REPUBLIC OF TÜRKİYE
ALTINBAŞ UNIVERSITY
Institute of Graduate Studies

Information Technologies

INTRUSION DETECTION IN WIRELESS SENSOR NETWORKS

Intisar Jumaah Jameel JAMEEL

Master's Thesis

Supervisor

Asst. Prof. Dr. Ayça KURNAZ TÜRKBEN

İstanbul, 2024

INTRUSION DETECTION IN WIRELESS SENSOR NETWORKS

Intisar Jumaah Jameel JAMEEL

Information Technologies

Master's Thesis

ALTINBAŞ UNIVERSITY

2024

The thesis/dissertation titled INTRUSION DETECTION IN WIRELESS SENSOR NETWORKS prepared by INTISAR JUMAAH JAMIL and submitted on 00/12/2023 has been **accepted unanimously** for the degree of Master of science in Information Technologies

.

Asst. Prof. Dr. Ayça Kurnaz TÜRKBEN

Supervisor

Thesis Defense Committee Members:

Asst. Prof. Dr. Ayça Kurnaz TÜRKBEN	Software Engineering, Altinbaş University	_____
Asst. Prof. Dr. Abdullahi Abdu IBRAHIM	Computer Engineering, Altinbaş University	_____
Asst. Prof. Dr. Zeynep ALTAN	Software Engineering, Istanbul Beykent University	_____

I hereby declare that this thesis/dissertation meets all format and submission requirements of a master's thesis

Submission date of the thesis to Institute of Graduate Studies: ___/___/___

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Intisar Jumaah Jameel JAMEEL

Signature

DEDICATION

To my amazing parents, who never stop giving of themselves in countless ways... (my late father and my late mother). To my soul mate, friend, lover and husband, To the person who wants to see me as a better person. The world is your dream come true (a dedication from the heart to you) To the sources of hope, who stand by my side in everything and lead me through the valley of darkness with the light of hope and support (my sisters) To my second mother who She prays to God for me with every step (my aunt) and my friends.

PREFACE

I indebted to ALMIGHTY ALLAH, the propitious, the benevolent and sovereign Whose blessing and glory flourished my thoughts and thrived my ambitions, giving me talented teachers, affectionate parents, sweet brothers, and sisters. Trembling lips and wet eyes, praise for HOLY PROPHET MUHAMMAD (P.B.U.H.) for enlightening our conscience with the essence of faith in ALLAH, converging all His kindness and mercy upon him.

The work presented in this manuscript was accomplished under the sympathetic attitude, politely behavior, animate directions, observant pursuit, scholarly criticism, cheering perspective and enlightened of my supervisor Assist. Prof. (Dr. Abdullahi Abdu Ibrahim) His thorough analysis and rigorous critique improved not only the quality of this dissertation, but also my overall understanding of Agricultural Extension. I am grateful to his ever-inspiring guidance, keen interest, scholarly comments and constructive suggestions throughout the course of my studies.

I would like to express my sincere gratitude to the head of the Electrical and Computer Engineering department (Prof. Dr. Mesut ÇEVİK) for giving me the opportunities to complete my studies and to overcome the difficulties that I faced during the study. I am also grateful for the insightful comments offered by the members of my Dissertation Committee. The generosity and expertise of the committee have improved this study in innumerable ways and saved me from many errors, those that inevitably remain are entirely my own responsibility. There are no proper words could ever adequately express my obligations to my affectionate and adoring my late father, mother, brothers, and sisters and all my family members whose hands always raised in prayers for me and without whose moral and financial support, the present distinction would have merely been a dream. They always acted as a lighthouse for me in the dark oceans of life path. My profound love thanks and indebtedness are due to my colleagues and friends for their moral support and motivation, which drives me to give my best. Finally, May ALLAH ALMIGHTY blesses all these people with long, happy, and peaceful lives (Ameen).

ABSTRACT

INTRUSION DETECTION IN WIRELESS SENSOR NETWORKS

JAMEEL,Intisar Jumaah Jameel
M.Sc., Information Technologies, Altınbaş University,
Supervisor: Asst. Prof. Dr. Ayça KURNAZ TÜRKBEN
Date: 01/2024
Page: 69

Given the possibility of using intelligent agents for the development of intrusion detection systems, obtained through a bibliographic survey, this work will first seek to develop the following hypothesis: The use of intelligent agents in the development of intrusion detection and prevention solutions in wireless sensor networks is quite suitable for such equipment, as they favor the development of solutions. In addition, the use of intelligent agents allows a better development of these systems, to provide satisfactory computational performance, adaptability, scalability, shorter time latency and load balancing. The general objective of this paper is to propose a framework, based on agents, for detecting intrusion in wireless sensor networks, with the best possible adaptation resources of network devices

Keywords: API, IDS, ML, DL, WSN, NLP.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

AI	:	Artificial Intelligence
DL	:	Deep Learning
ML	:	Machine Learning
CNN	:	Convolutional Neural Network
VGG	:	Visual Geometry Group
ACC	:	Accuracy
PREC	:	Precision
REC	:	Recall
F1-S	:	F1-Score

1. INTRODUCTION

In this first chapter, an introduction about the work will be presented. This first chapter will bring a contextualization and the problem for which it will aim to propose a solution, the motivation for such work, the proposed solution, the objectives — general and specific — of the present work, the methodology used for the development of the proposed solution, ending with the presentation of the structure of the entire work.

1.1 PROBLEM STATEMENT

With the advent of Ubiquitous Computing, which was mentioned for the first time in, informatics has become increasingly present in the environments in which human beings are inserted. Whether through wearable devices, context-aware devices or embedded systems, the presence of these devices in the human environment is consolidated. In addition, there was the emergence of wireless sensor technology.

Sensors are small-sized devices, usually arranged in large numbers (which vary from hundreds to thousands), whose purpose is to capture information from the physical environment in which they are inserted, such as temperature, humidity, movement, among others When these devices are located in a given geographic region and have connections between themselves, there is a network of wireless sensors

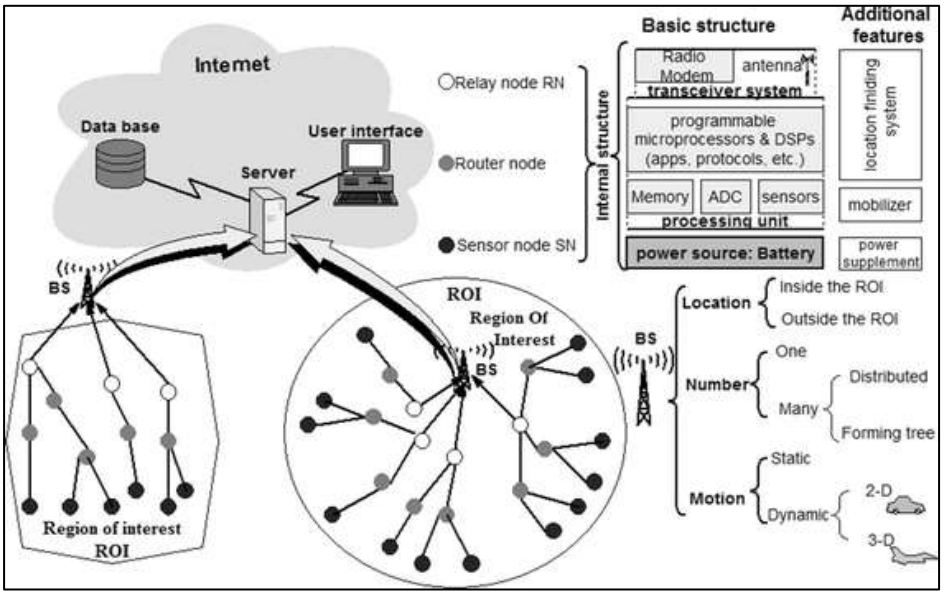

Figure 1.1: Structure of WSN.

WSNs have a wide range of applications: industrial use, medical (e-health), military, smart house construction, event sensing, etc. WSNs have the following characteristics, which decisively affect the performance of WSNs and the way they should be worked on:

a. Little or no infrastructure;
b. Nodes sensors with limited computational resources (memory, energy, bandwidth);
c. Easier compromise of sensor nodes;
d. Topology in constant change, due to faults or node mobility.

Due to the points listed above, security ends up becoming an important and complex factor in the use of WSNs. In addition to greater vulnerability to devices internal or even external to the network, sensor nodes also have a vulnerability against their own physical integrity, given their proximity to the source of collected data. . Therefore, it is necessary to create security measures aimed at mitigating or solving these security problems. However, due to the lower computational and energy power of wireless sensors, the development of security solutions for such devices must be differentiated, making use of appropriate techniques and approaches.

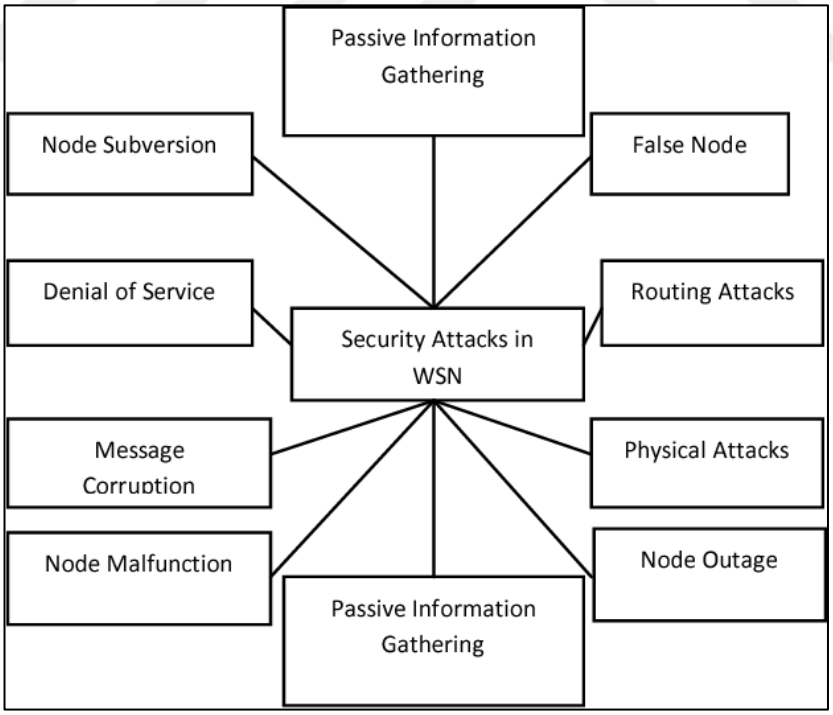

Figure 1.2: Security Concerns of WSN.

1.2 MOTIVATION

Security is a vital factor in any context where there is a computer network. In order to deal with this factor, several techniques and approaches have been developed, especially for wired networks, but these approaches, in general, end up not being as functional in wireless networks, which, by extension, it turns out not to be as functional in WSNs either. It becomes necessary, therefore, to make a compatible approach with WSNs and the wireless sensors that compose them.

Traditional approaches, such as encryption, use of VPN's and firewalls turn out to be inappropriate for use in WSNs, since they only provide protection against elements external to the network, in addition to the large consumption of computational resources that they demand. Therefore, an approach is needed that is simultaneously efficient enough for real-time containment of attackers not only external to the network, but also internal ones, and that requires fewer resources. computational.

For that, a possible approach is the use of intrusion detection systems.

(IDS). Intrusion detection systems can be defined as systems that automatically detect intrusions and attacks on a network, generating alerts and reports of the occurrence of IDSs being complex systems. , a promising approach is the decomposition of the system modules into several network nodes, thus making the IDS a distributed system. However, this distribution must also be done in a way that does not exhaust the scarce computational resources of wireless sensors. In order to achieve this requirement, intelligent agents can be used.

An agent is a computational entity capable of perceiving the environment in which it is located through sensors, performing processing based on the input provided by the sensors and, according to this processing, acting in the environment through actuators. work together and in a coordinated way – a distributed system, therefore –, the paradigm of multi-agent systems is given, which are systems that manage to provide the flexibility, efficiency and adaptability necessary for the proper functioning of an IDS.

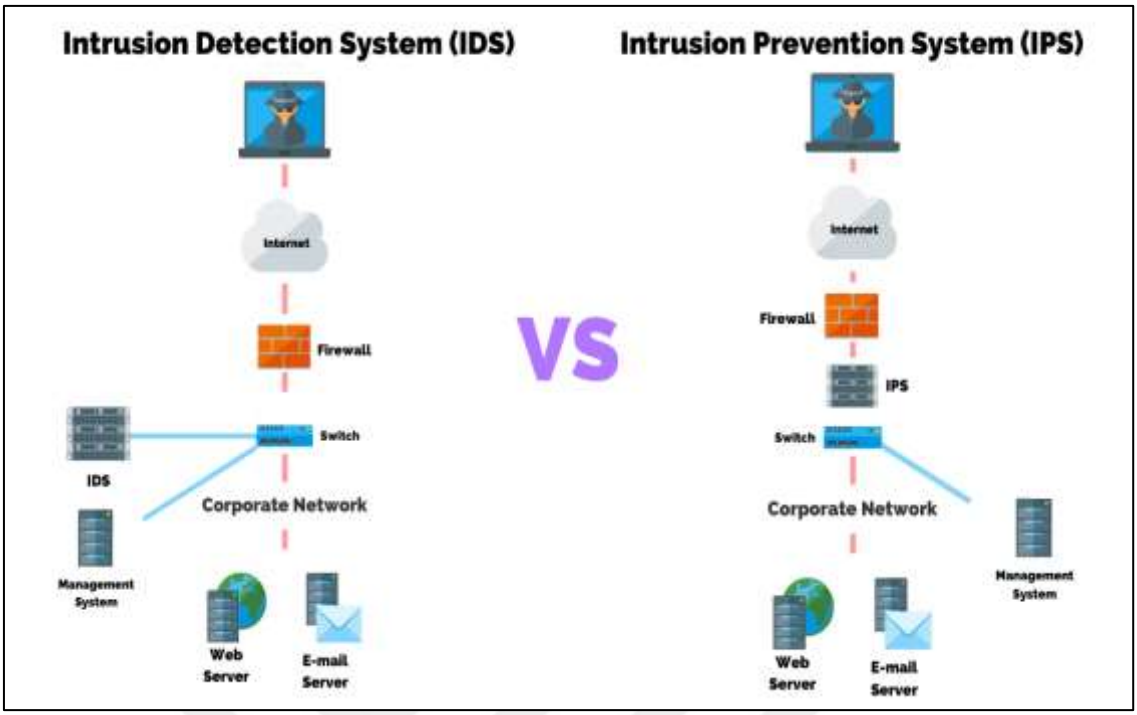

Figure 1.3: Intrusion Detection and Intrusion Prevention Systems.

Agents are different from other approaches as they act autonomously and based on goals. In addition, they have the following advantages:

a. Network latency reduction: Agents in a network node, acting autonomously, respond and act faster than if there were a central coordinator;
b. Autonomous execution: Agents continue to act, even with an eventual loss or damage to the network;
c. Performance in heterogeneous environments: Agents are capable of operating in heterogeneous environments, thus creating a multiplatform system;
d. Network load reduction: In case of need, an agent may be able to move between hosts on the network, increasing the system's load distribution.

1.3 AIMS AND OBJECTIVES

Given the possibility of using intelligent agents for the development of intrusion detection systems, obtained through a bibliographic survey, this work will first seek to develop the following hypothesis: The use of intelligent agents in the development of intrusion detection and prevention solutions in wireless sensor networks is quite suitable for such equipment, as they favor the development of solutions. oes whose performance is suitable for wireless sensors. In addition, the use of intelligent agents allows a better

development of these systems, in order to provide satisfactory computational performance, adaptability, scalability, shorter time latency and load balancing.

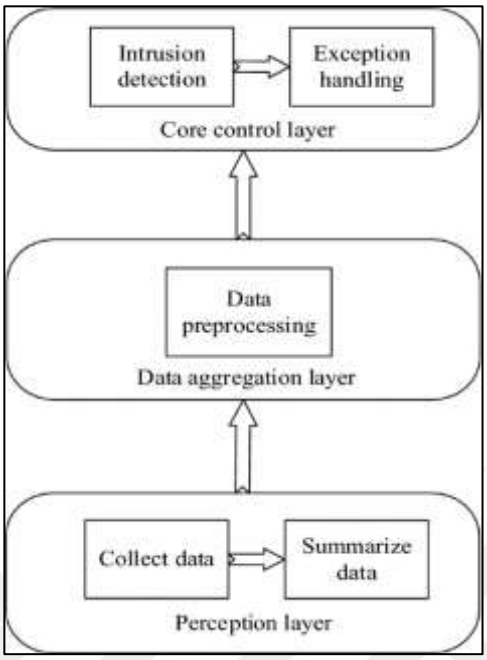

Figure 1.4: General Structure of The Proposed System.

In order to prove the veracity of this hypothesis, this dissertation outlines some general and specific objectives, described as follows.

