International Journal of Scientific Research in Science and Technology
Available online at : www.ijsrst.com
Print ISSN: 2395-6011 | Online ISSN: 2395-602X
doi : https://doi.org/10.32628/IJSRST251368

# Predictive Maintenance in Leveraging Supervised Machine Learning For Wireless Network Attacks

**Vidhya Vivek[1], Dr. J. Jaya Priya[2]**

[1]PG Scholar, Department of A Computer Science Engineering Karpaga Vinayaga College of Engineering & Technology Chinna Kolambakkam, Madhuranthagam(TK), Chengalpattu (DT), PIN 603 308, Tamil Nadu, India

[2]Associate Professor, Department of Computer Science Engineering Karpaga Vinayaga College of Engineering & Technology, Chinna Kolambakkam, Madhuranthagam(TK), Chengalpattu (DT), PIN 603 308, Tamil Nadu, India

ARTICLEINFO

**Article History:**

Accepted : 01 July 2025
Published: 03 August 2025

**Publication Issue :**
Volume 12, Issue 4
July-August-2025

**Page Number :**
872-880

ABSTRACT

Predictive maintenance has emerged as a crucial strategy for ensuring the reliability and efficiency of wireless networks amidst growing cybersecurity threats. This study explores the application of supervised machine learning techniques in predictive maintenance specifically tailored for detecting and mitigating attacks on wireless networks. Leveraging historical network data encompassing various network parameters and security incidents, a predictive model is developed to forecast potential network attacks. The supervised learning approach involves training the model on labeled datasets, where instances of network attacks are explicitly identified. Through feature engineering and selection, relevant network features are extracted to enhance the model's predictive capabilities. The trained model is then deployed to continuously monitor network traffic in real- time, identifying anomalous patterns indicative of potential attacks. Early detection of such threats enables proactive measures to be taken, including network reconfiguration, traffic filtering, and incident response, thus minimizing the impact of cyber attacks and ensuring uninterrupted network operations. The proposed predictive maintenance framework offers a proactive and adaptive approach to network security management, enhancing the resilience of wireless networks against evolving cyber security threats.

**Keywords-**Predictive Maintenance, Supervised Learning, Wireless Networks, Anomaly Detection, Machine Learning, Network Security, Django Framework.

## I. INTRODUCTION

In the realm of wireless network security, leveraging machine learning techniques has become pivotal for detecting and mitigating various types of attacks. This process begins with data preprocessing, where raw network traffic data is cleaned and transformed to ensure quality inputs for model training. Following this, data visualization plays a crucial role in understanding the underlying patterns and anomalies within the dataset, allowing for informed feature selection and engineering. To enhance attack detection accuracy, three different machine learning algorithms are compared, assessing their performance based on metrics like accuracy, precision, and recall. Finally, integrating these machine learning models with a Django framework facilitates the development of a user-friendly web application that can dynamically analyze network traffic and present insights, thereby empowering administrators to respond swiftly to potential threats.

### 1.1 Data Science

Data science is an interdisciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from structured and unstructured data, and apply knowledge and actionable insights from data across a broad range of application domains.

The term "data science" has been traced back to 1974, when Peter Naur proposed it as an alternative name for computer science. In 1996, the International Federation of Classification Societies became the first conference to specifically feature data science as a topic. However, the definition was still in flux.

The term “data science” was first coined in 2008 by D.J. Patil, and Jeff Hammerbacher, the pioneer leads of data and analytics efforts at LinkedIn and Facebook. In less than a decade, it has become one of the hottest and most trending professions in the market.

Data science is the field of study that combines domain expertise, programming skills, and knowledge of mathematics and statistics to extract meaningful insights from data.

Data science can be defined as a blend of mathematics, business acumen, tools, algorithms and machine learning techniques, all of which help us in finding out the hidden insights or patterns from raw data which can be of major use in the formation of big business decisions.

Data Scientist:

Data scientists examine which questions need answering and where to find the related data. They have business acumen and analytical skills as well as the ability to mine, clean, and present data. Businesses use data scientists to source, manage, and analyze large amounts of unstructured data.

Required Skills for a Data Scientist:

- Programming: Python, SQL, Scala, Java, R, MATLAB.
- Machine Learning: Natural Language Processing, Classification, Clustering.
- Data Visualization: Tableau, SAS, D3.js, Python, Java, R libraries.
- Big data platforms: MongoDB, Oracle, Microsoft Azure, Cloudera.

