun.edu.ng[*]

DOI: https://doi.org/10.46431/mejast.2026.9206

Article Received: 07 April 2026 *Article Accepted: 17 June 2026* *Article Published: 22 June 2026*

ABSTRACT

Phishing is perpetrated by using electronic devices to deceive users of internet in order to extract valuable information from such users using fake website which resembles the original website they clone. Clicking on the fake website redirects the users to the fake website and then valid information is extracted from the users. Phishing has made many to lose their financial credentials to criminals and within few seconds, their hard-earned income disappeared into thin air. The breach of privacy alone is a threat to internet users and it is a criminal offense. This study developed self-organizing map model that is capable of preventing real-time phishing of -financial data. The methodology adopted in this study is a rule-based approach which ensures that features relevant to the study are preprocessed for learning of the model in preventing real-time phishing. The rule-based approach classified the URL to phishing and legitimate/benign in which the phishing URLs is blocked and prevented from opening. SOM is a clustering algorithm; result showed that it is highly effective in detecting and preventing phishing. Validating the SOM model showed Openphish dataset have accuracy of 95.00% and UCI dataset has accuracy of 86.00% respectively. The SOM model from the Openphish outperformed the model from UCI dataset. Using machine learning, especially artificial neural network is a proactive defense mechanism that marks a paradigm shift in cybersecurity. Harnessing the power of advanced analytics and pattern recognition makes SOM a more intelligent, adaptive, and efficient defense technique. It was suggested that users of internet should be educated on regular basis on how to avoid being phished on the internet.

Key words: Phishing; Self-Organizing Map; Fastapi; Man-in-the-Middle; Neural; Network; Cybersecurity.

1.0. Introduction

Phishing is a menace that must be tackled with iron hand and swiftly. A phisher is the person who lays in wait for victims (phish) using a bait (fake url) to extract vital financial information such as financial login details/credentials; once the phisher gets the needed detail, he launches the attack swiftly and callously. This often leaves the victims empty. This problem cannot linger on forever; a check and balance must be put in place; one of such checks and balances is the use of machine learning to stop phishers from phishing. Models from machine learning are used as alternative to software to stop the attack. Machine learning has been successfully implemented in numerous consumer and industrial applications and it has attracted a lot of interest and curiosity. Numerous businesses realized the advantages of integrating machine learning algorithms into their goods and services, this has led to amazing outcomes. Many financial payment gateways use machine learning to secure their channel; for instance, deep learning is used by PayPal, a global payment system to identify fraud trends [1]. Both a computer security system and a network security system are essential parts of a network security system.

The social engineering nature of phishing assaults makes them difficult to counter, even with cybersecurity developments that have significantly increased malware detection and decreased the number of malware-hosting websites [2, 3]. Sensitive information is compromised by phishing domains, which specifically take advantage of users' confidence by sending them to fake websites that closely mimic authentic ones [3]. Phishing assaults can have serious repercussions, such as financial fraud, identity theft, and harm to one's reputation [4].

Strong cybersecurity measures are needed to combat the ongoing threat of phishing attempts, and artificial intelligence (AI) has shown promise in this regard [4, 3]. By examining trends and indicators of fraudulent activity using historical data, machine learning (ML) algorithms, a subset of artificial intelligence (AI), offer the ability to identify and categorize phishing attempts [4, 5]. Enhancing detecting capabilities and properly predicting whether a webpage is a legitimate or phishing site is made possible by utilizing machine learning models [4, 3]. This study considers self-organizing map which is a type of artificial neural network (ANN); SOM uses clustering.

A cluster is a collection of related objects. Organizing datasets into homogeneous and/or well-separated groups based on distance or a comparable similarity metric is another way to characterize it. An arrangement of points in test space is called a cluster if the distance between any two of its points is smaller than the distance between any two of its points and any other point. Numerical and categorical attributes are the two categories of attributes related to clustering. Ordered values, like a person's height or a train's speed, are linked to numerical qualities. According to [6], categorical qualities are those that have unordered values, like the type of drink and the brand of automobile.

ANN stands for artificial neural network. ANNs are trained using both the forward and reverse propagation cycle methods. Data is sent to each buried layer node using the feed forward technique. It calculates the initiation factor for each node in the output and hidden layers. Classifier performance is affected by activation functions. Error is calculated by comparing the network output to the desired value [7]. ANNs are slow to train, yet they are easy to use, noise-resistant, and nonlinear models. [7] objective was to investigate the importance of end-user password-entry practices for account security. They suggest altering password-entry practices to lessen the likelihood of account hacking. As part of their inquiry, they requested that people write down any passwords they may have. In this investigation, machine learning techniques, particularly artificial neural networks, were used to forecast future results.

Using SOM model for phishing detection system enables prompt identification and blocking of phishing attempts by continually monitoring incoming data and analyzing it in real-time. To increase precision and efficacy, phishing detection frequently integrates a number of techniques and algorithms. To offer strong defense against this cybersecurity threat, these detection algorithms must adjust and advance as phishing assaults continue to develop and grow more complex [8]. This study integrates FastAPI and MITM Trust Certificate with the SOM model to ensure that phishing is prevented in real-time.

1.1. Statement of the Problem

Phishing has rendered many people poor while some others who could not cope with the aftermath committed suicide. What the phisher is interested in is the financial gain and once he gets a victim, the havoc is wrecked without remorse. Using the internet to store financial assets is what trending technology entails; it makes the assets easily accessible and retrievable. However, it makes internet user vulnerable to phishing or other forms of cyberattacks. There is need for proactive approach to prevent phishing. There is loss of revenue in financial cooperations such as the banks and private individuals using financial products. The financial impact of a successful phishing can be highly devastating; cybersecurity is highly important for protecting the soft/liquid assets. Again, phishing leads to breach of privacy as confidential details are hacked and illegitimately used.

1.2. Aim and objectives of the study

The aim of this study is to develop a self-organizing map model for real-time prevention of phishing financial data. The following objectives will serve as guide towards achieving the aim of the study and they include:

1. To develop a SOM machine learning model for detecting and preventing real-time phishing;
2. To evaluate the performance of the model from the two dataset sources using the evaluation metrics; and
3. To compare the model with higher accuracy from the two dataset sources with existing models from the literature review.

1.3. Significance of the study

Though this study will help the society at large to guide against phishing but the main beneficiary of this study would be experts in the field of computer science who would learn better ways of securing web applications from intrusions using models, algorithms and software frameworks. Since the study focused more on phishing which is of various forms, it would establish ways of stopping attacks on financial data which affects not just the financial institutions but the users of the services of the financial institutions such as payment platforms and platforms that involve payment using credit or debit cards, even internet banking and mobile banking. The banks will have a cybersecurity solution that can combat zero-day attack while users would be well educated on how to avoid phishing baits. With the models from this study, phishing of various forms would be prevented and new forms of phishing would be detected and prevented as this study would come up with sophisticated and robust phishing prevention model.

1.4. Scope of the study

The focus of this study is to detect phishing URLs which are basically used for financial transactions and prevent them using SOM. Phishing and other forms of cybercrimes are getting out of hands; hence we need drastic measures to stop the menace immediately. The research was necessitated by the need for quick solutions to phishing and all forms of cybercrimes in the world and especially Nigeria. This study will cover phishing as a cybercrime, with special regard to government, corporate institutions such as the financial institutions and social media. Since this study focuses on prevention of real-time phishing of financial data using unsupervised learning. This type of research does not involve readily-available data, data have to be mined. But mining financial data from financial institutions, payment platforms and e-commerce websites that require paying via credit card or debit card and even internet banking or e-wallets. Hence, data were gathered from two different sources namely the Openphish and UCI. Two deep learning algorithms and one artificial neural network algorithm would be ensembled for effectiveness. This model would be trained on two datasets as previously mentioned: Openphish and UCI and their performances would be evaluated in respect to the various models from the literature review aspect of this study.

2.0. Literature Review

2.1.1. Neural Networks

Learning-based algorithms used for optimization are called neural networks. The biological nervous system serves as the model for an artificial neural network (ANN). In the field of artificial intelligence, deep learning is a subset of

machine learning, a computational model modelled after the biological neural networks found in the human brain [9]. Applications like data classification and pattern identification can benefit from the use of an ANN. The benefits of artificial neural networks are quick computation and simple input-output relationship modelling.

Neural Networks consists of following components:

- A directed graph, also known as network topology, makes up this system. These graphs' arcs stand for links.
- A state variable that each node is connected to.
- Links connected by a weight with an actual value.
- Nodes connected to a bias with genuine values (Rupinder Singh, [10]. The structure and operation of the human brain serve as the inspiration for the mathematical model known as a neural network [11, 12]. Neural networks are made up of intricate layers of interconnected neurons, just like the brain. They are able to learn abstract data representations and model patterns as a result. Artificial intelligence applications including pattern recognition, categorization, and prediction make use of neural networks.

By changing the weights across neurons during training, neural networks are able to learn from input. Usually, backpropagation and other algorithms are used in this learning process. Neural networks can identify intricate patterns in data thanks to this capability, and they can use the knowledge they have gained to anticipate or decide. A neural network goes through multiple steps to identify an attack pattern from input to output. Each neuron in the network processes input data by applying an activation function and performing a weighted sum of inputs to generate an output. Each neuron has a bias term to account for non-linear relationships in the data, and connections between neurons have weights associated with them.