The general objective of this dissertation is to propose a framework, based on agents, for detecting intrusion in wireless sensor networks, with the best possible adaptation resources of network devices.

In order to help achieve the general objective, this work has the following specific objectives:

a. Conceive, design, implement a framework based on agents for intrusion detection in wireless sensor networks;
b. Design and implement a computational environment from the proposed framework, which allows the construction of a controlled scenario;
c. Design and deploy simulated attack scenarios against the deployed wireless sensor network;
d. Design and implement measurement scenarios, which make it possible to measure the consumption of the hardware available for the framework;

1.4 CONTRIBUTIONS

In order to achieve the previously mentioned objectives, this work will make use of some methods, listed below:

a. Bibliographic review, carried out through a survey of articles, annals of events and web addresses;
b. Definition of the scope of the problem, so that the limits of the work and its subsequent contribution are clarified;
c. Define the necessary requirements for the computational environment built from the proposed framework;
d. Implement the necessary requirements for simulated attack scenarios in the controlled computing environment;
e. Implement the necessary requirements for the development and correct application of measurement tests.

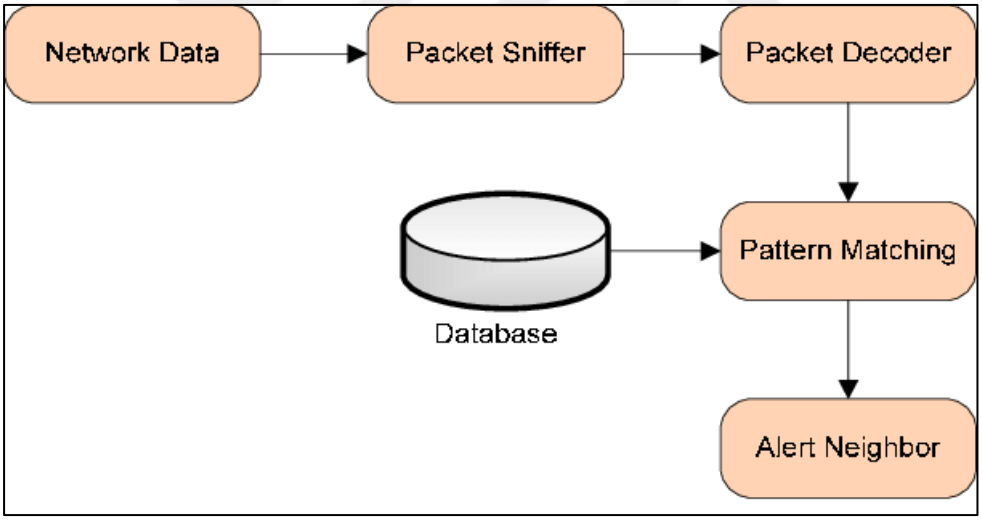

Figure 1.5: Steps Of the Proposed System.

1.5 THESIS STRUCTURE

The writing of this work is distributed over six chapters, which will be described in detail below:

Chapter 2: This chapter presents a technological context, which brings an approach to the theoretical foundations whose knowledge is necessary for the understanding of this work. In this chapter, the concepts related to Wireless Sensor Networks, Threats and

Countermeasures in Wireless Sensor Networks, Intrusion Detection Systems and Intelligent Agents are addressed. Finally, a summary of the chapter is presented;

Chapter 3: In this chapter, a state of the art is presented, that is, a presentation and a discussion of the most important and recent works referring to the concepts discussed in Chapter 2 is made, being made later a comparison between those considered most important for this work. At the end of the chapter, a summary is presented;

Chapter 4: In this chapter, the purpose of this dissertation is presented: an Intrusion Detection Framework in WSNs. The architecture of the proposal, its modeling and its prototyping, that is, the exposition of a framework functioning scenario will be presented. At the end of the chapter, a summary is presented;

Chapter 5: In this chapter, the evaluations and discussions about the tests carried out in the framework's operating scenario are presented, as well as the environment and data used in the tests, the test results and comparisons with related works. Finally, a summary of the tests is presented;

Chapter 6: In this chapter, the conclusions of the dissertation work will be presented, as well as a discussion about the achieved objectives, eventual limitations presented by the framework, academic publications referring to the work and suggestions for future work.

2. LITERATURE REVIEW

2.1 INTRODUCTION TO WSN SECURITY

A sensor network is a type of distributed system that comprises of sensor nodes that are connected to one another and a central base station node that is used to collect sensing data. Each sensor node in a sensor network has its own unique identifier. Nodes are what you call each individual sensor node that makes up a sensor network (light, vibration, temperature, etc.). We refer to a network as having "wireless sensor connections" whenever there are wireless connections between the sensors that make up that network. Additionally, we use the phrase "wireless sensor connections" interchangeably with this network's name (WSN). Since their inception, wireless sensor networks, sometimes referred to as WSNs, have been implemented in a wide range of distinct ecosystems [1, 2]. WSNs have been put to use in an extremely wide variety of diverse applications, ranging from the observation of animals [2, ecological research [3, observation of volcanoes [4, observation of water [5, and so on]. WSNs, on the other hand, are faced with a combination of challenges that are unique to them due to the one-of-a-kind nature of their construction. When constructing a WSN, it is strongly recommended that energy economy be taken into consideration. This will allow for additional reduction in expenditures that are associated to maintenance and redeployment [6]. For instance, the expected duration of life for a sensor node is limited by the battery that is attached to it, and the expected duration of life for the network is defined by the expected duration of life for the sensor nodes. However, in order to satisfy the requirements of sensor information gathering systems, it is necessary to set up a network that takes into account a number of challenges that need to be met and overcome. This is an extremely important part of the procedure. To get things started, it is quite likely that all of the area of interest will need to be covered by the sensors that are installed in order for the process of gathering sensor data to be successful. This is something that would be required for the process of collecting sensor data. It is anticipated that a large number of sensors will need to be positioned at just the proper angles in order to collect data from specified locations. It is possible to gather data from a wide range of sources when using a wide variety of sensors with a wide variety of sample rates. This makes it possible to collect data from a wide variety of sources. After the data have been collected, they can be analyzed. In the event that these issues are not resolved, the lifespan of a WSN may be significantly shortened as a result of

unequal use of energy. This could happen if the concerns are not addressed. One or more of a wide variety of causes may be to blame for this. When there are faults during the process of uploading data to the base station, it makes it more difficult to finish the procedures for data accumulation. This necessitates the formulation of novel strategies in order to enhance the operational capabilities of the network [7].

2.2 RELATED WORKS

2.2.1 Deep Learning Based Methods

The authors of [24] provided an innovative method for Geometric Area Analysis that makes use of Trapezoidal Area Estimation with each sample. This method was presented by the authors. As part of the Geometric Area Analysis project, this methodology was given. In order to put this plan into action, we relied on the Beta Mixture Model to conduct an analysis not just of the characteristics of the data but also of the connections that existed between them. Since it compares each record to a collection of known intervals, this adaptive and compact anomaly-based technique may be able to spot attacks. The normal activities that take place during the period of training have been retained in the form of a regular pattern in order to make this process simpler and more straightforward. [25] provides an overview of a prospective method for multi-objective FS, which is currently being taken into consideration. This model is used to evaluate the efficacy of the FS technique from a variety of different vantage points, such as the optimality of the selected feature count, the FAR, the true positive rate, precision, and accuracy. Specifically, this model is used to evaluate the efficacy of the FS technique from the perspective of the true positive rate. A multi-objective optimization strategy is now being created in parallel with the development of the strategy itself in order to facilitate the discovery of a solution to the aforementioned model. Following the construction of the evolution strategy pool and the dominance strategy pool, the algorithm will then select a dominance or evolution strategy at random from the possibilities that are available in order to promote convergence and variety. This will be done in order to ensure that there is a balance between the two goals of convergence and variety. This will be done in order to guarantee that the two aims of convergence and variety are met in an equitable manner. The authors of the article [26] propose a strategy for determining which features of a product ought to be included in the final edition of the product. This method can be found in the article. You can learn more about this approach by reading the article. After that, the hybrid ensemble learning that was suggested is applied to these subsets in

order to gather knowledge from them. This is done in order to increase the IDS's precision and consistency. The following is a list of the goals that this research endeavors to accomplish, each of which is important in its own right: 1) Decrease the dimensionality of the CICIDS2017 dataset by combining Correlation Feature Selection and Forest Panelized Attributes (CFS-FPA); 2) Determine the most effective machine learning strategy for aggregating the four modified classifiers (Support Vector Machine, Random Forests, Naive Bayes, and K-Nearest Neighbor (KNN); and 3) Validate the effectiveness of the hybrid ensemble scheme. 3) Validate the effectiveness of the hybrid ensemble scheme. Accuracy, detections, and false alarms are only a few of the many aspects of feature selection that need to be reviewed in addition to the CFS-FPA and other approaches to feature selection.

2.2.2 Feature Selection Based Methods

As a direct consequence of this, the data will be used to make the approach that was supplied for the selection of features even more effective in terms of its overall efficiency. It is possible for us to evaluate how well the various classification methods work both before and after they have been modified to make use of the AdaBoosting approach. This gives us the opportunity to compare the results of the two scenarios. Because of this, we will be able to evaluate how successfully the various classification algorithms were implemented. They will also evaluate any potential alternatives to the solution that was chosen and judge how well it compares to the one that was ultimately chosen. In [27], a deep neural network (DNN) feature extractor as well as a deep neural network (DNN) controlled k-means clustering component were developed. During the training process for both of these components, proximal policy optimization was utilized. (PPO). K-means clustering is utilized in order to recognize possible invaders, and the Integrated Defense and Homeland Cybersecurity System (IDHCS) monitors the DNN feature extractor in order to ensure that the most relevant features are gleaned from the architecture of the network. IDHCS is able to automatically learn and adjust its performance to the particulars of the network in which it is deployed since it is built using a reinforcement learning technique that is based on PPOs (program-based optimization). A learning algorithm is utilized in order to successfully complete this task. As a result of this, it is now in a position to conform more closely to the specifications of the network. Simulations were run on both the CICIDS2017 and the UNSW-NB15 databases so that the researchers could evaluate how well the methodology worked. This allowed the researchers to determine how well the methodology functioned.

The value of 0.96552 that is assigned to the F1-score of CICIDS2017 is significantly greater than the value of 0.94268 that is allocated to the F1-score of NB15 by UNSW. The integration of the two data sets produced a picture that was not only more complete but also more precise than the one that had been previously available. After going through all of the data and evaluating it, the researchers arrived at the conclusion that not only had there been an increase in the overall number of attacks, but there had also been an increase in the level of sophistication that was involved in each individual attack. A combined performance of between 97% and 99% is indicated by an F1 score of 0.93567 when evaluated against both CICIDS2017 and UNSW-NB15. [Here's a good example:] [Here's a good example:] According to the findings, the suggested IDHCS was able to increase the effectiveness of the IDS by automatically learning new types of attacks through the management of intrusion detection features. This enabled the IDS to better protect the network. Because of this, the overall level of security that the IDS offers was able to improve as a result. These qualities have not changed despite the fact that the underlying network architecture has been subjected to a number of modifications. The authors of this research paper [28] present a machine learning (ML)-based intrusion detection system (IDS) that is capable of distinguishing between different types of distributed denial of service (DDoS) attacks and is acceptable for use in a practical industry. They collect innocuous information from the semiconductor fabrication locations that are owned and operated by Infineon. This information includes data on network traffic, for example. They scan through the DDoSDB database, which is handled by the University of Twente, in order to discover fingerprints and actual DDoS attacks that have occurred in the real world. When training one of eight supervised learning algorithms (Linear Regression (LR), Naive Bayes (NB), Bayesian Network (BN), K-Nearest Neighbor (KNN), or Classifier), we can reduce the dimensionality of the input by employing feature selection methods such as Principal Component Analysis (PCA), for example. These methods allow us to select features from the data that are being used to train the algorithm. These strategies provide us the ability to choose features from the data that are being utilized in the process of training the algorithm. After that, the authors did research on a semi-supervised and statistical classifier that is known as the univariate Gaussian approach, as well as research on two unsupervised learning algorithms that are known as simple K-Means and expectation maximization (EM). In addition, the authors did research on an unsupervised learning algorithm that is known as the expectation maximization (EM)

algorithm (EM). It is the first study of its kind, as far as the research team is aware, to train machine learning models to detect distributed denial of service assaults using data acquired from genuine industrial scenarios. The information was compiled with the help of a wide range of various businesses. For previous publications, the solution to the challenge of recognizing distributed denial-of-service (DDoS) assaults in operational technology (OT) networks was to use data that was either developed particularly for the purpose or taken from an IT network. OT stands for operational technology.

2.2.3 DDOS Detection Methods

Despite the fact that machine learning has been applied in an attempt to solve the issue of identifying DDoS assaults in OT networks, this task has been successfully completed. The idea of hybrid feature selection (HFS) was initially presented by making use of an ensemble classifier as a result of the research that Jaw and Wang carried out [29]. This method detects key qualities in a manner that minimizes the amount of effort spent doing so and creates a category that is suitable for the assault. Initially. It achieved an outcome accuracy of 99.99% for the CIC-IDS2017 study, 99.73% for the NSLKDD study, and 99.997% for the UNSW-NB15 study, respectively. Additionally, it achieved a detection rate of 99.75% for each of these studies, respectively. In addition to this, its performance was superior to that of each of the individual classification methods that were chosen, as well as that of the most recent FS and some of the most recent IDS procedures. On the other hand, the outcomes are dependant on eleven, eight, and thirteen variables taken from the dataset, all of which were selected after a great lot of thought and deliberation. You may find an examination of intrusion detection systems (IDSs) written from the point of view of machine learning in the paper [30]. They discuss the three primary challenges that an IDS faces in general, as well as the concerns that an IDS faces for the internet of things (IoT). These challenges are idea drift, high dimensionality, and high computational. They also discuss the concerns that an IDS faces for the internet of things (IoT). In addition to this, they talk about the worries associated with an IDS and the internet of things (IoT). A conversation takes occur, during which prospective future study issues and studies that try to find answers to each difficulty are discussed as potential topics for future research. In addition, the authors of the study have included datasets pertaining to IDS in a section of the research that is only focused on carrying out the specific function that was indicated before. The KDD99 dataset, the NSL dataset, and the Kyoto dataset are three examples that really stand out from the rest of the

pack. According to the conclusions of this research, the idea of an intrusion detection system (IDS) that is built on neural networks needs to overcome the symmetrical issues of concept drift, an increasing amount of characteristics, and computational awareness in order to be successful (IoT). The authors of the paper [31] investigated the efficacy of six distinct supervised classifiers by employing the Weka tool and a 10-fold cross-validation methodology on the entire NSL-KDD training dataset. This allowed them to determine which of the six methods produced the most accurate results. The results of their investigation are presented in the following paragraph. This was accomplished both with and without the adoption of techniques for feature selection and reduction. As a part of their research, the authors of the study looked at different methods that could be used to improve and protect classifiers. This was done in order for them to accomplish their goal. They were particularly interested in classifiers that required the least amount of time to create models while simultaneously having the highest levels of detection accuracy. According to the findings of this study, employing a method for feature selection and reduction, such as the wrapper method in conjunction with the discretize filter, the filter method in conjunction with the discretize filter, or just the discretize filter, can significantly cut down the amount of time spent on model construction without sacrificing the accuracy of detection. These methods include the filter method in conjunction with the discretize filter, the wrapper method, and the discretize filter. These methods consist of the wrapper method, the filter method in conjunction with the discretize filter, and the discretize filter alone. Auto-encoder (AE) and principal component analysis are the two methods that Abdulhammed and his coworkers [32] utilize in order to reduce the dimensionality of the data to a level that is more manageable (PCA). An essential component of the process of developing an intrusion detection system is the step of incorporating the low-dimensional characteristics obtained by either method into classifiers such as RF, Bayesian Networks, Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis. This step is performed after obtaining the low-dimensional characteristics using either method. Either approach can be utilized in order to acquire these low-dimensional properties (QDA). Experiments that were conducted using binary and multi-class categorization with low-dimensional features indicated an improvement in performance in terms of detection rate, F-measure, false alarm rate, and accuracy. The findings of this study led to a reduction in the number of feature dimensions that were included in the CICIDS2017 dataset from 81 to 10, as well as an

increase in the accuracy of binary and multi-class classification to 99.6%. This was accomplished as a result of the fact that the number of feature dimensions that were included in the dataset decreased.