### 1.2 ARTIFICIAL INTELLIGENCE

Artificial intelligence (AI) refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. The term may also be applied to any machine that exhibits traits associated with a human mind such as learning and problem-solving.

Artificial intelligence (AI) is intelligence demonstrated by machines, as opposed to the natural intelligence displayed by humans or animals. Leading AI textbooks define the field as the study of "intelligent agents" any system that perceives its environment and takes actions that maximize its chance of achieving its goals. Some popular accounts use the term "artificial intelligence" to describe machines that mimic "cognitive" functions that humans associate with the human mind, such as

"learning" and "problem solving", however this definition is rejected by major AI researchers.

Artificial intelligence is the simulation of human intelligence processes by machines, especially computer systems. Specific applications of AI include expert systems, natural language processing, and speech recognition and machine vision.

AI applications include advanced web search engines, recommendation systems (used by Youtube, Amazon and Netflix), Understanding human speech (such as Siri or Alexa), self-driving cars (e.g. Tesla), and competing at the highest level in strategic game systems (such as chess and Go), As machines become increasingly capable, tasks considered to require "intelligence" are often removed from the definition of AI, a phenomenon known as the AI effect. For instance, optical character recognition is frequently excluded from things considered to be AI, having become a routine technology.

Artificial intelligence was founded as an academic discipline in 1956, and in the years since has experienced several waves of optimism, followed by disappointment and the loss of funding (known as an "AI winter"), followed by new approaches, success and renewed funding. AI research has tried and discarded many different approaches during its lifetime, including simulating the brain, modeling human problem solving, formal logic, large databases of knowledge and imitating animal behavior. In the first decades of the 21st century, highly mathematical statistical machine learning has dominated the field, and this technique has proved highly successful, helping to solve many challenging problems throughout industry and academia.

The various sub-fields of AI research are centered on particular goals and the use of particular tools. The traditional goals of AI research include reasoning, knowledge representation, planning, learning, natural language processing, perception and the ability to move and manipulate objects. General intelligence (the ability to solve an arbitrary problem) is among the field's long-term goals. To solve these problems, AI researchers use versions of search and mathematical optimization, formal logic, artificial neural networks, and methods based on statistics, probability and economics. AI also draws upon computer science, psychology, linguistics, philosophy, and many other fields.

The field was founded on the assumption that human intelligence "can be so precisely described that a machine can be made to simulate it". This raises philosophical arguments about the mind and the ethics of creating artificial beings endowed with human-like intelligence. These issues have been explored by myth, fiction and philosophy since antiquity. Science fiction and futurology have also suggested that, with its enormous potential and power, AI may become an existential risk to humanity.

As the hype around AI has accelerated, vendors have been scrambling to promote how their products and services use AI. Often what they refer to as AI is simply one component of AI, such as machine learning. AI requires a foundation of specialized hardware and software for writing and training machine learning algorithms. No one programming language is synonymous with AI, but a few, including Python, R and Java, are popular.

In general, AI systems work by ingesting large amounts of labeled training data, analyzing the data for correlations and patterns, and using these patterns to make predictions about future states. In this way, a chatbot that is fed examples of text chats can learn to produce life like exchanges with people, or an image recognition tool can learn to identify and describe objects in images by reviewing millions of examples.

AI programming focuses on three cognitive skills: learning, reasoning and self-correction.

Learning processes. This aspect of AI programming focuses on acquiring data and creating rules for how to turn the data into actionable information. The rules, which are called algorithms,

provide computing devices with step-by-step instructions for how to complete a specific task.

Reasoning processes. This aspect of AI programming focuses on choosing the right algorithm to reach a desired outcome.

Self-correction processes. This aspect of AI programming is designed to continually fine-tune algorithms and ensure they provide the most accurate results possible.

AI is important because it can give enterprises insights into their operations that they may not have been aware of previously and because, in some cases, AI can perform tasks better than humans. Particularly when it comes to repetitive, detail-oriented tasks like analyzing large numbers of legal documents to ensure relevant fields are filled in properly, AI tools often complete jobs quickly and with relatively few errors.