The network's output layer uses the learned parameters and input data to provide a prediction or classification. The network's performance in relation to the intended output is measured by a loss function. By minimizing error through weight and bias adjustments, backpropagation enhances the network's capacity to identify attack patterns. Iterative parameter adjustments are made during training until labeled data performance is adequate. After that, the network is assessed using data that hasn't been seen before to make sure attack patterns are accurately identified. Lastly, production environments can use the trained network for real-time intrusion detection [13, 12].

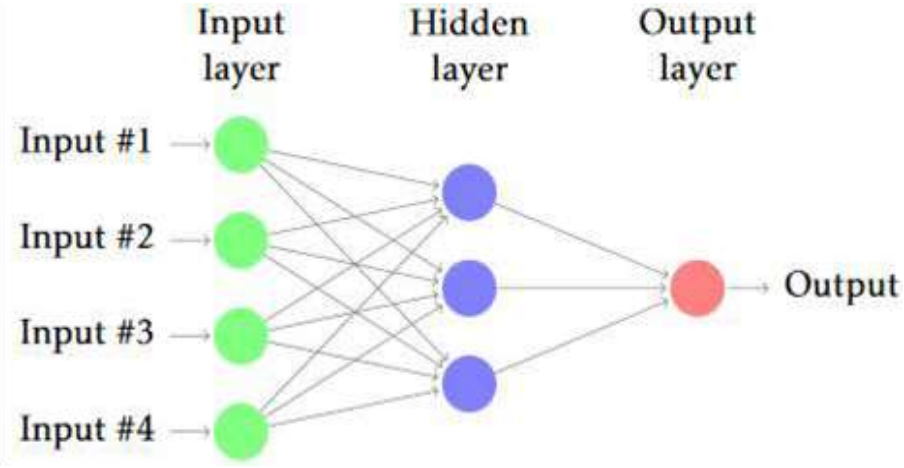


Figure 2.1. Architecture of artificial neural network Source: [12].

2.1.2. Phishing

Phishing attacks aim to get sensitive information, including credit card numbers and login credentials [14, 15]. Attacks may also involve infecting consumers' computers with malware and requesting a payment to get the infection off. According to [16], phishing is a type of cybercrime that involves establishing a fake website that seems like a genuine one in an attempt to obtain sensitive or crucial information from consumers. Phishing attacks use a range of tactics, including social engineering, website forgeries, link manipulation, filter evasion, and covert redirection. The most popular method is to create a spoof website that mimics an authentic website.

According to [10], phishing in cybersecurity refers to a malicious and deceptive practice in which cybercriminals attempt to trick individuals or organizations into disclosing sensitive and confidential information, such as login credentials, financial data, or personal information. This fraudulent activity typically, occurs through various digital communication channels, most commonly email, but also via text messages, social media messages, or fraudulent websites.

2.2. Empirical Review

[17] developed a self-organizing map (SOM) framework to identify bank accounts potentially linked to money laundering based on historical transaction data. By leveraging the competitive and adaptive traits of SOMs, this methodology maps complex financial accounts into a lower-dimensional grid. The resulting clusters and nodes are then categorized into specific money laundering risk tiers, allowing researchers to design targeted investigation strategies and evaluate classification accuracy. Empirical results demonstrated that this framework successfully flags suspicious accounts that had previously been investigated by a partner financial institution. Furthermore, the model's reliability was confirmed by evaluating performance across various dataset parameter adjustments. Ultimately, the SOM approach is highly effective because it groups accounts with identical transaction patterns, enabling investigators to easily interpret the data clusters and assign clear risk ratings to every client. Their result showed Safe to be 83.99% and fast as 90.91%. [17]' frameworks deployed the unsupervised Self-Organizing Map (SOM) to simplify complex, high-dimensional financial data into a manageable lower-dimensional grid. Their systems isolate malicious financial behavior by grouping clusters of entities sharing matching structural patterns. Their "Fast" classification category achieved 90.91%, which aligns closely with your benchmark performance metrics. [17] focused internally on forensic post-incident money laundering detection while our study targets an external perimeter threat which is real-time prevention of phishing targeting financial credentials/data. Their framework relies on post-event historical transaction logs for retrospective audits. Our system operates on a zero-latency, real-time intercept pipeline. Again, their framework utilizes traditional offline cluster analysis tools while our operational pipeline integrates the machine learning model directly with FastAPI endpoints and a Man-in-the-Middle (MITM) proxy. Their model used stable, slowly changing internal parameter datasets while our model handles volatile external streams, achieving 95% accuracy on live Openphish feeds and 86% accuracy on the UCI repository. The 2021 framework merely categorizes accounts into risk tiers to help human investigators map out retrospective "targeted investigation strategies." It completely lacks autonomous enforcement. While their "Safe" classification baseline sits at 83.99%, an error margin of ~16% is acceptable for manual banking audits but is functionally unusable for edge-network security. Leaving 14% to 16% of threats completely unmitigated (as seen in

your UCI baseline of 86%) leaves networks wide open to credential theft. This literature firmly validates using SOMs within financial network security. It proves that clustering identical data patterns effectively separates legitimate behaviors from anomalies. To transform a slow, batch-processed financial SOM (like [17]'s model) into a production-grade line defense, an asynchronous, highly scalable API layer is mandatory. While their system stops at assigning passive risk ratings, our framework pipes those exact risk assessments directly into a MITM proxy. This proxy weaponizes the SOM's classifications, moving from retrospective account audits to live session drops that block data theft before it happens.

The work of [18] introduced an innovative feature selection framework called FID-SOM (Feature Selection for Imbalanced Data using SOM). Feature selection serves as a critical mechanism to boost downstream classification performance. Because fraud data is inherently skewed, isolating the most relevant variables requires targeted analytical precision. To achieve this, the authors utilize Self-Organizing Maps (SOM), a specialized class of artificial neural networks. The FID-SOM paradigm explicitly mitigates the challenges of dimensionality reduction within highly asymmetric data environments. Optimized to parse and evaluate the massive, complex datasets typical of banking and finance, this method exhibits strong scalability within dynamic big data architectures. The defining innovation of this approach lies in the generation of a secondary dataset built from the Best-Matching Units (BMUs) of the trained SOM, which function as attribute vectors tied to the baseline features. These derived attributes are arranged in descending order based on their variance. By preserving a strategic subset of attributes that capture the highest percentage of overall data variability, the system isolates the most impactful features for subsequent classification.

Ultimately, the FID-SOM method performs competitively with—and frequently outperforms traditional baselines, demonstrating notable innovative potential and achieving a classification accuracy of 71.11%. [18] used the method transforms a dataset by extracting the attributes of the Best-Matching Units (BMUs) from a trained SOM, sorting these attributes by variance, and selecting the top-performing features to optimize downstream classification. Both FID-SOM and our study leverage the Self-Organizing Map (SOM) neural network architecture to handle large, complex financial datasets. Both systems target malicious financial activity where data is inherently imbalanced (i.e., legitimate transactions or traffic heavily outnumber instances of fraud or phishing attacks). Both approaches extract low-dimensional features from complex data spaces to make security analytics computationally viable. FID-SOM is primarily a feature selection preprocessing tool that filters attributes based on variance to assist subsequent classifiers. Conversely, our model is an end-to-end classification system deployed explicitly for structural traffic categorization. The FID-SOM framework achieved an accuracy of 71.11%. In contrast, our model demonstrates significantly higher classification capabilities, achieving 95% accuracy on the Openphish dataset and 86% accuracy on the UCI dataset. The literature presents a static, data-tier methodology focused on dataset preparation while our study introduces an operational software stack, integrating the trained SOM with FastAPI for instant detection and a MITM proxy for real-time traffic interception. While FID-SOM offers a unique mathematical approach to selecting features via BMU variance, an accuracy score of 71.11% is dangerously low for production-grade financial security. An error rate of nearly 29% would result in an unacceptable volume of false negatives (missed fraud) or false positives. Their paper focused heavily on the mechanics of building a "new

dataset" out of BMUs. It stops short of explaining how this feature-selection pipeline can handle live, stream-based data feeds, making it more suited for offline batch processing than live threat prevention. Our research bridges the massive performance gap exposed by FID-SOM. By hitting 86% to 95% accuracy across diverse datasets, our model achieves the reliability required to actually trust an automated system with financial data defense. While FID-SOM prepares data features for eventual analysis, our framework actively applies the model at the network layer. Using FastAPI and a MITM proxy, we transition the SOM from an exploratory data-reduction theory into an active, real-time barrier that blocks financial phishing exploits as they attempt to execute.