2.2.4 RNN Based Methods

The authors of article 33 put forward a novel idea for an intrusion detection system (IDS) in which random neural networks and ABC play a role in the detection process. (RNN-ABC). When the researchers are testing and refining the model that they have developed, they use the NSL-KDD data set as the standard against which they measure their success. The regular RNN that was trained using the gradient descent approach is compared to the recommended RNN-ABC, which is evaluated based on a number of different metrics, one of which is accuracy. This comparison is carried out so that the reader may see which model performs better. The two different networks are compared side by side in this article. Having said that, while still maintaining an accuracy rating of 95.02% across the board, In the article referred to as reference number 34, we take a look at a model of an intrusion detection system that was developed with the assistance of ML. The proposed technique reduces the amount of money that is spent on calculation while keeping the same high requirements for the precision of detection. In this section, we identify the smallest practicable training record group by analyzing the effect that a variety of oversampling procedures have on the smallest sample size that is required to train a particular model. This is done by looking at the effect that these procedures have on the minimum amount of data that must be collected in order to train the model. This is accomplished by analyzing the effect that the aforementioned processes have on the minimum number of data points needed to successfully train the model. As a result of the hyperparameters of the IDS, there are currently a great deal of distinct strategies for performance optimization that are the subject of research and mental effort. In order to determine the extent to which the newly developed system may be relied upon, a comparison is drawn between it and the dataset provided by NSL-KDD. According to the findings of the studies, using the strategy in question not only cuts down by 74% on the overall number of mandate items, but it also cuts down by 50% on the size of the feature set. (20-featured). This study proposes a conceptual framework for the recognition of traffic outliers that is known as a deep learning algorithm for IDS. The framework was developed as part of this research. In the course of this research, the framework was suggested. This algorithm is generated by first making use of a convolutional neural network in order to

obtain sequence attributes of data traffic, then reassigning the weights of each channel making use of attention methodology, and finally making use of a bidirectional long short-term memory (Bi-LSTM) in order to learn the architecture of sequence attributes. In order to locate traffic anomalies, it is necessary to carry out each and every one of these procedures in their entirety. There is a substantial amount of data inconsistency existing within the databases that are accessible to the general public in an unrestricted manner and that contain IDS information. These databases are entirely open to the public. Therefore, the research that is published in [35] uses a redesigned stacked autoencoder for the purpose of cutting down on the number of data features in order to make progress toward the objective of improving knowledge integration. Additionally, it uses adaptive synthetic sampling to increase the sample size of observations from minority categories in order to produce a dataset that is statistically representative of the entire population. This is done in order to generate a dataset that can be used to make inferences about the entire population. This is done so that a dataset that can be used to draw inferences about the entire population can be produced. Because it is built on a full-stack structure, the deep learning algorithm that is used in the IDS approach does not require any feature extraction to be conducted manually. Because of this, the IDS technique can reduce the amount of time and effort it requires. According to the findings of the study, using this method as a comparison to the public standard dataset for IDS, which is known as NSL-KDD, results in a greater accuracy (90.73%) and F1-measure (89.65%) than using previous methods of comparison. This dataset is known as NSL-KDD. The authors of the article [36] present a new feature-driven intrusion detection system that they refer to as 2-Bi-LSTM. This system was developed by the authors. This IDS makes use of a two-pronged technique in conjunction with the Bi-LSTM algorithm in order to perform the statistical analysis that it needs to undertake. The 2-Bi-LSTM method begins with a 2-based evaluation of each of the characteristics, and then it proceeds to an up-leading search in order to find the combination of those characteristics that produces the best possible result. This search is done in order to find the optimal combination of the characteristics. In the final stage, we take the ideal set and put it through a Bi-LSTM framework so that it may be categorized. This completes the process. Because of this, we are able to recognize patterns within the data. The empirical findings show that the suggested 2-Bi-LSTM strategy achieves a higher accuracy rate (95.62%), F-score (95.65%), and minimal FPR (2.11%) than the standard LSTM technique and possibly

several available attribute IDS alternatives. This is demonstrated by the fact that the 2-Bi-LSTM strategy has a higher F-score than the standard LSTM technique. The fact that these results were obtained using the 2-Bi-LSTM technique is evidence that demonstrates this point. These findings are predicated on the NSL-KDD dataset, which served as the basis for their development. Li et al. argue in their study [37] that software-defined 5G networks should be equipped with advanced intrusion detection systems (IDS). This is achieved by employing software-defined technologies to accelerate the process of invoking the multiple security-related function modules and standardizing the administration of those modules. These technologies also help to standardize the administration of those modules. In addition to this, it makes the process of activating the numerous function modules that are associated with security easier.

2.2.5 Boosting and Optimization Based Methods

In addition, it makes use of machine learning in order to automatically acquire rules by sifting through enormous amounts of data, and it makes use of flow classification in order to find dangers that were not anticipated. Both of these characteristics give it the ability to recognize threats that could not have been anticipated at the time. It accomplishes flow classification through the utilization of a combination of k-means and adaptive boosting (AdaBoost), and it separates flows through the application of radio frequency (RF). There are two different acronyms for adaptive boosting: k-means and AdaBoost (FS). The writers are of the belief that if the plan that has been stated is put into reality, the subsequent 5G networks will have a security level that is noticeably higher than it is currently at. The plasmodium, on the other hand, is responsible for the construction of an efficient network of protoplasmic tubes that connect the masses of protoplasm to the various components of the meal. A plasmodial network connects each of these tubes to one another. The veins that connect the expanding front end of the body to the remainder of the body and to other sections of the body expand along with it as the front end of the body continues to expand. The cytoplasm of the cell is able to enter the cell through these veins, which generate channels in the cell. Molds have a circulatory system that permits them to move around and look for food. This gives them the ability to move around and look for food. They create enzymes that allow them to map the locations of potential sources of food as they move about. It sheds light on the tactics that this organism employs in order to navigate its environment in search of a source of sustenance the fact that slime mold may develop even

in settings in which there is a lack of food. The ability of a slime to evaluate the proportion of positive and negative signals it receives as a secretion moves closer to its target enables the slime to determine the strategy that will be the most successful while it is attempting to collect food. This is because the slime can compare the signals it receives as the secretion gets closer to its target. This indicates that slime mold, with a sufficient amount of food, is capable of constructing a solid concrete road if the right conditions are provided for it. It makes a habit of going to the site that it has figured out to have the greatest number of diverse potential sources of sustenance, and it does this journey on a consistent basis. Molds look at these characteristics when they are foraging in order to calculate how quickly they need to move on to a new site in order to find food and a place to live. Molds consider all of these aspects when determining how quickly they should go to the next stage. Molds incorporate the information they have gathered into the mechanisms through which they make decisions. Since the slime mold is only able to acquire a limited quantity of data, it must rely on empirical principles in order to decide whether or not it should begin a new search and move from its current location. Even if it has access to a big supply of food, mold may still divide its biomass in order to make better use of the resources it already possesses. This may be done in an effort to maximize resource utilization. Even if there is an adequate quantity of food, this can still happen. It is quite possible that it will make impromptu adjustments to its search criteria in order to take into account shifting circumstances about the availability of food.

3. MATERIALS AND METHODS

3.1 CHAPTER INTRODUCTION

Current technologies in the production of electronic components, and in particular of microprocessors, make it possible to develop equipment of increasingly reduced size and weight. This has allowed the appearance of increasingly powerful portable computer objects, such as portable computers and personal assistants (PA), as well as communication between these devices which is of the wireless type.

The use of a wireless interface introduces differences compared to communication by cable. First, the radio spectrum, and therefore the capacity available for data transfer, is limited by regulation. Where an addition of cable suffices to increase the number of users who can be satisfied on a fixed network, the frequency band occupied by a mobile network cannot be extended. This restriction also limits the available throughput imposing the need for efficient use of the channel.

Then, the quality of the radio links can vary over time according to the various interferences and the mobility of the nodes. This situation therefore leads to a higher transmission error rate than on a wired network and above all to a very fluctuating rate.

The advances made in the miniaturization of electromechanical systems (MES) have allowed the appearance of a new type of wireless network: sensor networks. These networks are a special type of ad hoc networks. They use a large number of devices called nodes. These objects can autonomously collect and transmit environmental data.

Not so long ago, the only solution to route sensor data to the central controller was wiring, which had the main drawbacks of being expensive and cumbersome.

3.2 HISTORY OF TELECOMMUNICATIONS

Telecommunication is any transmission, emission or reception of signs, signals, writings, images, sounds or information of any kind, by wire, radioelectricity, optics or other electromagnetic systems.

3.2.1. Wire Communication

In the 1840s, the American Samuel Morse invented a simple electric telegraph: batteries, a switch, an electromagnet and wires were enough.

Morse's device, which transmitted the first public telegram in 1844, looked like a simple electrical switch. It allowed the passage of a current for a predefined time and then

interrupted it, all controlled with the pressure of a finger. The first Morse receiver was equipped with an electromagnetically controlled pencil. This pencil printed marks on a strip of paper, fixed on a cylinder animated by a clockwork. The marks varied in length with the duration of the pulses of electric current passing through the wires of an electromagnet and took the visual form of dots and dashes. Subsequently, the operators learned to recognize the lines and dots transmitted to them directly by ear. His device was adopted by most European countries and national networks based on the Morse telegraph were born in the United States, France, England... In 1866, after several unsuccessful attempts, the first transatlantic cable was installed and with it, the first truly global telecommunications network developed. Around 1940, the first era of modern computing made its appearance. Quickly, the adaptation of telecommunications technologies to computers was quickly essential. In 1957, the US Department of Defense created the Advanced Research Projects Agency (ARPA). In this context, the need to communicate the different research teams in the four corners of the United States is felt. This need led researchers at the

3.2.2. Wireless Communication

Recently, wireless communication systems offer users the possibility of enjoying the joys of telecommunications regardless of their geographical location. However, wireless communication is almost as old as wired communication...

In 1887, Heinrich Hertz verifies Maxwell's theories by experiment. The latter, established mathematically by James Maxwell, tell us that any electrical disturbance gives rise to electromagnetic oscillations. These oscillations will come to be known as Hertzian waves. In 1890, Edouard Branly discovered the first receiver sensitive to Hertzian waves. Based on the work of Branly, the Italian Gugliemo Marconi invented the first wireless telegraph device in 1895. Then, Marconi went from success to success by increasing the transmission distances to reach, in 1903, the complete transmission of a message over a distance of 3400 km!

Until the end of the 1980s, wireless technology was mainly used for radio, television or communications reserved for large organizations such as the army. The advent of GSM (Global System for Mobile communication) cell phones has given everyone the ability to communicate from anywhere, with anyone. However, such a device requires the deployment of an expensive infrastructure to ensure the relay between the mobile telephones and the wired telephone network.

3.3. WIRELESS NETWORKS

3.3.1. AD-HOC Networks

An ad hoc wireless network (or MANET, for Mobile Ad hoc NETwork [2]) is formed by a set of hosts which organize themselves on their own and in a totally decentralized manner, thus forming an autonomous and dynamic network not relying on any wired infrastructure. These hosts can be fixed or mobile. According to these hypotheses, any set of objects equipped with an adequate communication interface is likely to spontaneously form such a network. No infrastructure being available, these objects therefore have to dynamically discover their environment. An ad hoc network being above all a wireless network, the objects communicate with each other through a radio interface. These communications are therefore subject to the physical phenomena which govern radio waves, such as a strong attenuation of the signal with distance. Thus, only hosts close enough to each other are able to communicate directly with each other, and long-distance communications must take place through a mechanism called multi-hop: this simply means that certain objects must relay messages step by step until they are delivered. The use of an omnidirectional radio antenna also implies that a message sent by any transmitter is received by all receivers sufficiently close to it. it simply means that certain objects must relay the messages step by step until their routing is carried out. The use of an omnidirectional radio antenna also implies that a message sent by any transmitter is received by all receivers sufficiently close to it. it simply means that certain objects must relay the messages step by step until their routing is carried out. The use of an omnidirectional radio antenna also implies that a message sent by any transmitter is received by all receivers sufficiently close to it.

Figure 3.1 illustrates a classic wireless network such as can be found, for example, in train stations and airports. The infrastructure there is made up of two access points P1 and P2 connected by means of a conventional wired link, and which serve as entry points for the hosts of the network. When object a wishes to communicate with object d, it sends the messages to P1 which forwards them to P2, the latter sending them to d.

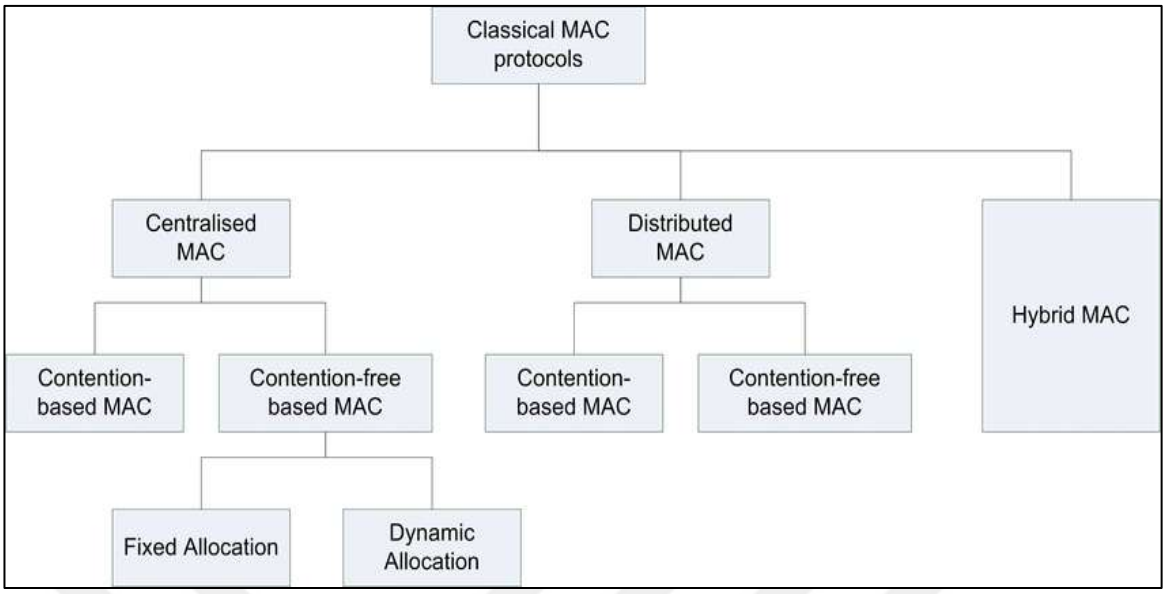

Figure 3.1: Classic Wireless Network.

Figure 3.2 illustrates an ad hoc network where no infrastructure is needed for hosts to communicate with each other. The object c must therefore serve as a relay so that a can communicate with d.

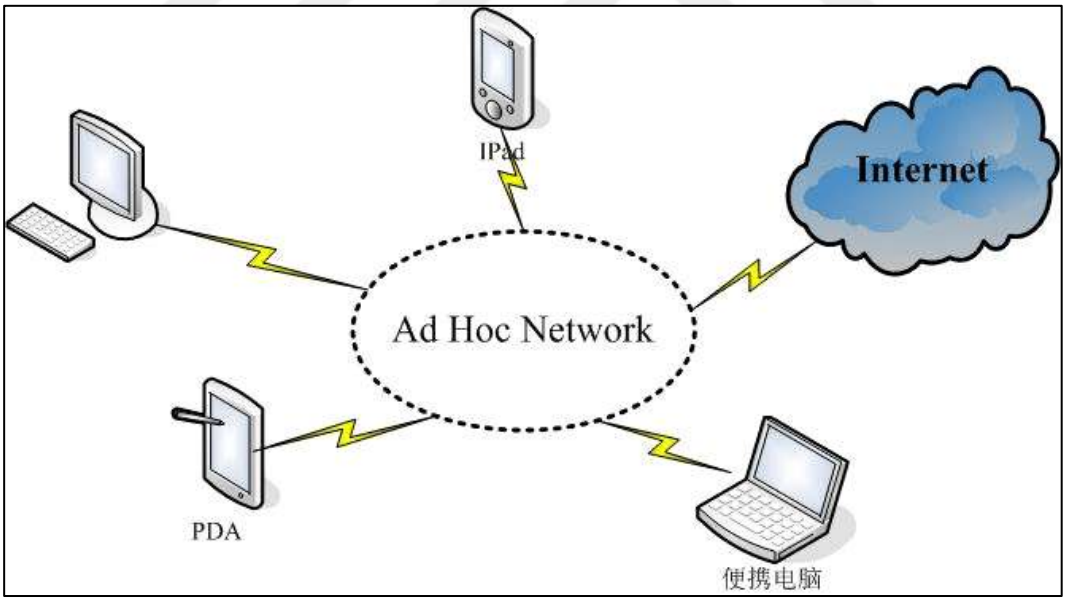

Figure 3.2: Ad Hoc Wireless Network.

The applications of these networks are multiple, and mainly concern areas where a wired infrastructure is unavailable or undesirable. This is the case, for example, in areas affected by a natural disaster, where the emergency services have a great need for communication. This is also the case when speed and discretion are determining factors: one cannot

reasonably imagine the deployment of a complete communications infrastructure during military maneuvers in enemy territory. Other lighter use cases may also arise. Thus, for cost reasons, it is not possible, for example, to set up a wired infrastructure for the duration of an outdoor meeting. In all these examples, the use of an ad hoc network can be

The design of communication protocols for ad hoc networks is mainly subject to three factors, which can be summarized as follows:

Limited Power Hosts run on a battery, which typically lasts a few hours of use; communications should therefore be kept to a strict minimum.