Artificial neural networks and deep learning artificial intelligence technologies are quickly evolving, primarily because AI processes large amounts of data much faster and makes predictions more accurately than humanly possible.

Natural Language Processing (NLP):

Natural language processing (NLP) allows machines to read and understand human language. A sufficiently powerful natural language processing system would enable natural-language user interfaces and the acquisition of knowledge directly from human-written sources, such as newswire texts. Some straightforward applications of natural language processing include information retrieval, text mining, question answering and machine translation. Many current approaches use word co- occurrence frequencies to construct syntactic representations of text. "Keyword spotting" strategies for search are popular and scalable but dumb; a search query for "dog" might only match documents with the literal word "dog" and miss a document with the word "poodle". "Lexical affinity" strategies use the occurrence of words such as "accident" to assess the sentiment of a document. Modern statistical NLP approaches can combine all these strategies as well as others, and often achieve acceptable accuracy at the page or paragraph level. Beyond semantic NLP, the ultimate goal of "narrative" NLP is to embody a full understanding of common sense reasoning. By 2019, transformer-based deep learning architectures could generate coherent text.

## II. RELATED WORK

Over the past decade, there has been significant research into integrating machine learning with cybersecurity systems, particularly for wireless network security. The need for predictive and proactive defense strategies has led to the application of anomaly detection, classification models, and ensemble learning techniques.

Zhou et al. [1] explored the impact of adversarial attacks on mining algorithms used in multi-network environments. They proposed a framework to detect and mitigate adversarial manipulations by analyzing structural perturbations in graph-based data, which are commonly encountered in wireless sensor networks (WSNs). Their work highlighted the importance of securing learning pipelines against manipulation—relevant to this study's goal of resilient prediction.

Breier et al. [2] introduced FooBaR, a fault-injection-based backdoor attack that compromises neural network training in edge devices and IoT environments. Their findings emphasized the critical nature of safeguarding the training process, which complements the supervised learning approach in our system by justifying the use of robust, interpretable algorithms like Extra Trees.

In their 2023 study, Houda et al. [3] proposed MiTFed, a federated learning-based attack mitigation framework that combines Software-Defined Networking (SDN) and blockchain technologies. Although effective in decentralized threat intelligence sharing, it lacks a centralized dashboard for real-time visibility, which our Django-integrated system addresses.

Other works have focused on enhancing intrusion detection systems (IDS) through deep learning and feature optimization. For instance, Alazab et al. applied hybrid feature selection to optimize detection performance in network intrusion systems using random forests and support vector machines (SVMs). Similarly, Kumar et al. proposed an ensemble model combining CNNs and LSTMs to capture both spatial and temporal patterns of cyberattacks in smart grids.

However, while existing models offer high accuracy in offline evaluations, few systems have been translated into **live, web-deployable solutions** for proactive security monitoring. This research bridges that gap by integrating model training, evaluation, and deployment into a unified system capable of detecting and mitigating wireless network attacks in real-time.

Furthermore, unlike most prior work that relies heavily on deep neural networks—which are complex, resource- intensive, and prone to adversarial attacks—this study leverages **interpretable, efficient, and ensemble-based supervised learning algorithms** suitable for deployment in bandwidth- and resource-constrained wireless environments.

## III.SYSTEM DESIGN AND ARCHITECTURE

The proposed system follows a modular architecture designed to provide a seamless workflow from **data ingestion to live attack prediction and visualization**. It consists of six interconnected layers: **Data Collection**, **Preprocessing**, **Modeling**, **Evaluation**, **Deployment**, and **Dashboard Interface**. Each module is developed with scalability, efficiency, and real-time responsiveness in mind.

A. Overview of Architecture

The architecture is divided into two main phases:

1. **Model Development Phase** (Offline): Involves historical data processing, feature extraction, model training, and validation.
2. **Model Deployment Phase** (Online): Involves real-time prediction using the best-performing model integrated into a Django-based dashboard.

The design follows a pipeline structure, enabling the flow of network traffic data through several analytical stages before outputting predictive results and actionable alerts.

B. Module Descriptions

1) Data Collection Layer

- **Function:** Captures wireless traffic logs from various network devices or existing datasets.
- **Details:** Includes features like time stamp, node behavior (Rank, CH, Join_CH), routing attributes, and energy consumption.
- **Tools Used:** Packet sniffers, CSV data logs, and network monitoring APIs.