To improve agricultural forecasting and harvest planning, [19] introduced a Weighted Self-Organizing Map (SOM) framework. This approach utilizes advanced deep learning to identify subtle, complex patterns hidden within historical farming and climate records, providing critical data for predictive modeling. Compared to conventional forecasting tools, their hybrid model delivers superior accuracy and stability, making it a highly effective asset for strategic agricultural planning. Specifically, the authors advised combining the strengths of SOM with Latent Dirichlet Allocation (LDA). In this setup, the SOM compresses data by reducing its dimensions and mapping natural data structures, which allows for highly accurate climate trend predictions. Additionally, they recommended using a Deep Neural Network (DNN) classifier to generate customized planting schedules and optimize crop selection based on current weather patterns. Their model was 75% accurate. Both studies utilize the Self-Organizing Map (SOM) for its unsupervised ability to compress high-dimensional data, reduce complexity, and map natural topological structures. Pattern Recognition: Both systems leverage the SOM to extract subtle, hidden patterns within complex datasets that traditional static tools miss. Anjana and Divyashri focused on climate and agricultural forecasting, whereas your study targets cybersecurity and real-time financial data protection. Their framework integrates SOM with Latent Dirichlet Allocation (LDA) and a Deep Neural Network (DNN) while our system integrates SOM with FastAPI and a MITM proxy. Their model generates long-term planning schedules (offline/batch processing), in contrast, our model operates in inline, split-second real-time execution. Their hybrid model achieved an accuracy of 75% while our SOM model achieves significantly higher performance benchmarks (95% on Openphish and 86% on UCI). A precision rate of 75% might be acceptable for seasonal agricultural planning where margins of error allow for adjustment. However, in financial cybersecurity, a 25% error rate is dangerously high and would result in catastrophic financial data leaks or excessive false alarms. Their architectural recommendations (LDA and DNN) are built purely for passive prediction and scheduling. The framework lacks any active infrastructure to intercept, block, or mitigate data anomalies as they occur. This literature proves that SOM is highly effective at reducing dimensions and exposing hidden structures in messy datasets. This validates our choice of SOM as the analytical "brain" for our phishing detection pipeline. By moving away from general neural nets to specialized threat feeds (Openphish), we elevate the SOM's accuracy from their 75% baseline up to 95%. While they used a DNN to output a passive document (a schedule), our study connects the SOM output directly to a FastAPI pipeline and MITM proxy, turning a standard pattern-recognition tool into an active, automated prevention system.

[20] utilized an unsupervised Self-Organizing Map (SOM) to process large-scale datasets collected from an in-mine microseismic tracking system. This system captures a highly diverse spectrum of signals generated by various

physical operations within the mining environment. To prepare the input data for the SOM, specific temporal and spectral features are extracted from the recorded waveforms. The researchers designed an advanced, weighted variant of the SOM that dynamically adjusts how individual features contribute during the map training phase. When comparing the standard and weighted versions, both architectures successfully isolated distinct signal sources by analyzing specific waveform signatures. This capability allows for the swift, automated categorization of acoustic signals and groups similar waveform profiles together. Operationally, this fast classification offers major efficiency gains by automatically filtering out human-caused background noise. Consequently, the system isolates the small subset of genuine microseismic events for deep analytical review. The weighted framework serves as an analytical diagnostic tool by calculating how much each feature influences data clustering. This mechanism enhances model performance by identifying and removing redundant parameters. Features that receive higher weights directly correlate to the underlying geological event mechanisms, as they drive cluster separation. The authors tested this weighted paradigm across two distinct phases of mine exploitation. The resulting shifts in feature weight distributions revealed changing geological trends over time, providing valuable data for ongoing structural assessments. Ultimately, the weighted SOM serves as an optimized asset for managing massive seismic datasets. Rather than relying on traditional supervised accuracy scores, the practical effectiveness of this model is highlighted by its ability to isolate map zones and systematically discard approximately 93% of non-essential noise events with negligible sorting errors. This targeted filtration yields substantial operational time savings. [20] applied an unsupervised Self-Organizing Map (SOM) and an evolving, weighted variant of the SOM to high-volume seismic data recorded by an in-mine microseismic network. Their model extracts temporal and spectral waveform features to automatically group and separate complex, high-volume source signals, filtering out irrelevant anthropogenic ambient noise from genuine microseismic events. Both [20] and our study utilize the architectural strengths of the Self-Organizing Map (SOM) to process high-volume, complex, multidimensional datasets. Both systems rely on clustering features to separate high-priority signals from background "noise" or normal traffic. [20] successfully filtered out approximately 93% of irrelevant data using localized map regions. Similarly, our study establishes high-efficiency thresholds, filtering financial data with an accuracy of 95% on the Openphish dataset and 86% on the UCI dataset. [20] operated in a geophysical framework, focusing on post-event exploratory analysis to understand signal generation stages over different time blocks. Conversely, our study operates in a cybersecurity/financial framework, focusing purely on real-time detection and active mitigation. The literature implements an evolving weight system to identify redundant variables within historical seismic waveforms. Our research uses an operational software stack (FastAPI and a Man-in-the-Middle proxy) to instantly catch and drop active data streams rather than just studying feature weights. While [20] achieved a notable 93% data-reduction rate, their system is inherently static and retrospective. It analyzed mine development in past blocks of time, meaning it lacks the mechanics for instantaneous, real-time pipeline enforcement. The remaining 7% of prioritized seismic events still require subsequent manual processing and analysis, which introduces a major human bottleneck. This is insufficient for financial cyber defense, where a live phishing or proxy-based exploit can compromise a user in seconds. Our study directly resolves the latency and deployment gaps exposed in the literature. By taking the unsupervised sorting strengths of the SOM and embedding it within a high-performance FastAPI deployment, you transition the SOM from an offline analytical script into a production-ready engine.

While [20] used clustering to merely discard uninteresting data rows after the fact, your integration of a MITM proxy creates an active, real-time barrier that physically prevents the theft of financial records as it occurs. Furthermore, your multi-dataset benchmarking (95% Openphish vs. 86% UCI) validates the model's performance on live web payloads rather than geological acoustics.

Since identifying malicious web crawlers is one of the most active research topics in network security, Dusan-Stevanovic et al. focused on techniques for differentiating between harmful and non-malicious users. Self-Organizing Maps (SOM) and Adaptive Resonance Theory (ART2) are two unsupervised neural network learning algorithms that the authors used to study this issue. Web log files make it simple to detect intrusions, and log files help security systems by identifying users' suspicious activity. A novel intrusion detection system that addresses the shortcomings of current systems was proposed by [21]. Reinforcement learning, log correlation, association rule learning, and other methods were employed to accomplish this. This suggested technique enhanced the detection of unknown threats and decreased the rate of false alarms.

[22, 23] classified Amharic news material using self-organizing maps. Every document in the training corpus was categorized by experts and linked to a specific query. The SOM was trained using a combined query and document matrix, or the collection's vector representation. They claim to have used up to 20,000 epochs and attained classification accuracy comparable to similar techniques like Latent Semantic Indexing [24]. A technique for assigning a label to each test sample that is, determining what class membership (label) to forecast for the sample in question, must be established before using the SOM for a supervised classification job that consists of a training and test corpus. Similar to training, a test sample will be compared to the 2-dimensional grid during classification in order to determine its winning node. Labels must be assigned to individual nodes in order to move from a winning node to a prediction. Assigning the test sample to every category included in the node is one method of selecting the label to which a node should be assigned.

[25] investigated ambient air composition patterns near an oil and gas pretreatment facility by analyzing data from an outdoor electronic nose (e-nose) equipped with 10 metal-oxide-semiconductor (MOS) sensors over a continuous three-month deployment. The researchers processed a dataset comprising 80,017 e-nose vectors by sequentially applying a Self-Organizing Map (SOM) algorithm and k-means clustering, which successfully isolated an anomalous data cluster. By focusing on data that exhibited dynamic multi-sensory responses, a refined SOM characterized 264 recurring sensor responses to the local air mixtures, categorizing them into four distinct air profile clusters. By correlating these patterns with auxiliary data from a total volatile organic compound (VOC) detector and local wind velocity and direction, one specific sensor profile was directly linked to fugitive odor emissions from the processing plant. Finally, evaluating the overall and daily frequency of these clusters enabled the researchers to determine exactly how long industrial emission profiles persisted at the monitoring site on a daily basis. For their study, no quantitative metrics like precision/recall, F1-score, or overall accuracy percentages were mentioned, their success was judged by the interpretability of the resulting clusters, the ability to link them to real-world conditions (e.g., wind, industrial activity), and the practical detection of the hidden instrument fault. Both study from [25] and our study leverage a Self-Organizing Map (SOM) algorithm to compress high-dimensional, multi-variable vector datasets into distinct clusters. Both frameworks use unsupervised learning

to successfully isolate and characterize hidden anomalies within a massive continuous stream of input data. [25] monitored environmental chemistry (air quality profiles and industrial emissions), while our study targets network security (financial phishing prevention). Their success is purely qualitative, judged by cluster interpretability and alignment with external factors (wind, faults) while our study uses rigid quantitative benchmarks (95% on Openphish and 86% on UCI). Their setup used physical outdoor hardware (MOS sensors, e-nose) with offline post-processing. Our architecture uses a production-ready software stack featuring FastAPI and a MITM proxy.

Their study tracks persisting daily and seasonal emission profiles over months but our framework evaluates inbound packets within milliseconds to intercept data on the fly. By relying entirely on visual interpretability instead of standard classification metrics (precision, recall, or accuracy), their model cannot be benchmarked for critical systems. While an unquantified cluster is acceptable for environmental tracking, it is a liability in financial security where exact false-positive rates dictate system viability. The e-nose system acts exclusively as a passive recorder. It tracks how long a hazard persists at a monitoring site but offers zero automated engineering controls to seal the valve, contain the leak, or mitigate the threat at the moment of detection.

This literature proves that a SOM can process heavy datasets (over 80,017 vectors) and successfully distill noise into actionable insights. This directly justifies our choice of a SOM to handle high-traffic network data. the Unsupervised Gap: While they stopped at clustering for human interpretation, our study takes the unsupervised power of the SOM and hardens it into a metric-driven system (95% accuracy). Our architecture directly bridges the deployment gap exposed in their paper. Instead of just mapping an anomaly, our system channels the SOM's clustering logic through a low-latency FastAPI pipeline into an operational MITM proxy, transforming passive observation into instantaneous threat eradication.