Autonomy of decision No central authority is present to manage the operations of the various hosts, the network is therefore entirely decentralized.

Dynamic topology Hosts are mobile and can be connected to each other in arbitrary ways. Radio links change regularly, when objects move, go out, or when obstacles appear or disappear.

3.3.2. Wireless Sensor Networks (WSN)

3.3.2.1 Sensor

It is a system which is used to detect, in the form of an often electrical signal, a physical phenomenon in order to represent it.

Sensors are small devices with a battery, capable of communicating with each other and detecting events if they are within their range of perception.

A sensor is a small device with mechanisms that allow it to pick up information about its environment. The nature of this information varies widely depending on the use made of the sensor: the latter can just as easily take readings of temperature, humidity or light intensity. A sensor also has the necessary hardware to perform wireless communications by radio waves [3].

3.3.2.2 Definition of an WSN

Wireless sensor networks are considered a special type of ad hoc networks where fixed communication infrastructure and centralized administration are absent and the nodes act as both hosts and routers. The sensor nodes are intelligent "smart sensors", capable of performing three complementary tasks: the reading of a physical quantity, the possible processing of this information and the communication with other sensors. All of these sensors, deployed for an application, form a network of sensors. The purpose of this is to

monitor a geographical area, and sometimes to act on it (in this case it is a question of sensor-actuator networks). Examples include a forest fire detection network, or a network for monitoring the strength of a bridge after an earthquake. The network can have a large number of nodes (thousands). The sensors are placed more or less randomly (for example by dropping from a helicopter) in potentially dangerous environments. Any human intervention after the deployment of the sensor nodes is most of the time excluded, the network must therefore be self-managed. So that the sensor nodes work in a cooperative way, the information collected is shared between them over the air. The choice of the radio link rather than the wired link allows easy and rapid deployment in an environment that may be inaccessible to humans [3, 5]. The sensors are placed more or less randomly (for example by dropping from a helicopter) in potentially dangerous environments. Any human intervention after the deployment of the sensor nodes is most of the time excluded, the network must therefore be self-managed. So that the sensor nodes work in a cooperative way, the information collected is shared between them over the air. The choice of the radio link rather than the wired link allows easy and rapid deployment in an environment that may be inaccessible to humans [3, 5]. The sensors are placed more or less randomly (for example by dropping from a helicopter) in potentially dangerous environments. Any human intervention after the deployment of the sensor nodes is most of the time excluded, the network must therefore be self-managed. So that the sensor nodes work in a cooperative way, the information collected is shared between them over the air. The choice of the radio link rather than the wired link allows easy and rapid deployment in an environment that may be inaccessible to humans [3, 5]. So that the sensor nodes work in a cooperative way, the information collected is shared between them over the air. The choice of the radio link rather than the wired link allows easy and rapid deployment in an environment that may be inaccessible to humans [3, 5]. So that the sensor nodes work in a cooperative way, the information collected is shared between them over the air. The choice of the radio link rather than the wired link allows easy and rapid deployment in an environment that may be inaccessible to humans [3, 5].

3.3.2.3 Basic objective of WSNs

The basic objectives of wireless sensor networks generally depend on the applications, however the following tasks are common to several applications:

a. Determine the values of a few parameters according to a given situation. For example, in an environmental network, one can seek to know the temperature, the atmospheric pressure, the amount of sunlight, and the relative humidity in a number of sites, etc.
b. Detect the occurrence of the events of interest and estimate the parameters of the detected events. In traffic control networks, one may want to detect the movement of vehicles through an intersection and estimate the speed and direction of the vehicle.
c. Classify the detected object. In a traffic network, is a vehicle a car, bus, etc.

3.3.2.4 Types of networks

WSNs can be classified according to two points of view:

a. The dynamic network model

a. Either the network consists of a set of mobile sensors evolving in a static environment. The purpose of such networks is most of the time the exploration of inaccessible or dangerous areas. Research work is often oriented robotics, the nodes playing both the role of sensor and actuator.
b. Either the network is made up of fixed sensors used to monitor the occurrence of events in a geographical area [4]. Here, the network only performs monitoring, the measured data is transmitted in multi-hop mode to a specific node called “sink” which is responsible, after reception, for implementing the necessary actions. This well can be connected, by wire for example, to another network.

b. The data delivery model

a. Either the sensors periodically transmit the information collected (continuous data delivery).
b. Either the sensor transmits information upon detection of an event (delivery of event-based data "event-driven").
c. It is up to the user to launch a request for the information ("observerinitiated").
d. Hybrid data issue where the different issues mentioned above are found at the same time.

3.3.3. Comparison of Sensor Networks and Ad Hoc Networks

The following table illustrates the difference between an WSN and an ad hoc network:

Table 3.1: Comparison Sensors / ad Hoc.

Sensors or sensors	**Ad hoc**
Targeted objective	Generic / communication
Nodes collaborate to complete a goal	Each node has its own purpose
Data stream all to one (Many-toone)	Flow all to all (Any-to-any)
Very large number of nodes not all having an ID identifier	Concept of ID
Energy is a determining factor, sensor node prone to failures	Flow is major

3.4 BASIC ARCHITECTURE OF A SENSOR

A sensor is made up of four main elements:

a. An element responsible for measuring the external environment (capture unit),-A computing unit,

b. A transmitter/receiver element, - A power supply.

Three additional components can be implanted in a sensor:

a. A location search system,

b. A power generator,

c. A mobile unit (allowing to move the sensor).

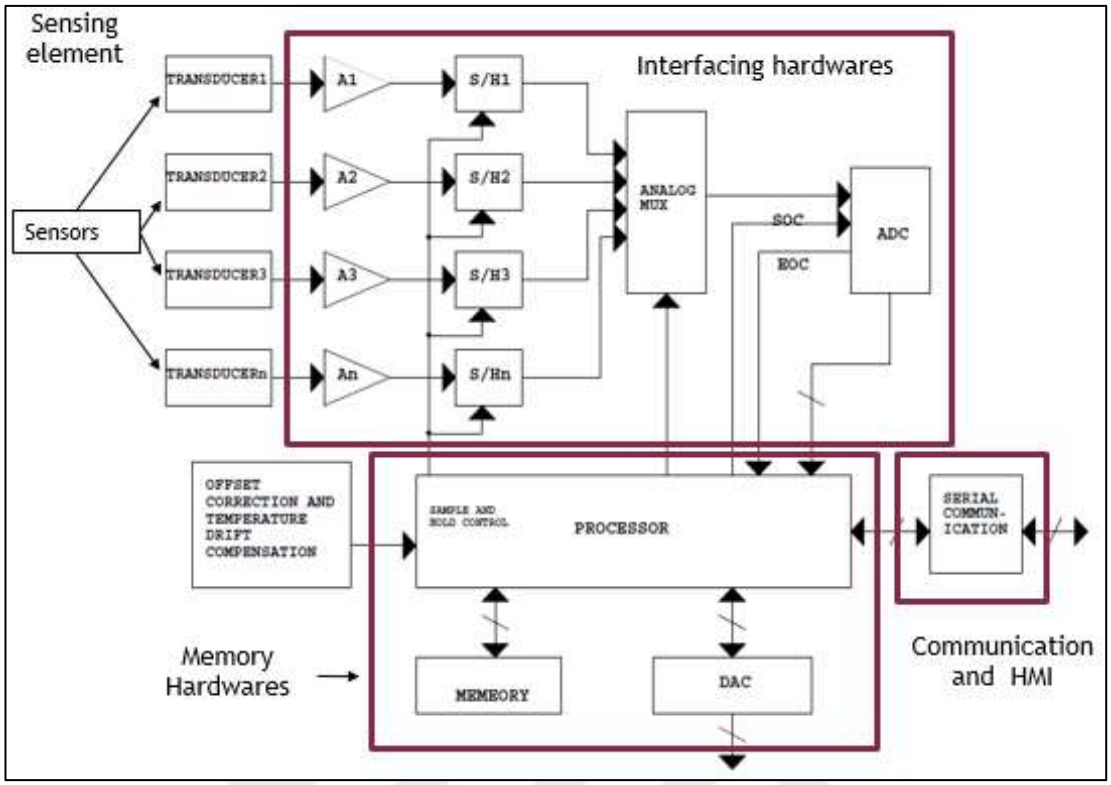

Figure 3.3: Basic Architecture of a Sensor [3].

Sensor: The capture unit or sensor element is made up of two sub-elements:

a. The sensor retrieving analog data,
b. A converter converting the analog data from the sensor to digital data (called ADC for analog to digital converter) sent to a calculation unit.

Calculation unit: The component includes:

a. A processor,
b. A reduced memory unit.

It stores data, performs the tasks of perception assigned to it.

Transmitter/Receiver: element used to connect the sensor to the network.

Power supply: the source of energy for the sensor, like any on-board device, they have an autonomous power supply such as a battery.

3.5 ARCHITECTURE OF AN WSN

The architecture of the sensor network is shown in the following figure (Figure 3.4). The user remotely accesses captured data through a node called the Task Manager Node. The task director node is connected to the Internet or to the satellite through a "sink" destination node. The latter acts as a gateway for the sensor network, i.e. it connects sensor networks to

other networks. This node is responsible, in addition to collecting reports, for broadcasting requests on the types of data required to the sensors via request messages. It also has other information processing capability for further processing if required. Sensor nodes are usually dispersed in a capture area called a wellfield. The sensor nodes gather the data and convey it to the recipient. In this way, users can search the information in destination nodes to monitor and control the environment remotely. Note that a network of sensors can contain several sink nodes broadcasting different interests (these are the description of the data required by the destination node using a combined attribute-value designation). For example, one well node may request all sensors in the northern region of the well field to send a temperature report every 1 minute, while another may be interested only in high temperatures (> 40° C) in the southern region. Therefore,

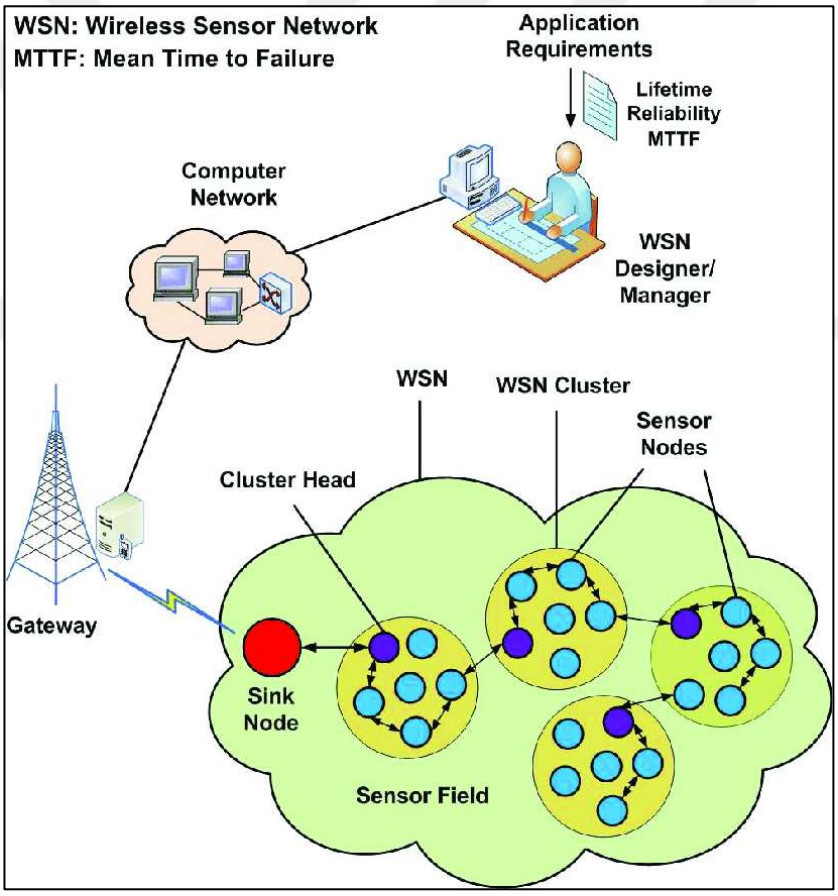

Figure 3.4: Architecture of a Sensor Network.

3.5.1 The Protocol Stack in an WSN

The role of this stack is to standardize communication between participants so that different manufacturers can develop compatible products (software or hardware).

This model includes 5 layers which have the same functions as those of the OSI model as well as 3 layers for power management, mobility management and task management.

The purpose of a layered system is to separate the problem into different parts (the layers) according to their level of abstraction.

Each layer of the model communicates with an adjacent layer (the one above or the one below). Each layer thus uses the services of the lower layers and provides them to the higher level.

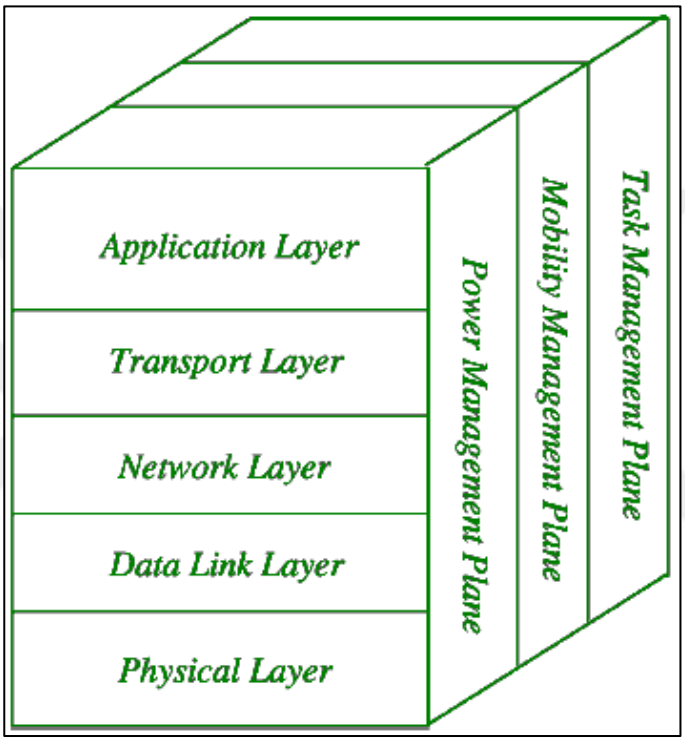

Figure 3.5: Protocol Stack of a Sensor Network Architecture [9].

3.5.2 The Physical Layer

Takes care of the specification of cabling, carrier frequencies, etc. …

This layer must provide data transmission, reception, and modulation techniques in a robust manner.

3.5.3 The Link Layer

It specifies how data is forwarded between two nodes/routers within a one-hop distance. It is responsible for data multiplexing, error control, media access, etc.

It provides the point-to-point and point-to-multipoint link in a communication network.

It is composed of the Logical Link Control (LLC) layer which provides an interface between the link layer and the network layer by encapsulating network layer message segments with

additional header information, and the Medium Access Control (MAC) layer that controls the radio.

As the environment of sensor networks is noisy and the nodes can be mobile, the data link layer must guarantee low power consumption and minimize the collisions between data broadcast by neighboring nodes.

3.5.4 The Network Layer

This layer makes it possible to manage the addressing and the routing of the data, that is to say their routing via the network.

3.5.5 The Transport Layer

This layer is responsible for transporting the data, breaking them up into packets, controlling the flow, preserving the order of the packets and managing any transmission errors.

3.5.6 The Application Layer

This layer provides the interface with the applications. It is therefore the level closest to the users, managed directly by the software.

3.5.7 The Energy Management Level

The functions integrated at this level consist in managing the energy consumed by the sensors. Therefore, a sensor can for example switch off its reception interface as soon as it receives a message from a neighboring node in order to avoid the reception of duplicate messages. Moreover, when a node has a low energy level, it can broadcast a message to the other sensors not to participate in the routing tasks, and conserve the remaining energy to the capture functionalities [3].

3.5.8 Mobility Management Level

This level detects and records all movements of the sensor nodes, so as to allow them to continuously keep a route to the end user and maintain a recent image on neighboring nodes. This image is necessary to be able to balance the execution of the tasks and the energy consumption [3].

3.5.9 The Level of Task Management

During a capture operation in a given region, the nodes making up the network do not necessarily have to work with the same rhythm. This essentially depends on the nature of

the sensor, its energy level and the region in which it has been deployed. For this, the task management level ensures the balancing and distribution of the tasks on the different nodes of the network in order to ensure a cooperative and efficient work in terms of energy consumption, and consequently, to extend the service life. of the network [3].