2) Preprocessing and Cleaning Layer

- **Function:** Prepares raw data for machine learning.
- ***Tasks:***
  - Missing value handling
  - Label encoding for categorical fields
  - Duplicate removal
  - Data balancing using ***RandomOverSampler***
  - Normalization and standard scaling

This layer ensures the dataset is consistent, balanced, and suitable for training.

3) Feature Engineering Layer

- **Function:** Selects and transforms relevant features that improve model performance.
- ***Key Features:***
  - Expaned_Energy (Power usage pattern)
  - send_code, ADV_R, Join_CH (Control messages)
  - Rank and who_CH (Routing and hierarchy)

Pearson correlation and domain analysis are applied to reduce dimensionality and eliminate noisy or redundant features.

4) Model Training Layer

- **Function:** Trains supervised machine learning models.
- ***Algorithms Used:***

- **Complement Naive Bayes**: Fast, suitable for skewed or sparse data
- **AdaBoost Classifier**: Adaptive boosting with decision stumps
- **Extra Trees Classifier**: High variance ensemble method with low overfitting

- ***Training Setup:***
  - 80/20 data split
  - Grid search for hyperparameter tuning
  - Cross-validation for robustness

5) Model Evaluation and Selection Layer

- **Function:** Compares performance across models and selects the most accurate.
- ***Metrics Used:***
  - Accuracy
  - Precision
  - Recall
  - F1-Score
  - Confusion Matrix
  - Hamming Loss
- **Result:** Extra Trees achieved the highest accuracy and F1-score, making it the final choice for deployment.

6) Web-Based Deployment and Visualization Layer

- **Platform:** Django framework (Python-based web backend)
- ***Features:***
  - Real-time upload of test logs or streaming input
  - Background model inference engine
  - Visual feedback with predictions (Safe / Threat types)
  - Live alert generation with logs and timestamps

Users can access the system via a secure web interface to monitor network status and receive warnings in real-time.

7) Architeture Diagram

The proposed system architecture for wireless network attack prediction is depicted in diagram, The framework integrates data preprocessing, machine learning model development, and a user-centric web interface for real- time prediction. The major components are detailed below:

A. Dataset Acquisition

The raw dataset is acquired from the Kaggle platform, which includes labeled network traffic instances representing various attacks such as DoS, Probe, R2L, and U2R. These datasets serve as the foundation for model training and evaluation.

B. Data Preprocessing and Visualization

Data preprocessing is performed using Python libraries such as Pandas and NumPy to clean, transform, and encode the dataset. Simultaneously, exploratory data analysis is conducted using visualization tools such as Matplotlib and Seaborn to uncover relationships and patterns among the features.

C. Model Development

The system employs supervised learning algorithms including Naive Bayes, AdaBoost, and Extra Trees Classifier. The model offering the highest classification accuracy is serialized into .pkl format using Python's joblib or pickle module for subsequent deployment.

D. Web Application Integration

The web application is developed using the Django framework. The backend is implemented in Python to handle model inference and communication with the database. The frontend is designed using HTML, CSS, and JavaScript, ensuring a responsive and user-friendly interface.

E. User Interface Workflow

The application consists of multiple pages, including:

- **Landing Page**: Introduction to the system.
- **Register/Login Page**: User authentication and session management.
- **Home Page**: Dashboard navigation.
- **Input Page**: Accepts user-submitted network data.
- **Prediction Page**: Displays the result of the prediction.