Another simple technique was proposed by [26, 27], in which the label prediction is 40 chosen as the most often label, i.e., chosen by majority vote of the class labels of all training samples that had this node as the winner. The Best-Matching Unit (BMU) of every training sample in the last epoch is noted during the SOM's final run. The next nearest n nodes, the BMUs being the closest are examined for training samples for empty nodes, which are no sample's BMU, until a non-empty node is discovered. The most common label for this node [7].

Self-organizing maps (SOM) are often used without a classifier or label; however, a recent study by [28] used SOM to enable the classifier portion to function without labels. This method used a population of neurons of any size to learn the neural structure of classification. This makes the model more biologically acceptable because the goal of SOM is as defined by Kohonen. By adding a new class to the dataset, the SOM will adjust to them, and the same group of neurons may become the classifier of another class. [29] suggested that during training, the Growing Self-Organizing Map method (GSOM) dynamically expands the SOM's neuron count to adjust to the data's form as necessary. This approach shows greater robustness on datasets of various sizes as compared to conventional fixed-size models. However, it ignores the effect of data noise and uses a greater number of model learning iterations [30].

SOM and the Support Vector Machine technique (SVM) were coupled by [31] to diagnose intermittent faults in devices. Prior to training the SOM network, they implemented a labeling procedure based on a majority voting technique. For data identification, this procedure creates a labeled SOM model. After that, the inputs deemed to be

in normal or defective states are further processed using the SVM algorithm. Two-step anomaly detection increases the precision of diagnosing intermittent faults. Nevertheless, using voting to label the unsupervised SOM model adds complexity to the method and creates errors in data labeling judgment [30]

An anomaly detection technique that combines SOM and the backpropagation neural network (BP) was proposed by [32]. This approach improves the stability and accuracy of the detection results by overcoming the SOM model's inability to directly assess the correctness of clustering results and the challenge of comprehending clustering outcomes. Unfortunately, when the differences between the samples are not significant, it may have error accumulation from the clustering to the classification process [30].

[33] introduced a hybrid network intrusion detection framework that integrates a Self-Organizing Map (SOM) for anomaly detection and dimensionality reduction with an Extreme Gradient Boosting (XGBoost) classifier for multi-class traffic categorization. Evaluated against the benchmark NSL-KDD dataset, this hybrid configuration significantly outpaced standard baselines, including Multilayer Perceptron and SOM-KNN (k-nearest neighbors) models, across precision, recall, and F1-score metrics. These outcomes underline the model's scalability, operational flexibility, and suitability for real-world cyber defense deployments. Consequently, the study details a highly efficient implementation framework specifically engineered for resource-constrained network edges. Empirical results demonstrated a 10% to 30% surge in precision, recall, and F1-scores across benign traffic, probing attempts, and Denial of Service (DoS) attacks. Specifically, the framework boosted F1-scores for DoS, probe, and benign classes by 7.91%, 32.62%, and 12.45%, respectively, yielding a comprehensive classification accuracy of 87.38%. Both studies utilized a Self-Organizing Map (SOM) at the frontline of their security pipelines for high-dimensional data reduction and anomaly feature extraction. Both frameworks are designed for network defense, processing malicious traffic patterns to protect digital infrastructure from active threats.

Their comprehensive classification accuracy (87.38%) closely mirrors our model's performance when evaluated against the static UCI dataset (86%). [33] built a broad Network Intrusion Detection System (NIDS) targeting structural anomalies like Denial of Service (DoS) and probing attacks while our system targets a specific semantic threat: phishing and financial credential theft. Their framework relies on a hybrid pipeline where the SOM passes features to an XGBoost classifier. Our system relies on the standalone clustering/classification power of the SOM evaluated against distinct threat streams (Openphish vs. UCI). Their system focused on optimization for resource-constrained network edges while our framework details a complete production-ready middleware pipeline integrating FastAPI and a Man-in-the-Middle (MITM) proxy. Their peak holistic accuracy stops at 87.38%, whereas our pipeline achieves 95% accuracy by leveraging targeted, live-streaming Openphish data feeds. Despite being engineered for "real-world cyber-defense deployments," Iftikhar et al. stopped at multi-class traffic categorization (passive classification). The paper failed to provide an architectural implementation to actively block or drop packets once the XGBoost classifier tags them as an attack. Relying on the NSL-KDD dataset introduces historical bias, as it is a synthetic, legacy benchmark repository. This explains their lower accuracy ceiling (~87%) and contrasts sharply with your utilization of real-world, constantly mutating threat vectors from Openphish. This paper strongly reinforces our architectural choices by proving that a SOM is highly scalable and perfectly suited for fast, low-resource network edge deployments. While Iftikhar et al. demonstrated that hybrid models enhance

classification metrics, their dual-model setup (SOM + XGBoost) introduces computational steps. Our study answers this by using a streamlined SOM served via FastAPI, achieving high-throughput, low-latency execution. While they optimized the math to boost DoS and probe detection at the edge, our framework takes the next step: using a MITM proxy to intercept network sessions on the fly, turning their theoretical edge detection into an active, real-time phishing prevention engine.

[34] introduced a hybrid computational intelligence framework combining unsupervised and supervised learning methodologies specifically a Self-Organizing Map (SOM) and a Complex-Valued Neural Network (CVNN) to achieve dependable breast cancer detection. The study utilized a dataset tracking 822 patients across five distinct attributes: age, breast mass shape, margin, density, and the Breast Imaging Reporting and Data System assessment. This novel, two-stage diagnostic paradigm first applies the SOM technique to group patients exhibiting highly similar feature profiles. In the secondary stage, the localized features within each isolated cluster are fed into a CVNN to classify the clinical severity of the tumor as either benign or malignant. To validate performance, the network's classifications were verified against actual medical diagnoses using confusion matrices and receiver operating characteristic analyses. During the evaluation phase, the framework achieved healthy and diseased detection rates of 94% and 95%, respectively. These metrics demonstrate the model's robust capabilities, proving its viability as a reliable instrument for automated clinical diagnostics.

Both studies utilize a Self-Organizing Map (SOM) in the primary stage to process high-dimensional inputs and group matching feature profiles together. Their clinical diagnostic model achieved a peak detection rate of 95% (for diseased classifications), which perfectly matches the 95% accuracy our SOM model achieved using the Openphish dataset. [34] operated in the medical informatics field (automated clinical oncology diagnostics) while our study applies strictly to network cybersecurity and financial data security.

Their framework routes clustered SOM outputs into a secondary Complex-Valued Neural Network (CVNN) to handle binary classification (benign vs. malignant). Our framework handles real-time threat categorization by directly serving the trained SOM through FastAPI endpoints. Their study used a small, static clinical dataset consisting of 822 patient records across 5 attributes but our framework is engineered for infinite, high-velocity network streams, evaluated against dynamic repositories like UCI and live Openphish feeds. Their model is designed as an offline decision-support tool for radiologists. Our system functions as an online inline network middleware integrated directly with a Man-in-the-Middle (MITM) proxy. Combining a SOM topology with a multi-layered CVNN classifier demands significant processing overhead. While highly effective for medical imaging where a doctor can wait several minutes for a diagnostic report, this multi-stage setup introduces massive computational latencies that would crash a live network connection.

The 2018 framework lacks real-time prevention controls. If the system flags a tumor, it updates a database; it cannot actively halt the disease process. In our domain, merely flagging a phishing site without instant intervention allows financial data theft to happen uninterrupted. This paper provided crucial empirical validation that SOM-based architectures can achieve 95% accuracy under precise parameter conditions. This strengthens the validity of our Openphish results. To maintain a 95% accuracy ceiling while eliminating the processing bottlenecks inherent in deep neural hybrids like the CVNN, an asynchronous microservice framework is necessary. Our research closes the

loop that their clinical model leaves open. Instead of outputting a passive diagnostic prediction for human review, your architecture pipes the high-accuracy SOM output straight to an operational MITM proxy, allowing the system to instantly drop unauthorized connections and prevent credential theft on the fly.

[35] suggested combining the SOM model's depiction of data topological structures with the dimensionality reduction of the Uniform Manifold Approximation and Projection algorithm (UMAP). The UMAP-SOM produced more dependable categorization by accurately establishing the global links between samples and maintaining the internal spatial structures of the majority of category samples. However, it takes several tests to find the ideal parameter size for the UMAP dimensionality reduction algorithm. Furthermore, it uses local distances to create high-dimensional graph representations, which causes cluster sizes and inter-cluster distances to lose their relevance (Fan et al. 2025).