3.6 CHARACTERISTICS OF A SENSOR

A sensor has the following characteristics:

a. Able to calculate
b. Able to communicate
c. always picks up
d. Random preposition/deployment
e. Battery Life Limitation
f. Density (small / large number)
g. Speed: this is the reaction time of a sensor between the variation of the physical quantity it measures and the instant when the information taken into account by the control part.
h. The extent of the measurement: this is the difference between the smallest signal detected and the largest perceptible without risk of destruction for the sensor.
i. Sensitivity: this is the smallest variation of a physical quantity that a sensor can detect.

3.6.1 Application of a WSN

Like much technology, the development of WSNs was prompted by military needs. Indeed, armies want to be able to discreetly spy on their enemies. The absence of cables between the nodes, their small size, the high number of deployable sensors to cover a large area meet these criteria. Thus, several applications have been realized including the Sound Surveillance System or the Distributed Sensor Network [6].

Then, the decrease in the manufacturing costs of the sensors as well as the reduction in their size led to their use in civil applications. For example, they can be used for environmental monitoring purposes. Indeed, sensors can be placed in glacial or tropical regions in order to accurately monitor the effects of global warming, climate change or increased pollution [6]. They can also be used for habitat monitoring because their deployment, for example in the mountains, could make it possible to identify the animals frequenting a given territory. Industrial applications can also adopt them. An idea of use could be to place these instruments at specific points, for example at the wear points of the machines, which would

make it possible to issue alerts when their general condition deteriorates [7]. Finally, some plan to implant sensors in the human body. This would make it possible to monitor the state of health of patients and thus to adapt their treatment, to prevent the deterioration of their state of health and consequently to be able to anticipate emergency hospitalization [8].

3.7 STANDARD SENSOR NETWORKS

Several product lines are currently on the market, but standardization could still change things. The working groups responsible for this standardization come from the IEEE in the United States and from ETSI in Europe. Figure 3.6 describes the different categories of networks according to their extent and Figure 3.7 the existing standards.

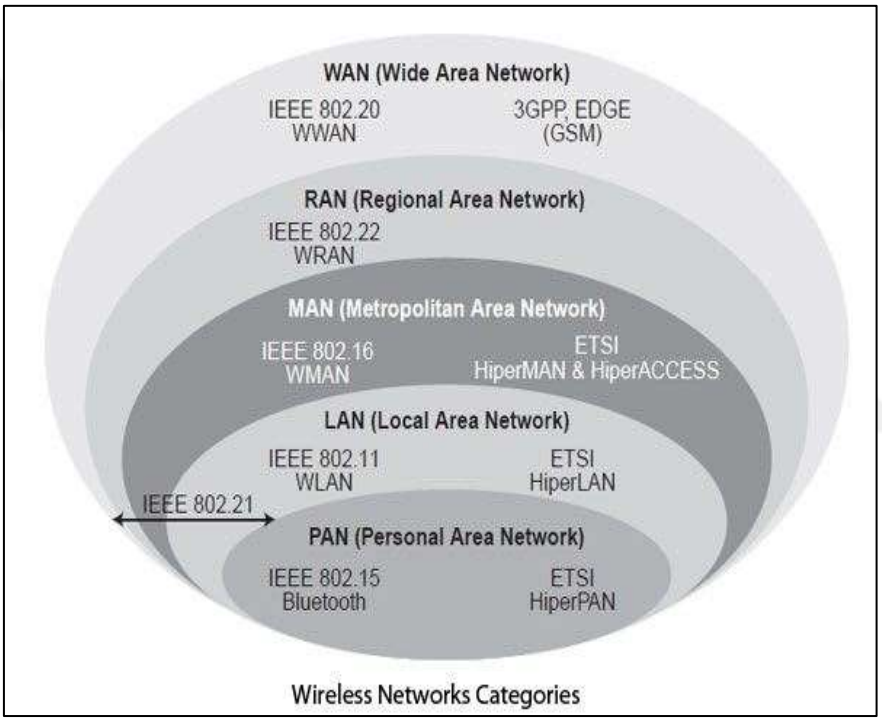

Figure 3.6: Categories of Wireless Networks [20].

The main standards are IEEE 802.15, for small personal networks with a range of ten meters, IEEE 802.11, or Wi-Fi (Wireless-Fidelity), for WLAN networks (Wireless Local Area Network), IEEE 802.16, for WMAN networks (Wireless Metropolitan Area Network) reaching more than ten kilometers, and IEEE 802.20, for WWAN networks (Wireless Wide Area Network), that is to say very large networks.

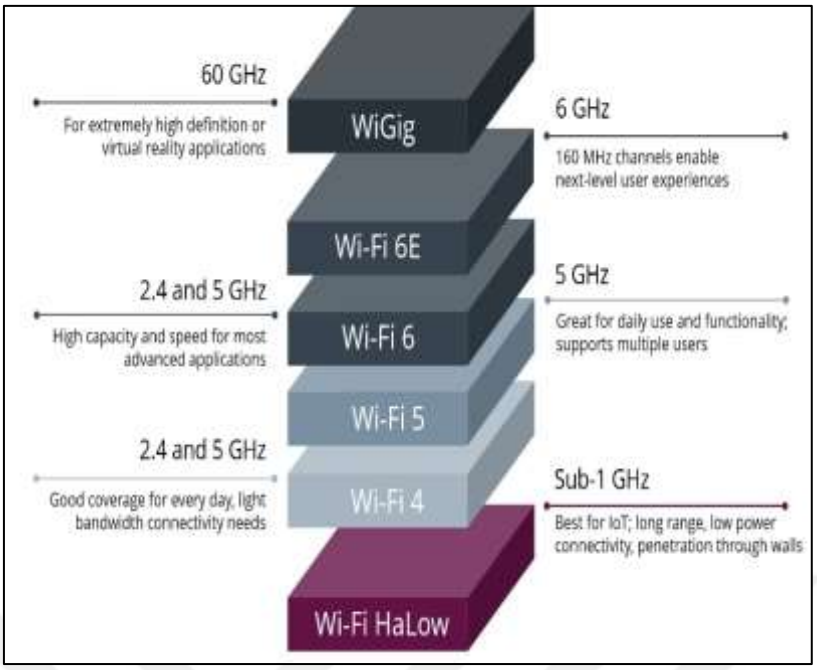

Figure 3.7: Main Wireless Network Standards [20].

In the following, we will present the standard WSNs that have been studied in the literature.

3.7.1 IEEE 802.15.1

IEEE 802.15.1, the best known, supports the Bluetooth standard, which is widely marketed today. But this standard is rarely used in WSNs because of its high power consumption.

3.7.2 IEEE 802.15.3

IEEE 802.15.3 defines the UWB (Ultra-Wide Band) standard, which implements a very special technology, characterized by transmitting at extremely low power, below ambient noise, but over practically the entire radio spectrum (between 3.1 and 10.6 GHz). The speeds reached are of the order of a gigabit per second over a distance of 10 meters [20].

3.7.3 IEEE 802.15.4

Wireless micro-sensor networks have been the subject of intensive research in recent years and are now emerging in industrial applications. An important step in this transition has been the release of the IEEE 802.15.4 standard which specifies interoperability in the physical layer and the MAC (Medium Access Control) layer targeting the transmitting radio of the sensor node [21, 22]. . The IEEE 802.15.4 standard supports different network topologies. In this standard, two types of topologies are discussed: "Star" topology and "Peer-to-peer" peer-to-peer topology. The IEEE 802.15.4 standard presents two types of nodes: a node with

a full load (Full Function Device (FFD)) and a node with a reduced load (Reduced Function Device (RFD)).

The standard indicates that the network is coordinated by one of the FFDs, the latter can route data (unlike the RFD). In this standard, the star topology emphasizes battery life since each RFD is connected directly to the coordinator. On the other hand, the pair-to-pair topology is concerned with reliability and scalability since all the nodes are FFDs and can therefore be linked together. The IEEE 802.15.4 standard can support other topologies, for example the "Cluster tree" cell tree topology which combines the two preceding topologies (star and pair to pair or meshed "Mesh"). The different network topologies supported by IEEE 802.15.4 are shown in the following figure [23, 22]:

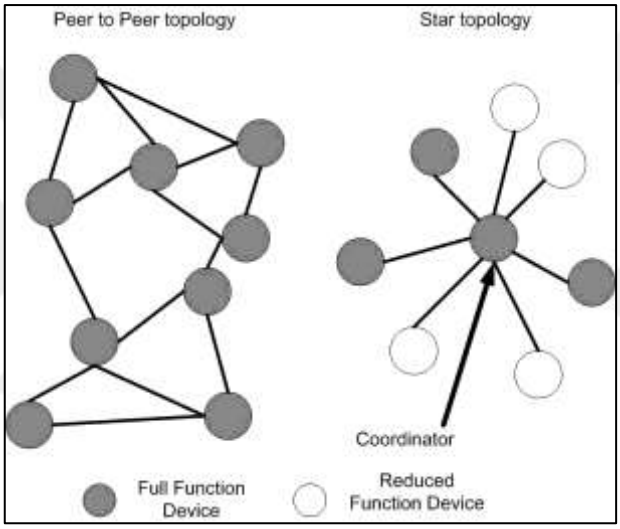

Figure 3.8: Network Topologies Supported by IEEE 802.15.4 [24].

3.7.4 ZigBee Alliance

In 2002, ZigBee Alliance was formed by an association of companies. The goal of ZigBee Alliance is to develop reliable control products with a reduced cost, and which do not require a lot of energy. These products must be able to be managed by a wireless network using a global standard. Zigbee Alliance operates globally on the 2.4 GHz frequency band but also at 915 MHz in America and 868 MHz in Europe. The speeds offered are: 250 Kbits/s at 2.4 GHz (10 channels), 40 Kbits/s at 915 MHz (6 channels) and 20 Kbits/s at 868 MHz (1 channel). ZigBee Alliance connects up to 255 devices per network over a range of up to 100 meters. ZigBee Alliance was ratified in August 2003 under the IEEE 802.15.4 standard [24].

3.7.5 IEEE 1451.5

The IEEE 1451.5 family of standards is supported by the Measurement Society's Institute of Electrical and Electronics Engineers (IEEE) Sensor Instrumentation Technology Technical Committee. IEEE 1451.5 was initiated to develop a standard for wireless communication methods and the data format of transducers (A transducer: is a means that allows the conversion of energy from one type to another for example, magnetic energy to electronic energy and vice versa).The purpose of this standard is to develop smart transducers[19] for sensors. Several wireless communication interfaces and protocols are developed for sensor networks by different manufacturers. These communication interfaces and protocols are specified by suppliers. IEEE 1451.5 supporting the various existing technologies will increase market acceptance and enable connectivity between devices from different vendors [23].

3.8 CONCEPTUAL FACTORS AND CONSTRAINTS OF WSNS

The design of WSNs, their protocols and algorithms are guided by several factors:

3.8.1 Fault Tolerance

The network must be able to maintain its functionality without interruption in the event of failure of one of its sensors. This failure may be caused by power loss, physical damage or environmental interference. The degree of tolerance depends on the degree of criticality of the application and the data exchanged.

A first challenge will therefore be to identify and formally model the failure modes of the sensors, then to rethink the fault tolerance techniques to be implemented in the field.

The reliability RK(t) of a sensor node is modeled in [9] by a Poisson distribution to capture the probability of not having a failure within a time interval (0; t):

$$RK(t) = \exp(\lambda K\, t) \qquad (3.1)$$

Where: λK is the failure rate of sensor node k and t is the time period.

3.8.2 Scale (Scalability)

One of the characteristics of WSNs is that they can contain hundreds or even thousands of sensor nodes. The network must be able to operate with this number of sensors while allowing the increase in this number and the concentration (density) of nodes in a region (which may exceed 20 nodes/m^3).

Such a large number of nodes generates a lot of internodal transmissions (implementation of error detection, flow control, etc.) and requires the sink to be equipped with a lot of memory to store the information received.

3.8.3 Operating System

TinyOS is among the open-source operating systems for sensor networks designed by the American University of BERKELEY. The open-source nature allows this system to be regularly enriched by a multitude of users. Its design was entirely carried out in NesC, a component-oriented language syntactically close to C. It respects an architecture based on an association of components, thus reducing the size of the code necessary for its implementation. This is in keeping with the memory constraints observed by sensors provided with very limited resources due to their miniaturization. However, the TinyOS component library is particularly complete since it includes network protocols, sensor drivers and data acquisition tools. A program running on TinyOS consists of a selection of system components and components developed specifically for the application for which it will be intended (temperature measurement, humidity rate, etc.). TinyOS relies on event-driven operation, ie it only becomes active when certain events occur, for example the arrival of a radio message. The rest of the time, the sensor is in a standby state, guaranteeing maximum lifespan knowing the low energy resources of the sensors. that is to say that it only becomes active on the appearance of certain events, for example the arrival of a radio message. The rest of the time, the sensor is in a standby state, guaranteeing maximum lifespan knowing the low energy resources of the sensors. that is to say that it only becomes active on the appearance of certain events, for example the arrival of a radio message. The rest of the time, the sensor is in a standby state, guaranteeing maximum lifespan knowing the low energy resources of the sensors.

3.8.4 Limited Physical Security

Wireless sensor networks are more affected by the security parameter than conventional wired networks. This is justified by the physical constraints and limitations which mean that the control of the transferred data must be minimized.

3.8.5 Production Cost

The cost of producing a single sensor is very important for evaluating the overall cost of the network. If the latter is greater than that necessary for the deployment of conventional sensors, the use of this new technology would not be financially justified. Therefore, reducing the production cost to less than $1 per node is an important goal for the feasibility of the wireless sensor network solution [10].

3.8.6 The Environment

The sensor nodes can be deployed near or inside the observed phenomenon. They can thus operate in remote geographical regions and under certain constraints, such as: congested intersections, tornadoes, biologically or chemically contaminated surfaces, attached to animals, etc.

3.8.7 Network Topology

Deploying a large number of nodes requires topology maintenance. This maintenance consists of three phases:

a. Deployment
b. Post-deployment (sensors may move, stop working, etc.)
c. Redeployment of additional nodes [9].

3.8.8 Material Constraints

As mentioned earlier, a sensor node can contain other network application-dependent units. Indeed, most capture operations and routing algorithms in wireless sensor networks require knowledge of the location of nodes with high precision, because these nodes are deployed randomly and operate in a random way. autonomous way. This makes the integration of a unit, dedicated to the localization system, very common in a sensor node. This is the reason why often these nodes have a GPS location system, although it has been shown that this solution is not reliable for wireless sensor networks. Another approach proposed in [11] consists in equipping a limited number of nodes with the GPS system, and help other nodes find their positions in a terrestrial way. The design of the sensor nodes can go so far as to provide a sensor mobility system to move it if necessary. All of these units may require integration into an enclosure with a minimum size of less than one cubic centimeter, and with a very light weight that allows the nodes to remain suspended in the air, if the

application requires it. Apart from the size, there are other demanding constraints for the construction of the sensor nodes which are: and with a very light weight that allows the nodes to remain suspended in the air, if the application requires it. Apart from the size, there are other demanding constraints for the construction of the sensor nodes which are: and with a very light weight that allows the nodes to remain suspended in the air, if the application requires it. Apart from the size, there are other demanding constraints for the construction of the sensor nodes which are:

a. Consume as little energy as possible
b. Operate in a high density
c. Have a reduced production cost
d. Be autonomous and able to operate without assistance
e. Be adaptive to the environment [10].

3.8.9 Transmission Medium

The communicating nodes are connected wirelessly. This link can be made by radio, infrared signal or optical media. [9].

It is necessary to ensure the availability of the transmission medium chosen in the capture environment in order to allow the network to accomplish all of its tasks. For communication links via radio frequencies, the ISM bands (Industrial Scientific Medical bands) can be used. For sensor networks, the integrated transmission units at the nodes must be small in size and low in power consumption. Indeed, the material constraints associated with the nodes, as well as the existing compromise between the efficiency of the antennas and the energy consumption, limit the choice of the frequency band used on the high frequency bands [10].