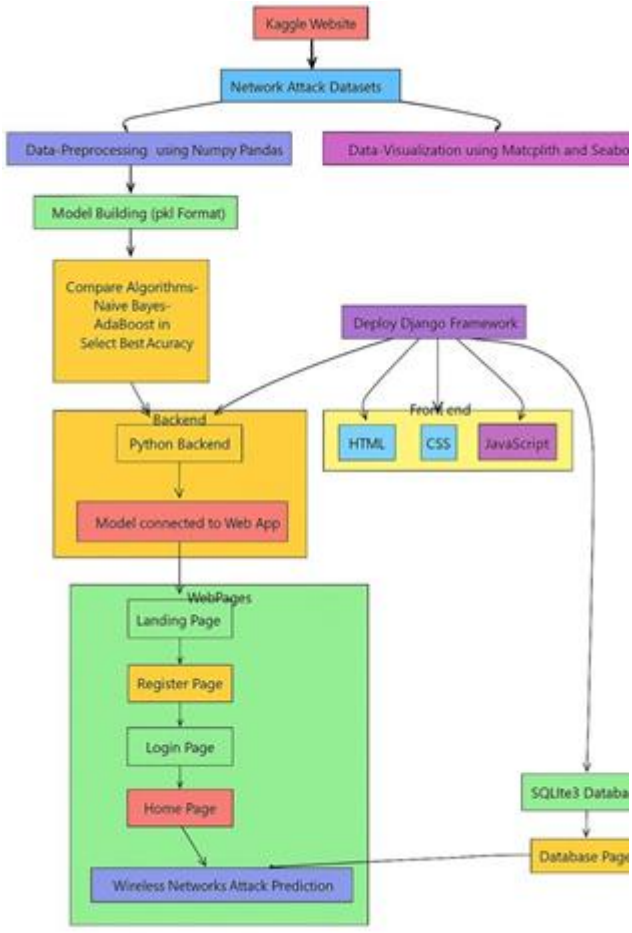
Each page interacts with the backend where the pre-trained model is invoked to analyze input and return results to the frontend.

F. Database Integration

An SQLite3 database is integrated to maintain user credentials, activity logs, and historical prediction records. An admin-accessible **Database Page** allows the review of stored results and ensures centralized monitoring of usage and alerts.

G. Prediction Output

Upon submission of network data through the input page, the backend triggers the pre-trained model and returns the prediction. The system identifies whether the traffic is **benign** or **malicious**, along with the specific attack type if applicable.




## IV. CONCLUSION AND FUTURE WORK

In the study of wireless network attacks using machine learning, we began with thorough data preprocessing to ensure data quality and relevance, which included handling missing values, normalizing features, and encoding categorical variables. Data visualization techniques were employed to uncover patterns and insights, revealing the distinct characteristics of various attack types. We then compared the performance of three different algorithms assessing their accuracy, precision, and recall metrics. The Random Forest algorithm consistently outperformed the others, demonstrating superior robustness and reliability in attack classification. Finally, we integrated our machine learning model with the Django framework, allowing for seamless deployment and monitoring of network security, thus providing an effective solution for detecting and mitigating wireless network attacks.

## V. FUTURE WORK

- Enhancing threat detection by integrating advanced machine learning algorithms to improve accuracy and reduce false positives in wireless network security systems.
- Developing adaptive models that can learn from evolving attack patterns and dynamically update defenses to mitigate emerging wireless network threats.

## REFERENCES

[1]. Adesina, D., Hsieh, C.-C., Sagduyu, Y. E., & Qian, L., "Adversarial Machine Learning in Wireless Communications Using RF Data: A Review," IEEE Commun. Surveys & Tutorials, 2022 arXiv+1ACM Digital Library+1.

[2]. Sagduyu, Y. E., Shi, Y., & Erpek, T., "Adversarial Machine Learning for 5G Communications Security," preprint (to appear IEEE), 2021 arXiv.

[3]. Kim, B., Sagduyu, Y. E., Davaslioglu, K., & Ulukus, S., "Channel-Aware Adversarial Attacks Against Deep Learning-Based Wireless Signal

Classifiers," IEEE Global Communications Conference (GLOBECOM), 2021 arXiv.

[4]. Shi, Y., Davaslioglu, K., & Sagduyu, Y. E., "Generative Adversarial Network in the Air: Deep Adversarial Learning for Wireless Signal Spoofing," preprint (IEEE submission), 2020 arXiv.

[5]. Ahmad, R., Wazirali, R., & Abu-Ain, T., "Machine Learning for Wireless Sensor Networks Security: An Overview of Challenges and Issues," Sensors, 2022 mdpi.com.

[6]. Reyes, A. A., Vaca, F. D., Castro Aguayo, G. A., Niyaz, Q., & Devabhaktuni, V., "A Machine Learning Based Two-Stage Wi-Fi Network Intrusion Detection System," Electronics, vol. 9, no. 10, 2020 mdpi.com.

[7]. Sood, T., Prakash, S., Sharma, S., Singh, A. & Choubey, H., "Intrusion Detection System in Wireless Sensor Network Using Conditional GAN," in Wireless Personal Communications, 2022 link.springer.com.