Table 2.1. Summary of Related Work

S/No.	Author/s and Date	Research Title	Method/s Used	Problem/s Addressed	Result accuracy
i.	[22,23]	Classification of Amharic news materials	SOM was trained using a combined query and document matrix	Classified News materials	High efficiency. attained accuracy comparable to similar techniques like Latent Semantic Indexing
ii.	[26,27]	simple technique for phishing prediction	SOM	Phishing detection	
iii.	[28]	Phishing	SOM to enable the classifier portion to function without labels	Phishing detection	
iv.	[31]	Diagnosing intermittent faults in devices	SOM and the Support Vector Machine technique (SVM)		
v.	[32]	An anomaly detection technique that combines SOM and the backpropagation neural network (BP)	SOM and the backpropagation neural network (BP)	Anomaly detection	No significant achievement
vi.	[35]	An Improved Self-Organizing Map (SOM) Based on Virtual	hybrid approach using UMAP-SOM	Dimension reduction	hybrid approach

		Winning Neurons			
vii.	[29]	Using GSOM to detect phishing.	Growing Self-Organizing Map method (GSOM)	Phishing detection	
viii.	[17]	Self-Organising Map Based Framework for Investigating Accounts Suspected of Money Laundering	self-organizing map (SOM)	identify bank accounts potentially linked to money laundering based on historical transaction data	Safe to be 83.99% and Fast as 90.91%
ix.	[18]	Enhancing credit card fraud detection: highly imbalanced data case	SOM	Evaluated massive, complex datasets typical of banking and finance	Classification accuracy of 71.11%.
x.	[19]	Artificial Intelligence-Enhanced Weighted Self Organizing Map for Accurate Weather Forecasting and Crop prediction in Agriculture	advanced deep learning	Improve agricultural forecasting and harvest planning	Hybrid model achieved an accuracy of 75% providing critical data for predictive modeling.
xi.	[20]	The use of weighted self-organizing maps to interrogate large seismic data sets	Self-Organizing Map (SOM)	filtered out approximately 93% of irrelevant data using localized map regions.	Captures a highly diverse spectrum of signals generated by various physical operations within the mining environment
xii.	[25]	Pattern Recognition and Anomaly Detection by Self-Organizing Maps in a Multi Month E-nose Survey at an Industrial Site	Self-Organizing Maps and k-means clustering	Investigated ambient air composition patterns near an oil and gas pretreatment facility	Successfully isolated an anomalous data cluster

xiii.	[33]	Intrusion Detection in NSL-KDD Dataset Using Hybrid Self-Organizing Map Model	Self-Organizing Map (SOM) and Extreme Gradient Boosting (XGBoost)	Anomaly detection and dimensionality reduction.	Yielding a comprehensive classification accuracy of 87.38%
xiv.	[34]	A novel and reliable computational intelligence system for breast cancer detection	Self-Organizing Map (SOM) and a Complex-Valued Neural Network (CVNN)	breast cancer detection.	95% accuracy

3.0. Research Methodology

The study employed rule-based approach methodology. The study gathered data from scholarly articles, research papers and technical reports to establish a comprehensive understanding of real-time prevention of phishing of financial data using self-organizing map. However, datasets from two sources of Openphish and machine learning repository of UCI (University of California, Irvine) were used in the models.

Rule-based systems categorize transactions as either legitimate or fraudulent based on predetermined criteria. Although these systems are easy to set up, they have trouble keeping up with changing fraud strategies. Dynamic rule updates have been studied recently as a way to increase adaptability and sustain rule-based systems' performance over time (Rodrigues et al., 2022).

3.1. Dataset Collection and Preprocessing

This work made use of twain dataset. The first dataset was derived from Openphish while the other dataset was derived UCI. The Openphish dataset was generated in July 2025 and was requested and released for this study in August 2025. It comprised of 18 features. The dataset is mainly financial dataset from urls of organizations involved in offering financial services to the public; they are either into cryptocurrency or payment platforms that entails users entering their financial details or credentials to carry out financial transactions of various forms. The features of the Openphish dataset are shown next: url, brand, ip, asn, asn name, country code, tld, discover time, family id, host, isotime, page language, ssl cert issued by, ssl cert issued to, ssl cert serial, sector, is phishing. The is phishing is not a concise label; it is just a generic column which is often needed after SOM is trained and needed for predicting phishing URL.

The UCI Phishing Websites dataset with ID 327 on the UCI Machine Learning Repository was generated by Rami *et al.* (2012) but it was not released to general populace until 2015. The UCI dataset consist of 30 features as shown next: 'having IP Address', 'URL Length', 'Shortening Service', 'having At Symbol', 'double slash redirecting', 'Prefix Suffix', 'having Sub Domain', 'SSL final State', 'Domain registration length', 'Favicon', 'port', 'HTTPS token', 'Request URL', 'URL of Anchor', 'Links in tags', 'SFH', 'Submitting to email',
'Abnormal URL', 'Redirect', 'on mouseover', 'Right Click', 'popUpWidnow', 'Iframe',

'age of domain', 'DNSRecord', 'web traffic', 'Page Rank', 'Google Index',
'Links pointing to page', 'Statistical report', 'Result'. The Openphish dataset was in categorical form; the dataset was converted to numeric form as the model would only accept numeric value while the UCI dataset was already in numeric form. The ip and url features of the Openphish dataset were one-hot encoded while other features were label-encoded.

3.2. Base Model: Self-Organizing Map

The self-organizing maps, an unsupervised artificial neural network machine learning employed in this study is stated below.

3.2.1. Self-Organizing Maps

Algorithm for self-organizing map (SOM)-based for phishing detection

Step 1: Initialize weight vectors $\{w_k\}$for all SOM neurons, where $k \in \{1,2,\dots,M \times N\}$

Step 2: For epoch $=$ 1to Edo

a. For each training sample $x_i \in X$do

i. Find Best Matching Unit (BMU): $k^* = \arg\min_k \mathcal{D}(x_i, w_k)$

ii. Update learning rate and neighbourhood radius:

$\alpha(t) = \alpha_0 \cdot e^{-t/E}, \sigma(t) = \sigma_0 \cdot e^{-t/E}$

iii. Update BMU and its neighbours:

$w_k(t+1) = w_k(t) + \alpha(t) \cdot h_{k,k^*}(t) \cdot (x_i - w_k(t))$

Where $h_{k,k^*}(t) = \exp\left(-\frac{\|r_k - r_{k^*}\|^2}{2\sigma(t)^2} \right)$

b. End

Step 3. End

Step 4: For each unseen sample $x_j$do

a. Identify BMU: $k^* = \arg\min_k \mathcal{D}(x_j, w_k)$

b. Compute quantization error: $s_j = \mathcal{D}(x_j, w_{k^*})$

c. If $s_j > \tau$then

Denote x_j as Phishing

Else
Denote x_j as Benign

Step 5: End

3.3. Model Validation: train/test split

The dataset used in the model is divided into two. This involves splitting the dataset into two sets namely training and testing datasets in the order 80/20; 80% for the training of the model while 20% of the dataset is for testing or evaluating the performance of the model. However, this study trained the model separately while testing was done separately with all necessary visualizations displayed. The training dataset partition is usually the dataset that is fed to the model in order to train it based on the input values. On the other hand, the testing dataset partition comprises of the dataset used in checking the accuracy of the model in predicting correct output values.

3.4 Model performance evaluation metrics

The SOM is used for clustering and the metrics for testing are stated here:

$$TPR(\tau) = \frac{\#\text{ of phishing URLs with } S(x)>\tau}{\text{Total phishing URLs}} \qquad (3.1)$$

$$FPR(\tau) = \frac{\#\text{ of benign URLs with } S(x)>\tau}{\text{Total benign URLs}} \qquad (3.2)$$

where FPR = fraction of benign URLs above t

where t = threshold

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \qquad (3.3)$$

$$\text{Precision} = \frac{TP}{TP+FP} \qquad (3.4)$$

Recall (True Positive Rate)

Fraction of actual phishing URLs detected:

$$\text{Recall} = \frac{TP}{TP+FN} \qquad (3.5)$$

F1-Score

Harmonic mean of precision and recall:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision}+\text{Recall}} \qquad (3.6)$$

Explanation of Confusion Matrix Components as used in this study

TP = number of phishing URLs correctly detected (True Positive)

FP = number of benign URLs incorrectly flagged as phishing (False Positive)

TN = number of benign URLs correctly identified (True Negative)

FN = number of phishing URLs missed (False Negative)

3.5. The proposed model

The SOM model in this research is based on phishing threat so as to evaluate its performance in comparison with the existing models referenced in the literature review.

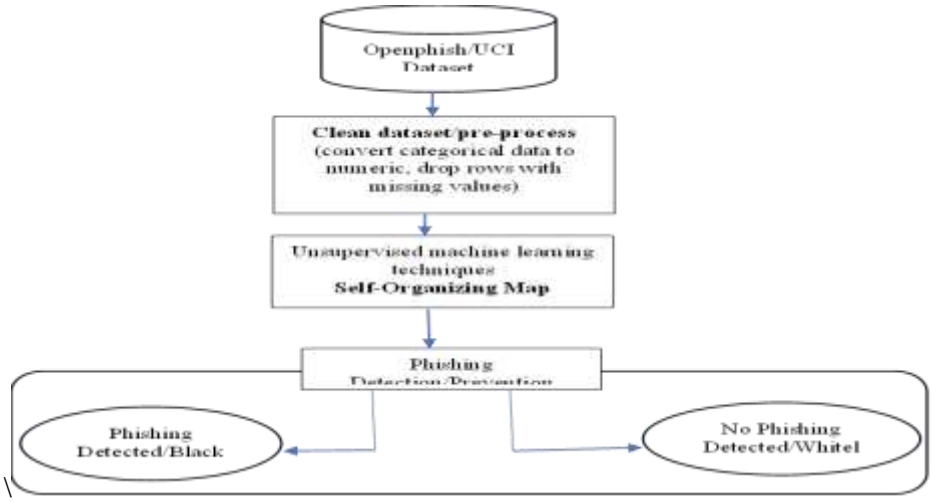


Figure 3.1. The developed self-organizing map model

Figure 3.1 is an architecture of the model built in this study. The shows that datasets were generated from twain sources which are Openphish and UCI. These datasets have been described previously. The next stage as illustrated in the figure is the data preprocessing phase in which the Openphish categorical dataset were converted to numeric using one-hot encoding and label encoding; the Openphish contained 18 features with 2141 rows/samples while the UCI dataset was numeric and contained 30 features and 11, 055 rows/samples.