3.8.10 Connectivity

A sensor network is said to be connected if and only if there is at least one route between each pair of nodes [12]. Connectivity essentially depends on the existence of roads. It is affected by topology changes due to mobility, node failure, attacks, etc. This has the following consequences: the loss of links, the isolation of nodes, the partitioning of the network, the updating of routes (routing), . . . etc

A graph G is said to be k-connected if there are at least k disjoint paths between any two nodes. Connectivity is a measure of fault tolerance or path diversity in the network. 1-

connectivity is a fundamental requirement for the network to be operational. Indeed, the connectivity of a network is expressed as follows [7]:

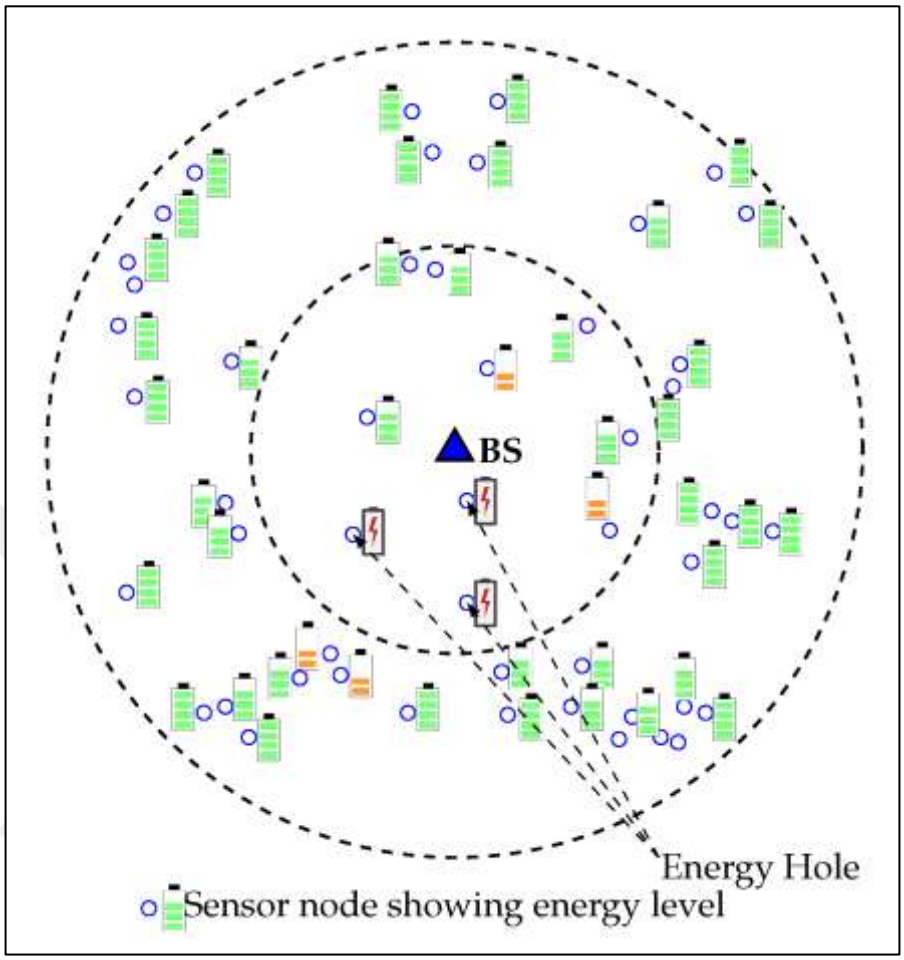

Figure 3.9: Sensor Node Showing Energy Level.

Where: • R is the transmission radius of a node

a. At the computing area

b. N the number of nodes located in area A

3.8.11 Energy Consumption

As the sensor nodes are micro-electronic components, they can only be equipped with limited sources of energy (<0.5 Ampere-hour, 1.2 V). Also, in some applications, these nodes cannot be provided with energy recharging mechanisms, therefore the lifetime of a sensor node is highly dependent on the lifetime of the associated battery. Knowing that sensor networks are based on multi-hop communication, each node plays both the role of data initiator and router as well. Failure of a number of nodes results in a significant change

to the overall network topology, and may require different packet routing and total network reorganization. This is why the power consumption factor is of paramount importance in sensor networks. Most of the research currently being carried out focuses on this problem in order to design algorithms and protocols specific to this type of network that consume the minimum amount of energy.

In sensor networks, energy consumption efficiency represents a significant performance metric, which directly influences the lifetime of the entire network. For this, designers can, when developing protocols, neglect other performance metrics such as transmission time and throughput, in favor of the energy consumption factor [10]. Indeed, the energy consumption in a WSN can be divided according to the following decreasing order [13]: Communication (transmission and reception), data processing, and acquisition or capture.

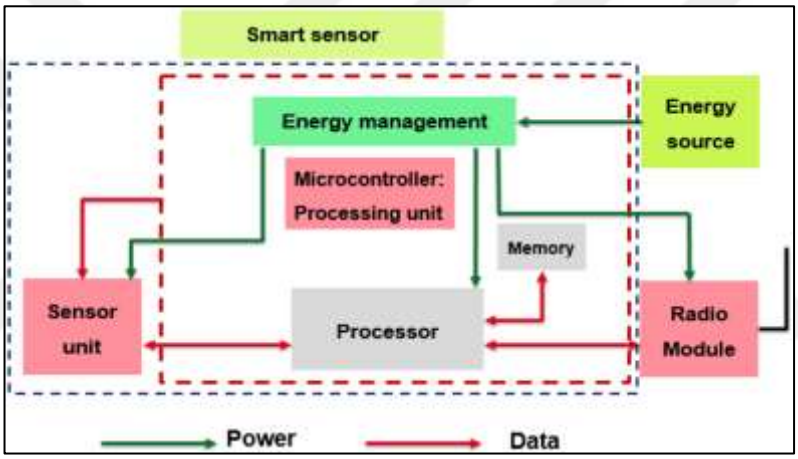

Figure 3.10: Energy Consumed by The Subsystems of a Sensor Node.

Where RX stands for energy for reception, TX for transmission, IDLE for channel listening, SLEEP is sleep state.

3.8.11.1 Capture energy

This task is carried out by the capture unit (Figure 3.3) which translates the physical phenomena into an electrical signal and it can be digital or analog. There are several types of this component which measure environmental parameters like temperature, sound, image, pressure, etc. Sources of power consumption in these components can be: signal sampling, conversion of physical signals to electrical signals, signal processing and analog-to-digital conversion. The energy consumption by these components is dependent on their tasks, temperature or earthquake sensors are less energy consuming compared to imaging or video [14].

Solution. The power consumed during capture can be reduced by using low-power components but thus reducing their performance. Or by eliminating unnecessary capture by reducing capture times.

3.8.11.2 Energy of treatments

This task includes controlling capture components and running communication protocols and signal processing algorithms on collected data. It is performed by microprocessors. The choice of the latter is based on the application scenario, and it generally makes a trade-off between performance level and power consumption [14].

Solution. There are two approaches for energy minimization when processing data by a sensor node:

a. The system partitioning approach: consists of transferring a calculation that is prohibitive in terms of calculation time to a base station which has no energy constraints and which has a large calculation capacity.
b. The DVS "Dynamic Voltage Scaling" approach consists of adaptively adjusting the supply voltage and the frequency of the microprocessor to save computing power without performance degradation [18].

3.8.11.3 Communication energy

Communication energy represents the largest proportion of the total energy consumed at a node [15]. This communication is provided in most WSNs by the radio transmission medium. The power consumption of the latter is affected by several factors: the type of modulation system, the amount of data to be communicated, the transmission power (determined by the transmission distance), etc. [22, 16].

In general, radios can operate in four different modes of operation: transmit (the radio is transmitting a packet), receive (the radio is receiving a packet), idle (the radio is on, but not used) and dormant (The radio is turned off). It has been shown that the radio consumes much more power in the transmit and receive modes. However, the active mode is also energy intensive. In most cases, the power consumption is relatively high in the active mode, since the latter requires the radio module to be powered up and continuously decode the radio signals to detect the arrival of the packets.

Solution. Energy minimization during communication is mainly related to protocols developed for the MAC layer [17] and the network layer. The purpose of the protocols of

the latter is to find the optimal routes in terms of energy consumption. Indeed, the loss of energy due to poor routing of data packets has an impact on the lifetime of the network.

3.9 SENSORS IN PICTURES

The sensors are available in a multitude of models in relation to the application for which it is intended. Among the most common models, there is for example the "weC" sensor from the University of Berkeley to capture temperature and brightness. Figure 3.10 shows the images of some sensors.

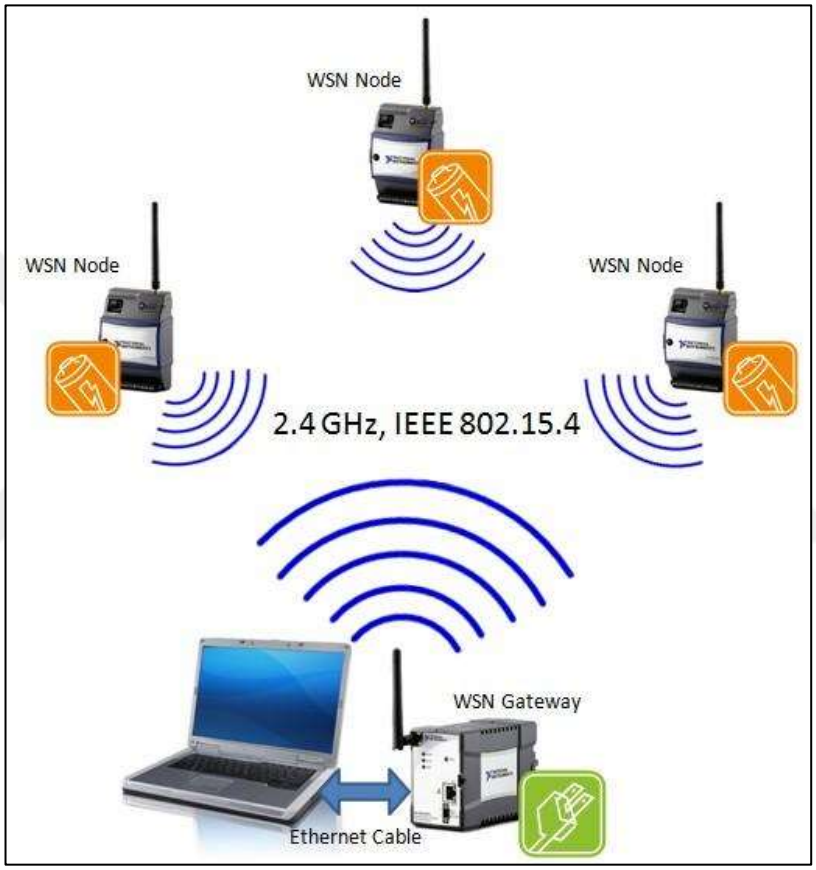

Figure 3.11: Wireless Sensor.

Sensor networks remain a new technology that is not easily accessible to the general public. It is mainly prevalent in research laboratories.

Progress is still to be made in this area. Nevertheless, they correspond to a certain vision of the future and will allow improvements in innumerable areas of daily life.

4. PROPOSED METHOD

4.1 INTRODUCTION

Intrusion detection is one of the most studied security problems, with many papers and researcher's contributing new methods yearly. Despite this some reports estimate that as much as 50%-80% of the world's spam is sent from zombie computers, or computer that have been taken control of through an security intrusion One approach that hasn't been applied as extensively is trying to identify if the outgoing traffic from a computer fits a normal range for that user based on the NLP of the sent messages. Two approaches to this task are building language models such as HMM with semantic analysis, and another is using NLP techniques to build a feature set, and then using machine learning to classify intrusive behavior. In this paper WE will discuss and outline how we went about testing this hypothesis for the machine learning approach.

4.2 DATASET

Originally WE wanted to use a collection of email data for the training and testing sets, but after some experimenting with the Enron email datasets We found them to have two problems.

Firstly, the dataset was rather sparse on a user by user basis. Using a frequently cited dataset provided on August 21, 2009 at http://www.cs.cmu.edu/~einat/datasets.html we was able to collect 700 email messages. From these we wrote a script to separate them into files by author, using the first "From:" token in the email as the indication of the sender, and their email as their named key. After running this code we found that, when excluding the primary Enron announcement email, the user with the most sent emails in the dataset had only 7 emails to his name. The dataset we need for intrusion detection needs to be large and exhaustive, so that small amounts of intrusive messages can be inserted into it. If the dataset was 8 emails with 7 legit and 1 intrusive it wouldn't be very conclusive.

Secondly, the emails are raw text are rather poorly formed. There were many nested, forwarded emails which were difficult to parse and remove such that we were only analyzing the characters typed in the keyboard by the user sending the email. In addition, there was lots of non-human written text in the emails such as ID numbers and header information, this again proved a parsing challenge.

Our first solution to this problem was to use twitter feeds as a different source of human generated input. From open twitter datasets it is possible to collect thousands of tweets from individual people. we found that this would work, but the character limit and frequent use of hashtags may not correlate to a good dataset. The final solution WE settled on was to utilize the social networking site WSN data. WSN data is a popular forum board with hundreds of millions of users, many of them with more than 5 years of public history of comments and posts with many thousands of entries. WSN data doesn't have character limits, yet still has blends of formal and informal dialog depending on which board you post to. In order to get this dataset we wrote a MATLAB scrapper to fetch the X most recent comments posted by user Y to WSN data using the websites API. For our experiments WE fetched 1000 posts from our own WSN data user account, and paired it with 100 posts from another random users WSN data account. From this base line we pruned 2 entries from the set of 1000 due to formatting errors and was left with a dataset of 1098, in a near 10 to 1 ratio of legitimate posts and hypothetical intrusive posts. The data was fairly rich, containing 7,313 unique words with the following distribution of data.

Table 4.1: Dataset Metrics. The Code for Metric Extraction is Datasetm.

METRIC	VALUE
AVERAGE NUMBER OF WORDS/ENTRY	43
SHORTEST ENTRY	1
LONGEST ENTRY	646
STANDARD DEVIATION OF WORDS/ENTRY	53.538
NUMBER OF UNIQUE WORDS:	7313

In order to recollect this data, or collect another set of data the data generation MATLAB files are in the attached folder. To extract a set of user comments from WSN data use the following command line argument:

MATLAB DatasetCollector.m <WSN data account name> <number of comments to collect>

These comments will be stored in an xml file with the users name.xml as the file. The users account name must be a valid WSN data username. For the dataset used in this paper WE used the following code:

Superbug is the username of a random person we found on WSN data. we didn't look at any of his data before querying to eliminate selection bias. WE verified that his account had more than 100 posts and then collected it. WE put both files in the TrainingData folder. Next, to generate the complete data set to read into MATLAB ML LIBRARY WE ran the MATLAB dataset generator file pointing to the folder. The DatasetGenerator.m file takes two command line arguments, the first is the path to a folder holding the data, the second is the name of the output file. It will take every item out of the folder and extract them out of the XML format (output from the scrapper) and put them in a single CSV format containing two columns, entry and username. This CSV file can be read into MATLAB ML LIBRARY in order to do preprocessing. For this dataset WE ran the following:

MATLAB DatasetGenerator.m TrainingData Dataset1.csv

Using this process many datasets could be generated from WSN data users, in any combination of users and entries by simply calling these programs.

4.3 PROPOSED APPROACH

The machine learning method was able to fairly successfully classify the authors based on the preceding work, but the machine learning approach leaves out semantic meaning from the phrases, and thus might be further improved upon by more sophisticated natural language methods. In order to test this hypothesis we conducted an experiment using Authorship profiling using the common N-grams (CNG) distance metric. we evaluated this method using three feature sets: words, parts of speech, and synsets from the WordNet library. For each comment in the data set the distance from the normal authorship profile was calculated using the CNG approach and then these distance metrics were used to train a classifier to find and detect the intrusive messages.

4.3.1 The Process and Model

The model we used was loosely based off of this diagram:

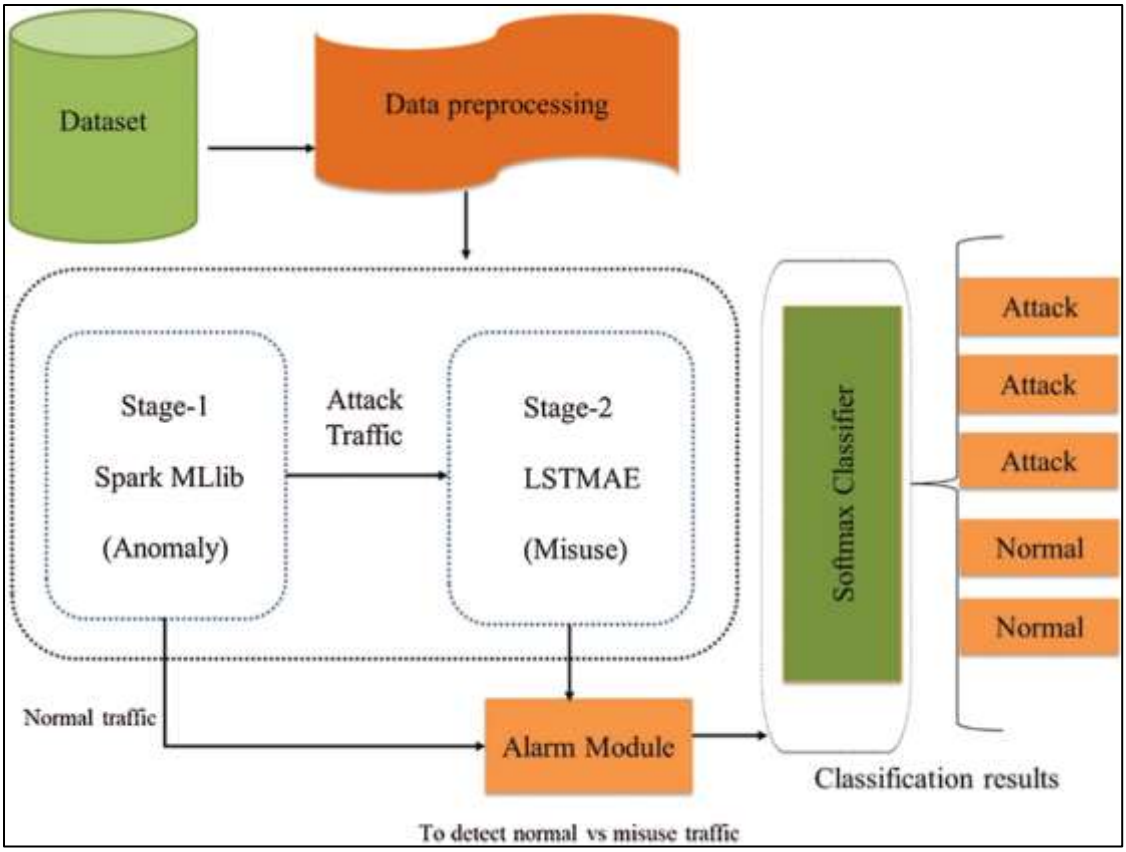

Figure 4.1: Initial Proposed Model.