[8]. Islabudeen, M. & M. K. Kavitha Devi, "Smart Approach for Intrusion Detection and Prevention in Mobile Ad Hoc Networks," Wireless Personal Communications, 2020 link.springer.com.

[9]. Sinha, S. & Paul, A., "Neuro-fuzzy Based Intrusion Detection System for Wireless Sensor Networks," Wireless Personal Communications, 2020 link.springer.com.

[10]. Rajadurai, H. & U. D. Gandhi, "Stacked Ensemble Learning Model for Intrusion Detection in Wireless Network," Neural Computing and Applications, 2020 link.springer.com.

[11]. Tama, B. A. & Lim, S., "Ensemble Learning for Intrusion Detection Systems: A Systematic Mapping and Cross-Benchmark Evaluation," Computer Science Review, 2021 link.springer.com.

[12]. D'hooge, L., Wauters, T., Volckaert, B., & Turck, F. D., "Inter-dataset Generalization Strength of Supervised ML Methods for Intrusion Detection," J. Information Security Applications, 2020 link.springer.com.

[13]. Singh, A., Amutha, J., Nagar, J., Sharma, S., & Lee, C. C., "AutoML-ID: Automated Machine Learning Model for Intrusion Detection Using Wireless Sensor Network," Sci. Reports, vol. 12, 2022 link.springer.com.

[14]. Subbiah, S., Anbananthen, K. S. M., Thangaraj, S., et al., "Grid-Search Random Forest with Boruta Feature Selection for WSN Intrusion Detection," J. Commun. Netw., 2022 link.springer.com.

[15]. Niknami, N. & Wu, J., "Advanced ML/DL-Based Intrusion Detection Systems for Software-Defined Networks," in Network Security Empowered by AI, Springer, 2024 link.springer.com.

[16]. Aminanto, M. E., Choi, R., Tanuwidjaja, H. C., Yoo, P. D., & Kim, K., "Lightweight Real-Time WiFi-Based Intrusion Detection Using LightGBM," Wireless Networks, 2023 ACM Digital Library.

[17]. Kolias, C., Kambourakis, G., Stavrou, A., & Gritzalis, S., "Intrusion Detection in 802.11 Networks: Empirical Evaluation & Dataset," IEEE Commun. Surveys & Tutorials (update 2021) ACM Digital Librarylink.springer.com.

[18]. Restuccia, F., et al., "Hacking the Waveform: Generalized Wireless Adversarial Deep Learning," arXiv preprint (to appear IEEE), 2020 ACM Digital Library.

[19]. Flowers, B., Buehrer, R. M. & Headley, W. C., "Evaluating Adversarial Evasion Attacks in Wireless Communications," IEEE Trans. Inf. Forensics Security, vol. 15, 2020 ACM Digital Library.

[20]. Zhong, C., Wang, F., Gursoy, M. C., & Velipasalar, S., "Adversarial Jamming Attacks on DRL-Based Dynamic Multichannel Access," in IEEE WCNC, 2020 ieee-wcnc.netACM Digital Library.

[21]. Wang, F., Zhong, C., Gursoy, M. C., & Velipasalar, S., "Adversarial Jamming Attacks and Defense via Adaptive Deep Reinforcement Learning," arXiv 2020 ACM Digital Library.

[22]. Sadeghi, M. F. & Larsson, E. G., "Physical Adversarial Attacks Against Autoencoder Communication Systems," IEEE Commun. Lett., vol. 23, no. 5, pp. 847–850, 2020 ACM Digital Library.

[23]. Dewal, P., Narula, G. S., Jain, V., & Baliyan, A., "Security Attacks in Wireless Sensor Networks: A Survey," in Cyber Security, Springer, 2020 ACM Digital Librarylink.springer.com.

[24]. Parashar, M., Poonia, A., & Satish, K., "A Survey of Attacks and Mitigations in SDN," Proc. ICCCNT, 2019 (updated surveys cited in 2021) ACM Digital Library.

[25]. Kim, B., Sagduyu, Y. E., Erpek, T., & Davaslioglu, K., "Channel Effects on Surrogate Models of Adversarial Attacks Against Wireless Signal Classifiers," Proc. IEEE ICC, 2021 ACM Digital Library.