4.0. Result and Discussion of Findings

Here is the setup for the experiment for the SOM model developed and the various results derived from the two datasets used. The study used twain datasets from Openphish which was generated in July 2025 but requested by the author of this work in August, 2025 and released in the same month as well as UCI dataset which was generated in 2015 and was released for scholarly use in 2015 but was collected by the author of this work from machine learning repository of UCI (University of California, Irvine).

4.1. Experiment setup

The Openphish and UCI dataset would be compared as the work progresses in this chapter. The two datasets were collected for the experiment in order to verify the variability of results from two or more dataset sources.

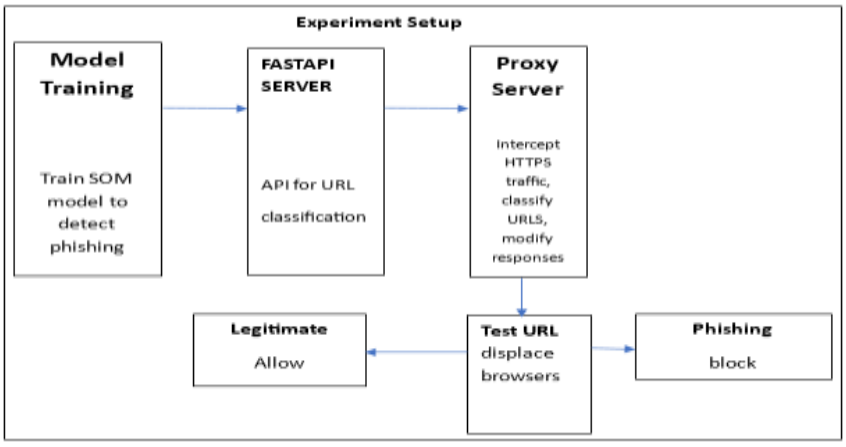


Figure 4.1. SOM model experiment setup

Figure 4.1 reveals the setup for the whole experiment. The experiment starts with training of the datasets from the two sources of Openphish and UCI. Once the data is trained, the artifacts are then fed into FastAPI code which detects which URL is benign or phishing. At this point, a Man in the middle certificate authority software from Microsoft is installed and proxy setting is configured on port 8080. MITM script will control the operation of the proxy server. Once the proxy setting is done manually in the browser setting, all URLs will have to pass through the MITM and the proxy server. Any benign URL would be allowed while a phishing URL will be prevented/blocked from opening.

Server response

Code	Details
200	Response body


```
{
  "result": {
    "timestamp": "2025-09-19T17:09:10.041751Z",
    "url": "string",
    "ip": "0.0.0.0",
    "prediction": "benign",
    "raw_label": 0,
    "bmu": [
      4,
      2
    ],
    "qe": 3.1080953567750456
  },
  "stats": {
    "allowed": 33695,
    "blocked": 14398
  }
}
```


Response headers

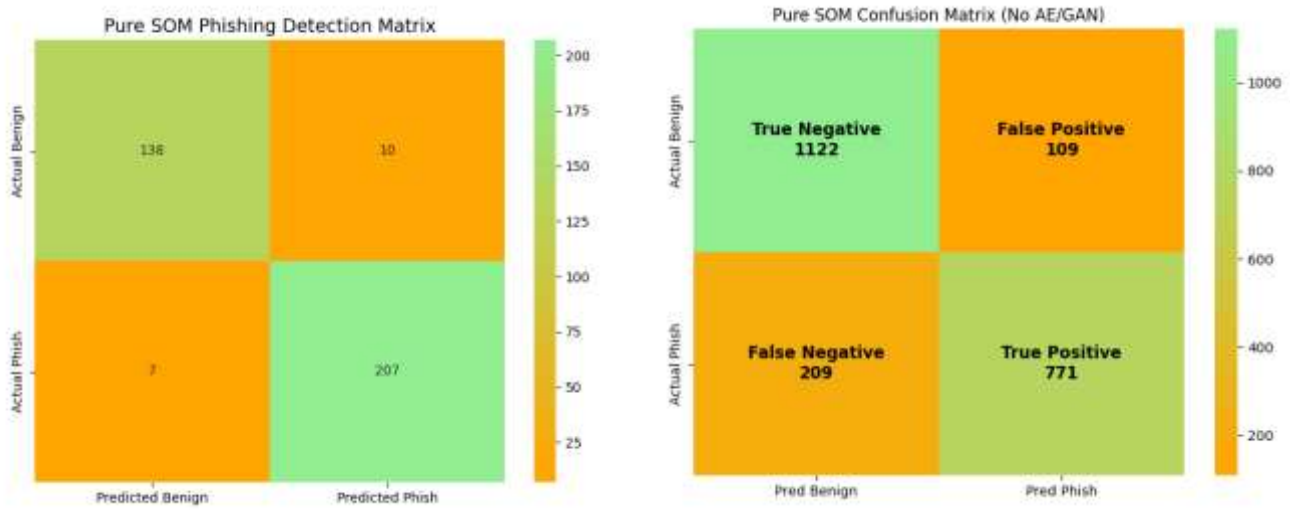

```
content-length: 198
content-type: application/json
date: Fri,19 Sep 2025 17:09:08 GMT
server: uvicorn
```


Figure 4.2. Response from FastAPI interface revealing the performance of the SOM model

Figure 4.2 displays the response of the bashing of the FastAPI script in browser. The FastAPI/Swagger is accessed using "127.0.0.1:8000/docs. Here, single url can be tested for phishing while multiple urls in csv format can be uploaded to the FastAPI and executed while the prediction is made ready for download as a csv file from the FastAPI interface.

4.2. Results and Discussion of Findings

The various results derived from the experiment performed are discussed here. The charts used and their interpretations are done in this section.



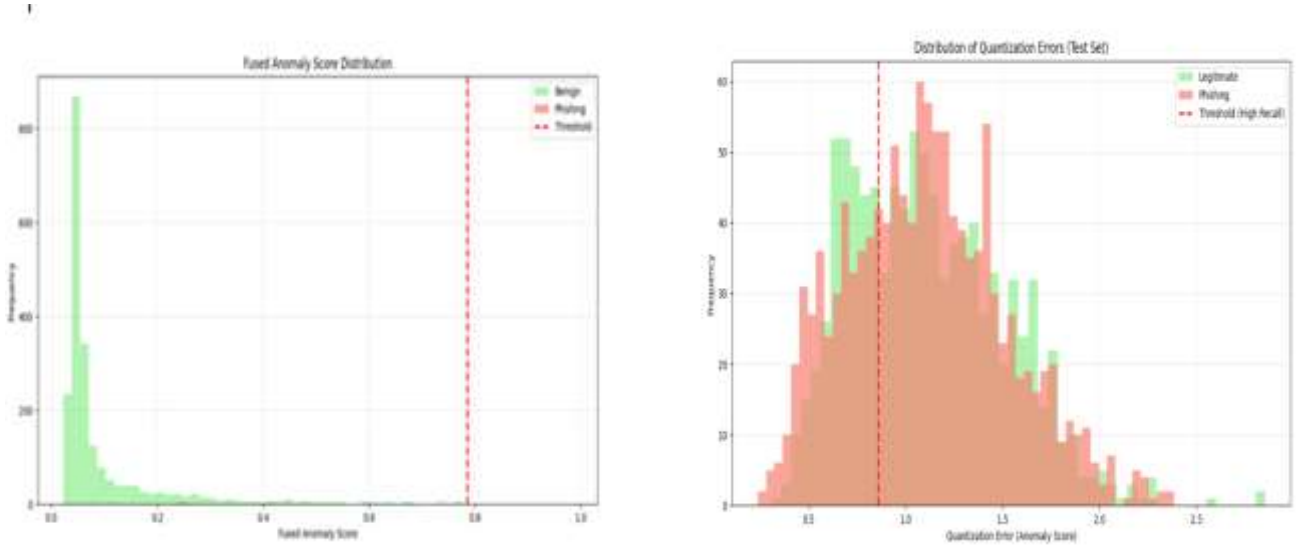
(a) (b)

Figure 4.3. (a) SOM confusion matrix using Openphish dataset, **(b)** SOM confusion matrix using UCI dataset.

Figure 4.3a shows the confusion matrix for both OpenPhish dataset and UCI dataset. True Positive indicates the number of phishing URLs correctly detected (True Positive); False Positive indicates the number of benign URLs incorrectly flagged as phishing (False Positive); while True Negative indicates the number of benign URLs correctly identified (True Negative) and False Negative indicates the number of phishing URLs missed (False Negative). This confusion matrix shows operational behavior of the model at deployment threshold.

Openphish 362 observations were used due to missing values and imbalanced nature of the dataset, originally, the 20% of 2141 is 428. Out of 362 instances eventually used, 138 were true negative which are number of benign URLs correctly identified (True Negative); 30 instances were false positive that were number of legitimate URLs inaccurately flagged as phishing (False Positive); only 7 instances are false negative, it means the number of phishing URLs missed (False Negative) and finally, 207 instances were number of phishing URLs correctly detected (True Positive).