The change made to this model was that in our data set there was no training for a second B author. This is because in a real system employed at an organization you won't be able to build a data set of intrusive messages to train from. The idea is to find outliers that are so far from the normal that they become detected, not classify them as an intruder already seen before.

A more thorough model of the entire process we actually used is given in the image below. This model shows the data collection from WSN data, the authorship profiling phase, the distance calculation phase, and finally the output to MATLAB ML LIBRARY for building the classifier.

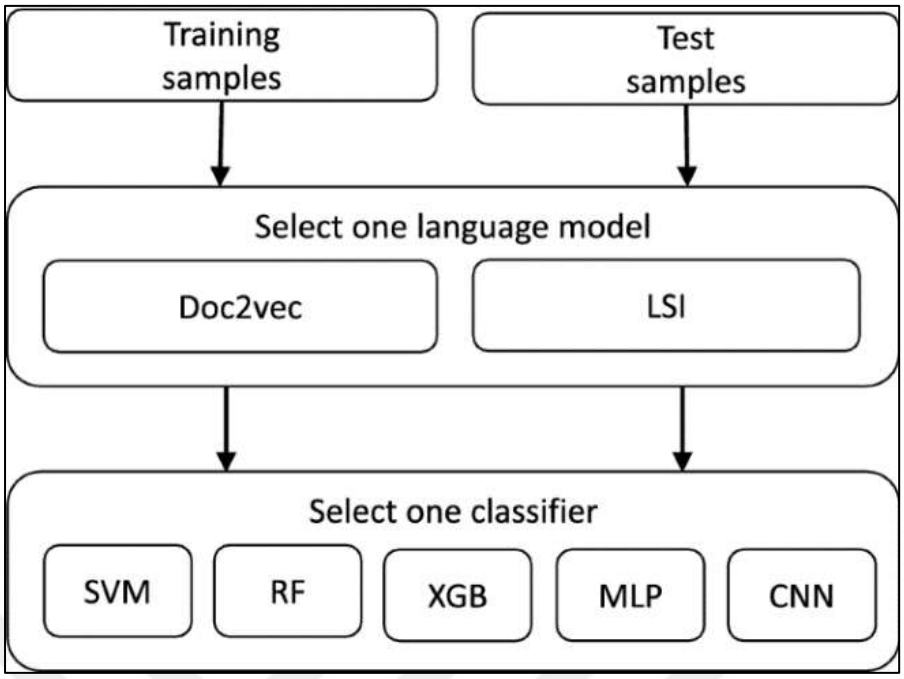

Figure 4.1: Final Model for Building The NLP Intrusive Detection Classifier.

4.3.2 Common N-Grams Distance Metric

The Common N-Grams distance metric was first outlined in Kessel et al, in PACLING 2003: N-gram-based author profiles for authorship attribution. They use the following formula to calculate the distance between any two documents x and y. we used this same metric to train our classifiers using the NLP techniques. WE chose to use the top 5000 N-grams from the training corpus of 1000 legitimate author comments.

$$d(PR(x), PR(y)) = \sum_{g \in P(x) \cup P(y)} \left(\frac{2(f_x(g) - f_y(g))}{f_x(g) + f_y(g)} \right)^2 \quad (4.1)$$

4.3.3 N-grams Comparisons

WE tried three methods of N-grams for the three features words, parts of speech, and synsets. WE created a unigram, bigram, and trigram dataset for each of these features and then ran information gain feature selection to rank them in order of importance. From this WE found that the Synsets were the most valuable, followed by the words, and lastly followed by the parts of speech. This relationship held true across all three N-gram data sets. Additionally, WE found that the trigram data set was better able to classify the data into intrusive and normal comments than either of the other two data sets.

4.3.4 Pre-Processing

Now that we have a CSV file with user names as classes and entries as strings we decided to apply some NLP techniques to the data. Luckily the MATLAB ML LIBRARY tool has some built in tools for NLP use. In order to import the CSV file from above into MATLAB ML LIBRARY, go into the GUI and click open file, and then navigate to the file. You will likely receive an error message, instead of clicking OK click “Use Converter” you will need to specify that the first column is a string variable by type by typing in a 1 in the field for “String attributes”. After this press OK and the program should read the data in without a problem. WE wanted to apply NLP techniques that we discussed in the class on the data set first, so WE chose to apply the StringToWordVector filter in MATLAB ML LIBRARY. In the configuration for this filter WE chose to apply the filter to only the first attribute, and then WE set “lowerCaseTokens” to true. Next WE applied the IteratedLovinsStemmer, the Rainbow stopwords removal algorithm, and the alphabetic tokenizer. The stemmer and stopword removal algorithm are discussed in detail in the MATLAB ML LIBRARY docs, and are based off of research papers in those areas of study. The alphabetic tokenizer creates tokens from the sentences by extracting all contiguous sequences of alphabetic tokens. In other words, every character that isn’t “a-z” is treated as a delimiter. Note: The dataset collectors remove all punctuation from the data to remove the issue of conjunctions. Lastly, WE set the outputWordCounts to true, and wordsToKeep to a high value such that no words are lost. Click apply to create the new data set (Found in the submission as Dataset1_FeatureExtracted.arff). WE call this data dataset we. next we took this data, and decided to run it through a feature selection algorithm. WE used two different methods for feature selection. In the first method we used a statistical significance test to find the most likely tokens, this reduced the token count from 5000+ to 324. The other feature selection method was to run a GreedyStepwise search on subsets of the data set using MATLAB ML LIBRARYs tools. This selection resulted in a dataset 59 features. Lastly, We repeated the above processes to generate a set of data from the original CSV file, using the same stemmer and stopword removal, but instead generating all unigrams to trigrams of the words as features. we then reduced this dataset using the same statistical feature selection method as above.

Table 4.2: Breakdown of Datasets Used and Their Construction Methods. Under Construction Methods The First Point is the Tokenizer Used, and The Second is the Feature Selection Method Used.

Dataset	File in Submission Folder	Number of Features	Construction Method
Dataset I	Dataset1_FeatureExtracted.arff	3322	- Alphabetic - No Selection
Dataset II	Dataset1_StatisticalFeatureSelection.arff	324	- Alphabetic - Statistical
Dataset III	Dataset1_SubsetFeatureSelection.arff	59	- Alphabet - CfsSubsetEval
Dataset IV	Dataset1_NGrams_SFE.arff	4431	-NGrams(1-3) - Statistical

5. RESULTS

From these four data sets we conducted experiments using the MATLAB ML LIBRARY GUI for many different machine learning algorithms. For all of the following results WE report 10-fold cross-validation test results using Accuracy and F-scores as metrics for determining how effective a classifier was in detecting the intrusion. From the collection of experiments WE chose to report the results of the following, because they stood out as have the highest F-score for the detection of the intrusive messages, while maintaining a high accuracy rate with low false positives: Logistic Regression, NaiveBayes, NaiveBayesMultinomial, Support Vector Machines trained with Stochastic Gradient Descent, and a Voted Perceptron Neural Network system. All of these classifiers and their functionality are described in detail in the MATLAB ML LIBRARY documentation and are included in the basic MATLAB ML LIBRARY 3.7 package download, so we won't go into details of how the algorithms work.

5.1 NLP RESULTS

To evaluate the success of the NLP method we ran a 10 fold cross validation on the same original data as the machine learning methods in the previous part of this paper. we experimented with different model n-gram sizes, and different combinations of features. Our best data set was the trigram model with all three CNG distances included. WE found that using logistic regression classifier on this set of data was fairly accurate, and on par with the average machine learning methods employed in the first part.

Table 5.1: Best Classifier for NLP with CNG Distances.

Notable Methods	Accuracy	F-Measure (Legitimate)	F-Measure (Intrusive)
Logistic Regression	92.81%	0.961	0.470

For the base data set with no feature selection, the NaiveBayesMultinomial (NBM) performed slightly better than the SVM trained using the SGD. The NBM had an accuracy of 91.62% and an F-measure of 0.954 (Legit class) and an F-measure of 0.549 (Intrusive class). These results were not great, but were the best of the set of attempts using no feature selection. we also tried an SVM trained used SGD. This resulted in 91.71% accuracy, but

the F-measure of the intrusive class was only 0.428. The F-score of the legitimate class was 0.955. we tried various other learning methods such as JRip, and PART rule based classification methods, as well as the J48 decision tree, and a random forest with 100 trees and all of these methods had F-measures below the NBM and the SVM. WE tried polynomial kernals on the SVM training with orders 1, 2, and 3 and found the best results at order 1, leading me to believe the data is linearly separable. (Up to the results we achieved with the SVM).

Table 5.2: Summary of Results: Dataset I.

Notable Methods	Accuracy	F-Measure (Legitimate)	F-Measure (Intrusive)
SVM with SGD	91.71%	0.955	0.428
NaiveBayes Multinomial	91.62%	0.954	0.549

Table 5.3: Comparing our Method to NB on Dataset I.

Notable Methods	Confusion Matrix	
SVM with SGD	973	25
	66	34
NaiveBayes Multinomial	950	48
	44	56

For the second data set, the two classifiers with above average performance were the NBM again, along with a voted perceptron system. In general the second data set performed the best of all four data sets, across most classifier methods. For the voted perceptron system 369 perceptrons were trained over 10 iterations. The MATLAB ML LIBRARY default of 1 for the exponent and kMAX parameters were used. The Voted Perceptron system on the statistically pruned feature set had an accuracy of 94.26% with only 4 false-positives on the legitimate user's comments. Of the 100 intrusive comments 41 out of 59 were captured in the 10-fold CV tests. The F-measures were 0.969 for legitimate comments, and 0.566 for intrusive comments. The NaiveBayes Multinomial performed well on the reduced feature set, increasing its accuracy to 96.72% and had 0 misclassifications of the legitimate data. Of the intrusive data set the classifier predicted 64 of the 100 entries correctly. The F-score was

0.982 for legitimate data, and 0.780 for intrusive data. This classifier, feature selection combination had the best results of all the methods tried by me during the project.

Table 5.4: Summary of Results: Dataset II.

Notable Methods	Accuracy	F-Measure (Legitimate)	F-Measure (Intrusive)
Voted Perceptron	94.26%	0.969	0.566
NaiveBayes Multinomial	96.72%	0.982	0.780

Table 5.5: Comparing our Method to NB on Dataset II.

Notable Methods	Confusion Matrix	
Voted Perceptron	994	4
	59	41
NaiveBayes Multinomial	998	0
	36	64

Using the best subset feature selection method with a greedy stepwise search algorithm we obtained the third data set. This dataset performed in general better than no feature selection at all, but not as well as the statistical feature selection. WE'll present the NBM results to compare, as it performed in the top for the two preceding sets, but using logistic regression yielded the best set of results for this data set.

Table 5.6: Summary of Results: Dataset III.

Notable Methods	Accuracy	F-Measure (Legitimate)	F-Measure (Intrusive)
Logistic Regression	94.44%	0.970	0.573
NaiveBayes Multinomial	93.90%	0.967	0.504

Table 5.7: Comparing our Method to NB on Dataset III.

Notable Methods	Confusion Matrix	
Logistic Regression	996	2
	59	41
NaiveBayes Multinomial	997	1
	66	34

For the fourth data set WE wanted to try not just word tokens but the ngram models we discussed in class for NLP. WE was able to use a build in MATLAB ML LIBRARY filter to split the data into features of unigram to trigram values. This resulted in many hundreds of thousands of values which WE was able to use statistical feature selection on to yield the final dataset. There is no ngrams dataset with the subset evaluation feature selection method because the program was taking too long to find the optimal subset of over one hundred thousand features. From this data set the NaiveBayes Multinomial performed similar to dataset II with the addition of 2 more of the 100 intrusive entries are caught, with no change in legitimate entries. The rest of the classifiers performed at or below their values for unigram models.

Table 5.8: Summary of Results: Dataset IV.

Notable Methods	Accuracy	F-Measure (Legitimate)	F-Measure (Intrusive)
NaiveBayes Multinomial	96.90%	0.983	0.795

Table 5.9: Comparing our Method to NB on Dataset IV.

Notable Methods	Confusion Matrix	
NaiveBayes Multinomial	998	0
	34	66

a. First Neural Network with one hidden layer was implemented.
b. Different values of lambda was used to check the performance.

Table 5.10: Lambda and Accuracy.

Lambda (λ)	0.01	0.1	1	10
Accuracy (%)	70.15	65.50	61.2	57.35

Bar plot:

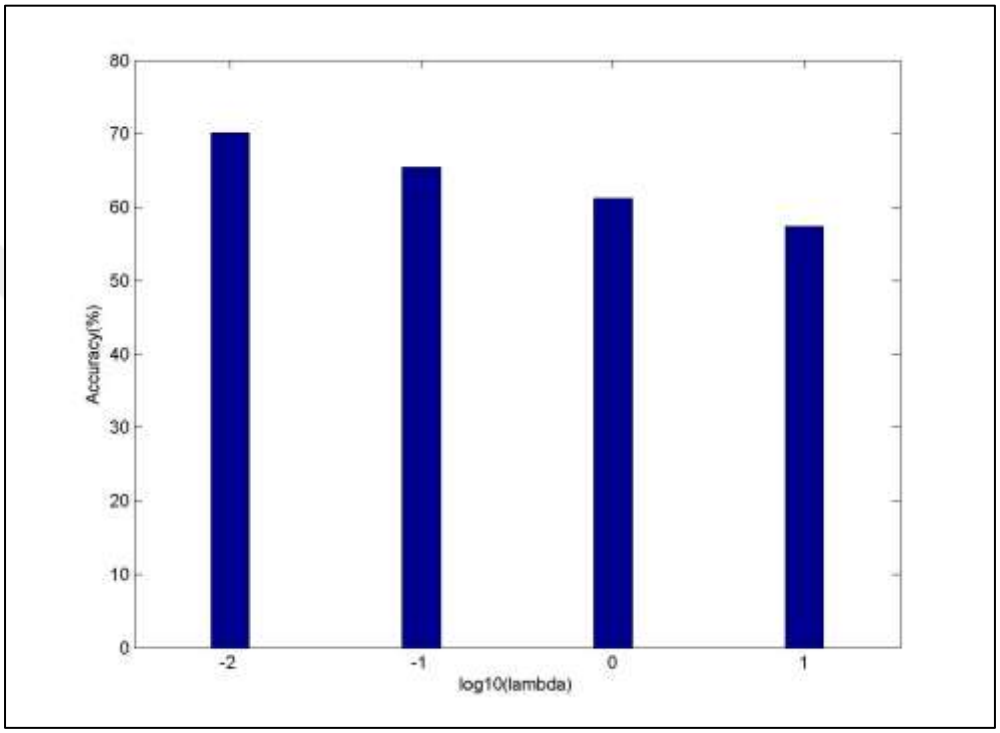

Figure 5.1: Accuracy Results.

a. For Neural Network with two hidden layers, results were pretty good.

b. Again, different values of lambda was used to compare the performance.

Table 5.11: Neural Network with two Hidden Layers.