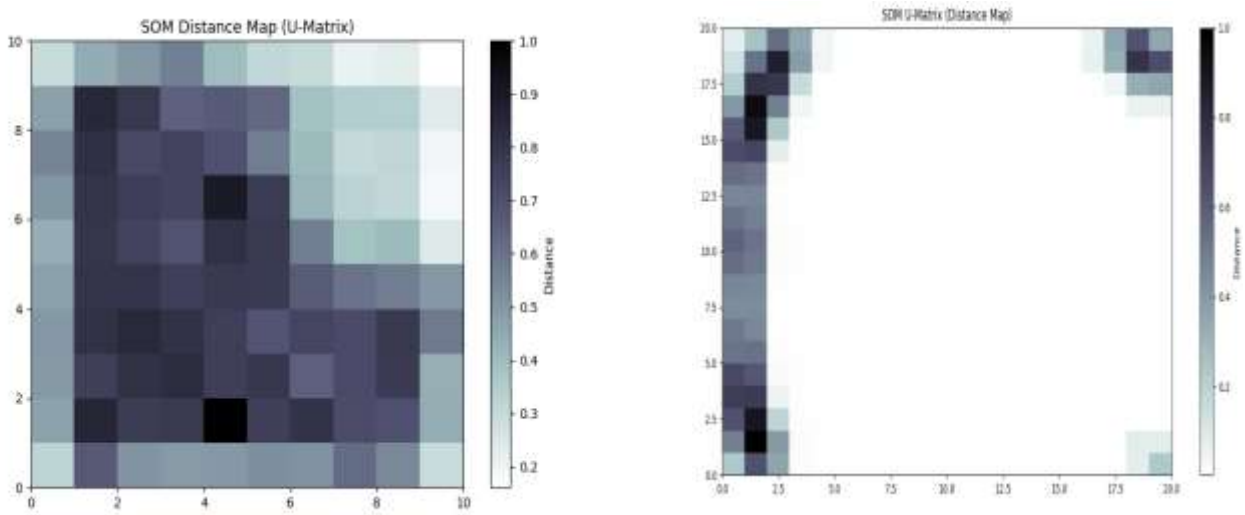
Figure 4.3b is an illustration of confusion matrix for model from UCI dataset. Out of 2211 instances, 1122 were true negative which are number of benign URLs correctly identified (True Negative); 109 instances were false positive which total benign URLs inaccurately flagged as phishing (False Positive); only 209 instances were false negative which means the total phishing URLs missed (False Negative) and finally, 771 instances were total phishing URLs accurately detected (True Positive).



(a) (b)

Figure 4.4. (a) SOM Score distribution using Openphish dataset, **(b)** SOM Score distribution using UCI Dataset.

Figure 4.4a and Figure 4.4b show the graph of both Openphish and UCI dataset anomaly score distribution of self-organizing map. X-axis depicts the normalized anomaly score while the Y-axis shows the density/frequency. The green bars/curves indicate legitimate/benign urls while the pink bars/curves indicate phishing urls. However, the pink vertical line shows that decision threshold t as used in the predictions. With this, the benign urls are separated from the phishing urls. Further interpretation of the chart shows benign urls scores clustered close to zero while phishing scores drifted towards one. With the low scores, we have normal web behavior while with the high scores, we have suspicious lexical patterns.

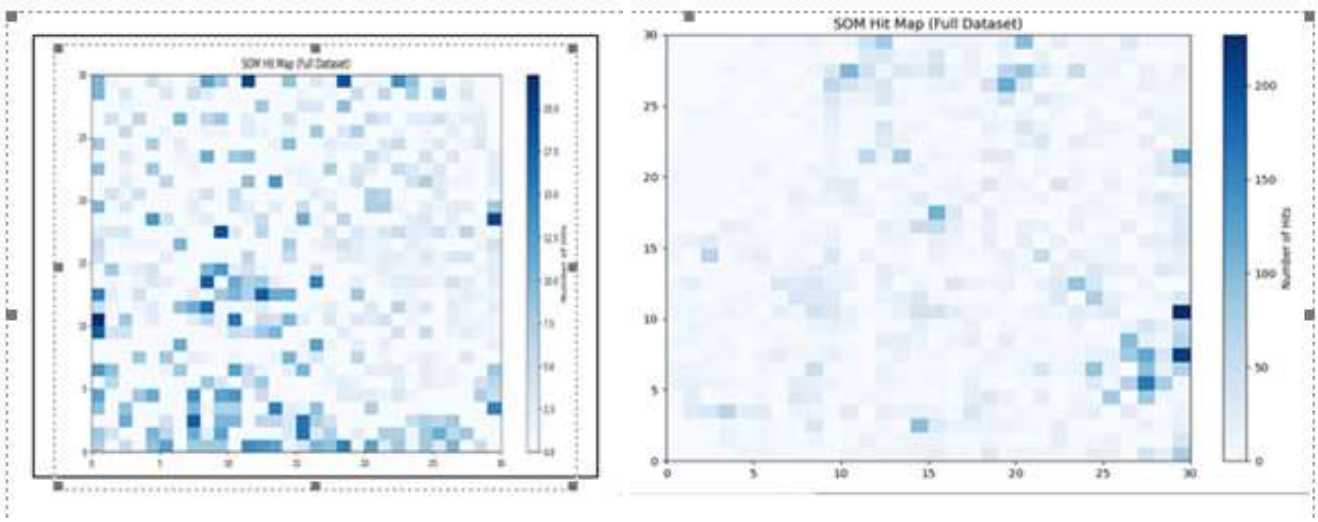


(a) (b)

Figure 4.5. (a) SOM U-Matrix for Openphish Dataset, **(b)** SOM U-Matrix for UCI Dataset.

Figure 4.5a and Figure 4.5b show SOM U-Matrix (Unified Distance Matrix); it shows the distance between neighbouring SOM neurons, where the bright regions are large distances (cluster boundaries) while dark regions

represent homogeneous areas. This represents topological quality validation. The benign clusters are compact, smooth regions while the phishing clusters are near boundaries or isolated zones. In summary, the U-Matrix highlights distinct cluster boundaries, with phishing samples frequently located near high-distance regions, confirming their deviation from normal traffic patterns.



(a) (b)

Figure 4.6. (a) SOM Hit Map for Openphish dataset, **(b)** SOM Hit Map for UCI dataset.

Figure 4.6a and Figure 4.6b show the SOM hit map for both Openphish and UCI datasets.The SOM Hit Map is also known as Activation Frequency Map. The chart shows grid of SOM neurons with each cell showing how often it was activated. The bright blue colors show frequently activated neurons while the dull blue colors show rarely activated neurons. This chart is density estimation and the bright blue colors or high-hit regions show where benign URLs dominate and means normal web behaviour with the dull blue colors or sparse region showing phishing URLs domination. In this case, phishing is rare and scattered.

4.3. Validation of the models from Openphish and UCI datasets

This aspect of the study shows the result from validation of the model from Openphish dataset and UCI dataset.

Table 4.1. Evaluation metrics of Openphish and UCI datasets

Metrics	**Openphish dataset (%)**	**UCI dataset (%)**
Precision	95.00	84.00
Recall	97.00	91.00
F1 Score	94.00	88.00
Accuracy	95.00	86.00

Table 4.1 shows that model from Openphish Dataset with accuracy of 95% outperformed UCI dataset with accuracy of 86%. It is easy to conclude from the table that model from Openphish dataset catches more phishing URLs than model from UCI dataset.

4.4. Comparison of the Model Developed with Other Models in the Literature Review

The outcome of the model built from two different datasets namely Openphish and UCI indicated that the models from Openphish dataset with 428 instances as 20% but 362 were eventually used for the testing due to cleaning as a result of the imbalance of the Openphish dataset and UCI dataset have accuracy of 95% and 86% respectively. This model detected and prevented phishing using SOM.

Our study performed better than 99% of models from other authors reviewed with exception of just the study of Zadeh-Shirazi et al. (2018) left with 1% of the reviewed works. Though the study of Zadeh-Shirazi et al. (2018) achieved similar accuracy of 95% with our study, it is worth noting that our model work in real-time to detect and prevent phishing of financial data whereas their study cannot prevent breast cancer.

5.0. Conclusion

The study did well to use self-organizing map, an artificial neural network model which made the prevention of phishing highly efficient and thus preventing phishing urls in real-time with accuracy of 95%. Phishing is evolving in strategy and pattern; it started with spamming, now to various forms of phishing. Who knows what is up the sleeves of the cybercriminals roaming the cyberworld. Integrating the SOM model with FastAPI and MITM certification authority, detecting and preventing phishing was highly effective. Anti-phishing software and models are very effective but much more effective is user education; internet users need to be educated on phishing strategy and phishing tools and how to use them. Users of internet should be able to identify phishing websites/urls so as to safeguard their financial credentials from being phished.

ML is a key component of contemporary cybersecurity methods because of its capacity to identify minute irregularities, automate reactions, and continuously learn from changing threats. Large amounts of data can be analyzed using machine learning and artificial intelligence algorithms to find trends, abnormalities, and risks that were previously undetected. Using historical data, machine learning models may be trained to identify known threat trends and gradually increase their accuracy. Since SOM is highly efficient for detecting and preventing phishing, it is highly recommended that further study combine SOM with deep learning models to prevent phishing.

Declaration

Source of Funding

This study received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Competing Interests Statement

The authors have declared that no competing financial, professional, or personal interests exist.

Consent for publication

All the authors contributed to the manuscript and consented to the publication of this research work.

Authors' contributions

All the authors took part in literature review, analysis, and manuscript writing equally.

Availability of data and materials

Supplementary information is available from the authors upon reasonable request.

Ethical Approval

Not applicable for this study.

Institutional Review Board Statement

Not applicable for this study.

References

[1] Basnet, R.B., Shash, R., Johnson, C., Walgren, L., & Doleck, T. (2019). Towards detecting and classifying network intrusion traffic using deep learning frameworks. Journal of Internet Services and Information Security, 9(4): 1–17. https://doi.org/10.22667/jisis.2019.11.30.001.

[2] Alkhalil, Z., Hewage, C., Nawaf, L., & Khan, I. (2021). Phishing attacks: A recent comprehensive study and a new anatomy. Frontiers in Computer Science, 3: 563060. https://doi.org/10.3389/fcomp.2021.563060.

[3] Omari, K. (2023). Comparative study of machine learning algorithms for phishing website detection. International Journal of Advanced Computer Science and Applications, 14(9): 417–425. https://doi.org/10.14569/ijacsa.2023.0140945.

[4] Simion, C.A., Gavrilut, D.T., & Luchian, H. (2019). An adversarial machine learning approach to evaluate the robustness of a security solution. In Proceedings of the 21st International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2019).

[5] Kamal, O. (2023). Comparative study of machine learning algorithms for phishing website detection. International Journal of Advanced Computer Science and Applications, 14(9): 417–423. https://doi.org/10.14569/ijacsa.2023.0140945.

[6] Rao, R.S., & Ali, S.T. (2015). PhishShield: A desktop application to detect phishing webpages through heuristic approach. Procedia Computer Science, 54: 147–156. https://doi.org/10.1016/j.procs.2015.06.017.

[7] Pei, L., Ghimire, M., Byanjankar, N., Joshi, M., Mushtaq, N., Ji, M., & Joshi, T.P. (2022). Impacts of land use on surface water quality using self-organizing map in middle region of the Yellow River Basin, China. Water, 14(17): 2736. https://doi.org/10.3390/w14172736.

[8] Ahmed, A.A., & Abdullah, N.A. (2016). Real time detection of phishing websites. In 2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON): 1–6, IEEE. https://doi.org/10.1109/iemcon.2016.7746247.

[9] Sarker, I.H. (2020). Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2(3): 160. https://doi.org/10.1007/s42979-021-00592-x.

[10] Kaur, A., & Mian, S.M. (2023). A review on phishing technique: Classification, lifecycle and detection approaches. In 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Pages 336–339, IEEE. https://doi.org/10.1109/icacite57410.2023.10182964.

[11] Kowsher, M.D., Tahabilder, A., Sanjid, Z.I., Prottasha, N.J., Uddin, S., Hossain, A., & Jilani, F.A.K. (2021). LSTM-ANN & BiLSTM-ANN: Hybrid deep learning models for enhanced classification accuracy. Procedia Computer Science, 193: 131–140. https://doi.org/10.1016/j.procs.2021.10.014.

[12] Ahmad, S.A., Mohammed, B.K., Fqi, K.O., Karim, F.R., & Rafiq, S.K. (2024). Soft computing modeling including artificial neural network, non-linear, and linear regression models to predict the compressive strength of sustainable mortar modified with palm oil fuel ash. Construction, 4(1): 52–67. https://doi.org/10.15282/construction.v4i1.10209.

[13] Gugnani, G., Rawat, A., Shastri, M., & Kumar, P. (2015). Anomaly recognition in online social networks. International Journal of Security and Its Applications, 9(7): 109–118. https://doi.org/10.14257/ijsia.2015.9.7.10.

[14] Back, S., & Guerette, R.T. (2021). Cyber place management and crime prevention: The effectiveness of cybersecurity awareness training against phishing attacks. Journal of Contemporary Criminal Justice, 37(3): 427–451. https://doi.org/10.1177/10439862211001628.

[15] Ansari, M.F., Sharma, P.K., & Dash, B. (2022). Prevention of phishing attacks using AI-based cyber security awareness training. International Journal of Smart Sensor and Adhoc Network, 3(3): 61–72. https://doi.org/10.1109/odicon54453.2022.10010185.

[16] Shahrivari, V., Darabi, M.M., & Izadi, M. (2020). Phishing detection using machine learning techniques. ArXiv. https://doi.org/10.48550/arxiv.2009.11116.

[17] Alshantti, A., & Rasheed, A. (2021). Self-organising map-based framework for investigating accounts suspected of money laundering. Frontiers in Artificial Intelligence, 4: 761925. https://doi.org/10.3389/frai.2021.761925.

[18] Breskuvienė, D., & Dzemyda, G. (2024). Enhancing credit card fraud detection: Highly imbalanced data case. Journal of Big Data, 11: 182. https://doi.org/10.1186/s40537-024-01059-5.

[19] Anjana, U., & Divyashri, C. (2026). Artificial intelligence-enhanced weighted self-organizing map for accurate weather forecasting and crop prediction in agriculture. International Journal of Engineering Research & Technology, 15(1).

[20] Meyer, S.G., Reading, A.M., & Bassom, A.P. (2022). The use of weighted self-organizing maps to interrogate large seismic data sets. Geophysical Journal International, 231(3): 2156–2172. https://doi.org/10.1093/gji/ggac322.

[21] Deokar, S., Khandare, P., & Dixit, A. (2020). Improvement of traditional protection system in the existing hybrid microgrid with advanced intelligent method. International Journal of Data Science, 1(2): 72–81. https://doi.org/10.53188/ijds.2020.12.72.

[22] Asker, N.S., & Essa, E.I. (2024). A review of social engineering attack detection based on machine learning techniques. Nanotechnology Perceptions, 20(s2): 636–641.

[23] Eyassu, S., & Gambäck, B. (2005). Classifying Amharic news text using self-organizing maps. In Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages: 71–78, Association for Computational Linguistics. https://doi.org/10.3115/1621787.1621801.

[24] Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., & Harshman, R. (1990). Indexing by latent semantic analysis. Journal of the American Society for Information Science, 41(6): 391–407. https://doi.org/10.1002/(sici)1097-4571(199009)41:6.

[25] Ličen, S., Di-Gilio, A., Palmisani, J., Petraccone, S., de-Gennaro, G., & Barbieri, P. (2020). Pattern recognition and anomaly detection by self-organizing maps in a multi month e-nose survey at an industrial site. Sensors, 20(7): 1887. https://doi.org/10.3390/s20071887.

[26] Saarikoski, J., Järvelin, K., Laurikkala, J., & Juhola, M. (2009). On document classification with self-organising maps. In International Conference on Adaptive and Natural Computing Algorithms, Pages 140–149, Springer. https://doi.org/10.1007/978-3-642-04921-7_15.

[27] Saarikoski, J., Laurikkala, J., Järvelin, K., & Juhola, M. (2011). Self-organising maps in document classification: A comparison with six machine learning methods. In International Conference on Adaptive and Natural Computing Algorithms: 260–269, Springer. https://doi.org/10.1007/978-3-642-20282-7_27.

[28] Khacef, L., Rodriguez, L., & Miramond, B. (2020). Improving self-organizing maps with unsupervised feature extraction. In Yang, H., Pasupa, K., Leung, A.C.-S., Kwok, J.T., Chan, J.H., & King, I. (Eds.), Neural Information Processing: ICONIP 2020, Pages 474–486, Springer. https://doi.org/10.1007/978-3-030-63833-7_40.

[29] Sandoval-Lara, Z., Gómez-Gil, P., Moreno-Rodríguez, J.C., & Ramírez-Cortés, M. (2023). Self-organizing clustering by Growing-SOM for EEG-based biometrics. In 2023 International Conference on Artificial Intelligence and Applications (ICAIA), Pages 1–4, IEEE. https://doi.org/10.1109/icaiaatcon60371.2023.10551669.

[30] Fan, X., Zhang, S., Xue, X., Jiang, R., Fan, S., & Kou, H. (2025). An improved self-organizing map (SOM) based on virtual winning neurons. Symmetry, 17(3): 449. https://doi.org/10.3390/sym17030449.

[31] Cui, W., Xue, W., Li, L., & Shi, J.A. (2020). Method for intermittent fault diagnosis of electronic equipment based on labeled SOM. In 2020 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), Pages 149–154, IEEE. https://doi.org/10.1109/sdpc49795.2020.9352908.

[32] Zhang, X., Xu, J., & Zhang, H. (2023). Fault diagnosis of levitation controller of medium-speed maglev train based on SOM-BP neural network. In Proceedings of the 2023 35th Chinese Control and Decision Conference (CCDC), Pages 4162–4167, IEEE. https://doi.org/10.1109/ccdc58219.2023.10326703.

[33] Iftikhar, N., Rehman, M.U., Shah, M.A., Alenazi, M.J.F., & Ali, J. (2025). Intrusion detection in NSL-KDD dataset using hybrid self-organizing map model. Computer Modeling in Engineering & Sciences, 143(1): 639–671. https://doi.org/10.32604/cmes.2025.062788.

[34] Zadeh-Shirazi, A., Seyyed Mahdavi Chabok, S.J., & Mohammadi, Z. (2018). A novel and reliable computational intelligence system for breast cancer detection. Medical & Biological Engineering & Computing, 56(5): 721–732. https://doi.org/10.1007/s11517-017-1721-z.

[35] Chen, W., An, N., Jiang, M., & Jia, L. (2024). An improved deep temporal convolutional network for new energy stock index prediction. Information Sciences, 682: 121244. https://doi.org/10.1016/j.ins.2024.121244.