Lambda (λ)	**0.01**	**0.1**	**1**	**10**
Accuracy (%)	**81.56**	**76.25**	**69.78**	**63.86**

Bar Plot:

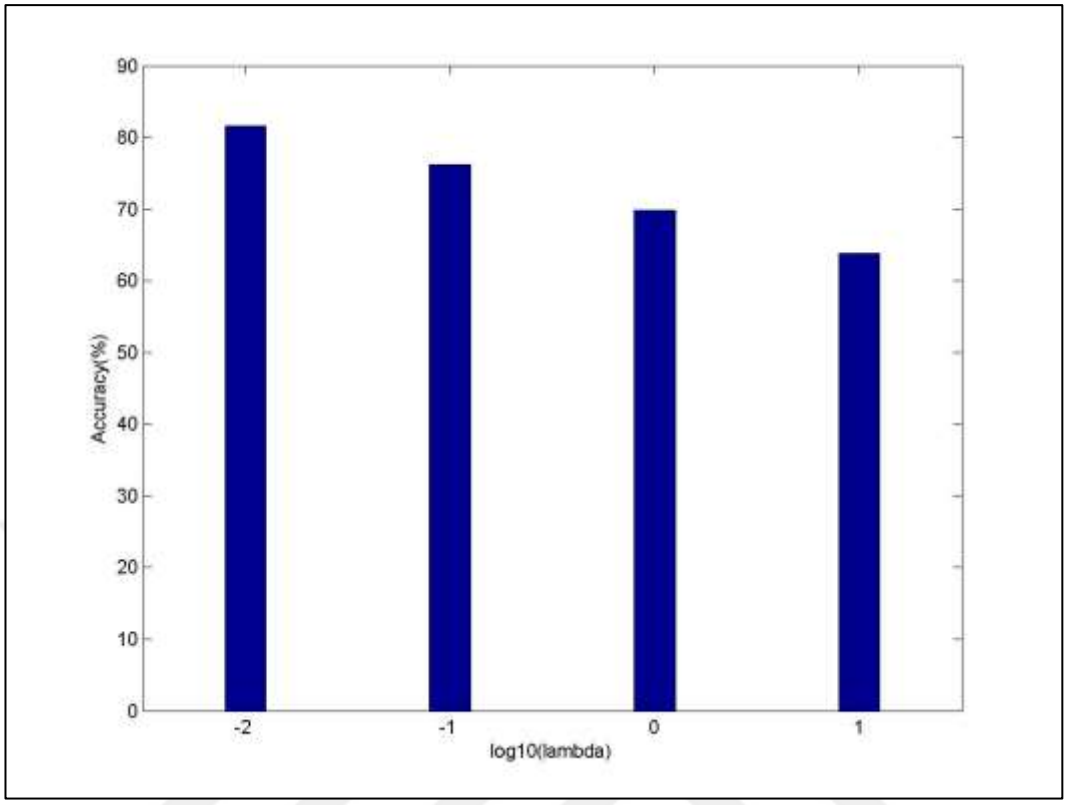

Figure 5.2: Neural Network With Two Hidden Layers.

6. CONCLUSION AND FUTURE WORK

6.1 CONCLUSIONS

By applying the NLP methods of stemming, stopword removal, and tokenization we were able to extract a set of features that allowed a machine learning program to detect intrusive behavior in the traffic submit by a user of a computer. In the best of all cases we used a NaiveBayes Multinomial classifier paired with a statistical feature selection algorithm to get a 96.9% accurate classifier, with no false positives and catching 66 of 100 intrusive messages. This system may not be powerful enough in isolation to detect intrusive activity on a network, but it may be a successful tool worth adding to the toolbox of a security professional trying to eliminate intruders and zombies in the organization he is protecting.

6.2 FURTHER WORK

These experiments were interested to run, and led to some decent results. The problems with this method are that the user and the intruder have no syntactical information extracted from the setting. It is viable to assume that an intruder will behave in a specific manner, and not just in a different manner. Therefore, there perhaps is much room for improvement of this system by including syntactical information from libraries such as WordNet and utilizing HMMs for language analysis. Also sentiment analysis and verb usage may play a telling role in revealing the intruder. The Intruder will likely have an agenda which might be revealed by these NLP methods. There might be better ways to combine the NLP and machine learning methods to build a better classifier. The Synset methods from WordNet appeared to be very useful in the NLP model, and perhaps incorporating that into a machine learning approach with large bags of binary features may outperform both of these techniques.

REFERENCES

[1] Tubaishat, M.; Madria, S. Sensor networks: An overview. *IEEE Potentials* 2003, *22*, 20–23. [CrossRef]

[2] Tolle, G.; Polastre, J.; Szewczyk, R.; Culler, D.; Turner, N.; Tu, K.; Burgess, S.; Dawson, T.; Buonadonna, P.; Gay, D.; et al. A macroscope in the redwoods. In Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, San Diego, CA, USA, 2–4 November 2005.

[3] Selavo, L.; Wood, A.; Cao, Q.; Sookoor, T.; Liu, H.; Srinivasan, A.; Wu, Y.; Kang, W.; Stankovic, J.; Young, D.; et al. LUSTER: Wireless sensor network for environmental research. In Proceedings of the 5th International Conference on Embedded Networked Sensor Systems, Sydney, Australia, 6–9 November 2007.

[4] Werner-Allen, G.; Lorincz, K.; Johnson, J.; Lees, J.; Welsh, M. Fidelity and yield in a volcano monitoring sensor network. In Proceedings of the 7th Symposium on Operating Systems Design and Implementation, Seattle, WA, USA, 6–8 November 2006.

[5] Kim, Y.; Schmid, T.; Charbiwala, Z.M.; Friedman, J.; Srivastava, M.B. NAWMS: Nonintrusive autonomous water monitoring system. In Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, Raleigh, NC, USA, 5–7 November 2008.

[6] Pantazis, N.A.; Vergados, D.D. A survey on power control issues in wireless sensor networks. IEEE Commun. Surv. Tutor. 2007, 9, 86–107. [CrossRef]

[7] Rajagopalan, R.; Varshney, P.K. Data-aggregation techniques in sensor networks: A survey. IEEE Commun. Surv. Tutor. 2006, 8, 48–63. [CrossRef]

[8] Murad, A.R.; Maarof, M.A.; Anazida, Z. A Survey of Intrusion Detection Schemes in Wireless Sensor Networks. Am. J. Appl. Sci. 2012, 9, 1636–1652.

[9] Almomani, I.; Al-Kasasbeh, B.; Al-Akhras, M. WSN-DS: A Dataset for Intrusion Detection Systems in Wireless Sensor Networks. J. Sens. 2016, 2016, 4731953. [CrossRef]

[10] Peddabachigari, S.; Abraham, A.; Grosan, C.; Thomas, J. Modeling intrusion detection system using hybrid intelligent systems. J. Netw. Comput. Appl. 2007, 30, 114–132. [CrossRef]

[11] Butun, I.; Morgera, S.D.; Sankar, R. A Survey of Intrusion Detection Systems in Wireless Sensor Networks. IEEE Commun. Surv. Tutor. 2014, 16, 266–282. [CrossRef]

[12] Modares, H.; Salleh, R.; Moravejosharieh, A. Overview of Security Issues in Wireless Sensor Networks. In Proceedings of the Third International Conference on Computational Intelligence, Modelling & Simulation, Langkawi, Malaysia, 20–22 September 2011; pp. 308–311.

[13] Farooq, N.; Zahoor, I.; Mandal, S.; Gulzar, T. Systematic analysis of DoS attacks in wireless sensor networks with wormhole injection. Int. J. Inf. Comput. Technol. 2014, 4, 173–182.

[14] Hubballi, N.; Suryanarayanan, V. False alarm minimization techniques in signature-based intrusion detection systems: A survey. Comput. Commun. 2014, 49, 1–17. [CrossRef]

[15] Ni, X.; He, D.; Ahmad, F. Practical network anomaly detection using data mining techniques. VFAST Trans. Softw. Eng. 2016, 4, 21–26. [CrossRef]

[16] Yu, Y.; Long, J.; Liu, F.; Cai, Z. Machine learning combining with visualization for intrusion Detection: A survey. In Proceedings of the 13th International Conference on Modeling Decisions for Artificial Intelligence, Sant Julià de Lòria, Andorra, 19–21 September, 2016; Springer: Cham, Switzerland; pp. 239–249.

[17] Fernandes, G.; Carvalho, L.F.; Rodrigues, J.J.P.C.; Proença, M.L. Network anomaly detection using IP flows with Principal Component Analysis and Ant Colony Optimization. J. Netw. Comput. Appl. 2016, 64, 1–11. [CrossRef]

[18] Bamakan, S.M.H.; Wang, H.; Yingjie, T.; Shi, Y. An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization. Neurocomputing 2016, 199, 90–102. [CrossRef]

[19] Li, S.; Chen, H.; Wang, M.; Heidari, A.A.; Mirjalili, S. Slime mould algorithm: A new method for stochastic optimization. Future Gener. Comput. Syst. 2020, 111, 300–323. [CrossRef]

[20] Abraham, A.; Jain, R.; Thomas, J.; Han, S.Y. D-SCIDS: Distributed soft computing intrusion detection system. J. Netw. Comput. Appl. 2007, 30, 81–98. [CrossRef]

[21] Khammassi, C.; Krichen, S. A GA-LR wrapper approach for feature selection in network intrusion detection. Comput. Secur. 2017, 70, 255–277. [CrossRef]

[22] Ferriyan, A.; Thamrin, A.H.; Takeda, K.; Murai, J. Feature selection using genetic algorithm to improve classification in network intrusion detection system. In Proceedings of the International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC), Surabaya, Indonesia, 26–27 September 2017; pp. 46–49.

[23] Li, J.; Zhao, Z.; Li, R.; Zhang, H. AI-Based Two-Stage Intrusion Detection for Software Defined IoT Networks. IEEE Internet Things J. 2019, 6, 2093–2102. [CrossRef]

[24] Moustafa, N.; Slay, J.; Creech, G. Novel Geometric Area Analysis Technique for Anomaly Detection Using Trapezoidal Area Estimation on Large-Scale Networks. IEEE Trans. Big Data 2019, 5, 481–494. [CrossRef]

[25] Zhang, Z.; Xie, L. A many-objective integrated evolutionary algorithm for feature selection in anomaly detection. Concurr. Comput. Pract. Exp. 2020, 32, e5861. [CrossRef]

[26] Mhawi, D.N.; Aldallal, A.; Hassan, S. Advanced Feature-Selection-Based Hybrid Ensemble Learning Algorithms for Network Intrusion Detection Systems. Symmetry 2022, 14, 1461. [CrossRef]

[27] Han, H.; Kim, H.; Kim, Y. An Efficient Hyperparameter Control Method for a Network Intrusion Detection System Based on Proximal Policy Optimization. Symmetry 2022, 14, 161. [CrossRef]

[28] Saghezchi, F.B.; Mantas, G.; Violas, M.A.; de Oliveira Duarte, A.M.; Rodriguez, J. Machine Learning for DDoS Attack Detection in Industry 4.0 CPPSs. Electronics 2022, 11, 602. [CrossRef]

[29] Jaw, E.; Wang, X. Feature Selection and Ensemble-Based Intrusion Detection System: An Efficient and Comprehensive Approach. Symmetry 2021, 13, 1764. [CrossRef]

[30] Adnan, A.; Muhammed, A.; Abd Ghani, A.A.; Abdullah, A.; Hakim, F. An Intrusion Detection System for the Internet of Things Based on Machine Learning: Review and Challenges. Symmetry 2021, 13, 1011. [CrossRef]

[31] Alabdulwahab, S.; Moon, B. Feature Selection Methods Simultaneously Improve the Detection Accuracy and Model Building Time of Machine Learning Classifiers. Symmetry 2020, 12, 1424. [CrossRef]

[32] Abdulhammed, R.; Musafer, H.; Alessa, A.; Faezipour, M.; Abuzneid, A. Features Dimensionality Reduction Approaches for Machine Learning Based Network Intrusion Detection. Electronics 2019, 8, 322. [CrossRef]

[33] Qureshi, A.-U.-H.; Larijani, H.; Mtetwa, N.; Javed, A.; Ahmad, J. RNN-ABC: A New Swarm Optimization Based Technique for Anomaly Detection. Computers 2019, 8, 59. [CrossRef]

[34] Maheswaran, N.; Bose, S.; Logeswari, G.; Anitha, T. Multistage intrusion detection system using machine learning algorithm. In Mobile Computing and Sustainable Informatics; Springer: Singapore, 2022; pp. 139–153.

[35] Fu, Y.; Du, Y.; Cao, Z.; Li, Q.; Xiang, W. A Deep Learning Model for Network Intrusion Detection with Imbalanced Data. Electronics 2022, 11, 898. [CrossRef]

[36] Imrana, Y.; Xiang, Y.; Ali, L.; Abdul-Rauf, Z.; Hu, Y.-C.; Kadry, S.; Lim, S. χ2-BidLSTM: A Feature Driven Intrusion Detection System Based on χ2 Statistical Model and Bidirectional LSTM. Sensors 2022, 22, 2018. [CrossRef]

[37] Li, J.; Zhao, Z.; Li, R. Machine learning-based IDS for software-defined 5G network. IET Netw. 2018, 7, 53–60. [CrossRef] 38. ISCX. NSL-KDD: Information security Centre of Excellence (ISCX), Univ. New Brunswick. 2015. Available online: http: //www.unb.ca/cic/research/datasets/nsl.html (accessed on 5 April 2022).

[38] Canadian Institute of Cybersecurity. CSE-CIC-IDS2018; Canadian Institute of Cybersecurity: Fredericton, NB, Canada, 2018.

[39] Hettich, S. KDD Cup 1999 Data; The UCI KDD Archive; University of California: Irvine, CA, USA, 1999.

[40] Houssein, E.H.; Mahdy, M.A.; Blondin, M.J.; Shebl, D.; Mohamed, W.M. Hybrid slime mould algorithm with adaptive guided differential evolution algorithm for combinatorial and global optimization problems. Expert Syst. Appl. 2021, 174, 114689. [CrossRef]

[41] Ni, X.; He, D.; Ahmad, F. Practical network anomaly detection using data mining techniques. VFAST Trans. Softw. Eng. 2016, 4, 21–26. [CrossRef]

[42] Yu, Y.; Long, J.; Liu, F.; Cai, Z. Machine learning combining with visualization for intrusion Detection: A survey. In Proceedings of the 13th International Conference on Modeling Decisions for Artificial Intelligence, Sant Julià de Lòria, Andorra, 19–21 September, 2016; Springer: Cham, Switzerland; pp. 239–249.

[43] Fernandes, G.; Carvalho, L.F.; Rodrigues, J.J.P.C.; Proença, M.L. Network anomaly detection using IP flows with Principal Component Analysis and Ant Colony Optimization. J. Netw. Comput. Appl. 2016, 64, 1–11. [CrossRef]

[44] Bamakan, S.M.H.; Wang, H.; Yingjie, T.; Shi, Y. An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization. Neurocomputing 2016, 199, 90–102. [CrossRef]

[45] Li, S.; Chen, H.; Wang, M.; Heidari, A.A.; Mirjalili, S. Slime mould algorithm: A new method for stochastic optimization. Future Gener. Comput. Syst. 2020, 111, 300–323. [CrossRef]

[46] Abraham, A.; Jain, R.; Thomas, J.; Han, S.Y. D-SCIDS: Distributed soft computing intrusion detection system. J. Netw. Comput. Appl. 2007, 30, 81–98. [CrossRef]

[47] Khammassi, C.; Krichen, S. A GA-LR wrapper approach for feature selection in network intrusion detection. Comput. Secur. 2017, 70, 255–277. [CrossRef]

[48] Ferriyan, A.; Thamrin, A.H.; Takeda, K.; Murai, J. Feature selection using genetic algorithm to improve classification in network intrusion detection system. In Proceedings of the International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC), Surabaya, Indonesia, 26–27 September 2017; pp. 46–49.

[49] Li, J.; Zhao, Z.; Li, R.; Zhang, H. AI-Based Two-Stage Intrusion Detection for Software Defined IoT Networks. IEEE Internet Things J. 2019, 6, 2093–2102. [CrossRef]

[50] Moustafa, N.; Slay, J.; Creech, G. Novel Geometric Area Analysis Technique for Anomaly Detection Using Trapezoidal Area Estimation on Large-Scale Networks. IEEE Trans. Big Data 2019, 5, 481–494. [CrossRef]

[51] Zhang, Z.; Xie, L. A many-objective integrated evolutionary algorithm for feature selection in anomaly detection. Concurr. Comput. Pract. Exp. 2020, 32, e5861. [CrossRef]

[52] L. Chao, S. Wen, and C. Fong, "CANN: an intrusion detection system based on combining cluster centers and nearest neighbors," Knowl-Based Syst, vol. 78, pp. 13–21, 4// 2015

[53] S. Chebrolu, A. Abraham, and J. P. Thomas, "Feature deduction and ensemble design of intrusion detection systems," Computers & Security, vol. 24, no. 4, pp. 295–307, 6// 2005

[54] W.-H. Chen, S.-H. Hsu, and H.-P. Shen, "Application of SVM and ANN for intrusion detection," Comput Oper Res, vol. 32, no. 10, pp. 2617–2634,2005/10/01/ 2005

[55] M. Cova, C. Kruegel, and G. Vigna, "Detection and analysis of drive-by-download attacks and malicious JavaScript code," Presented at the Proceedings of the 19th international conference on world wide web, Raleigh, North Carolina, USA, 2010

[56] C. Cowan et al., "Stackguard: automatic adaptive detection and prevention of buffer-overflow attacks," in USENIX security symposium, 1998, vol. 98, pp. 63–78: San Antonio, TX

[57] G. Creech, "Developing a high-accuracy cross platform host-based intrusion detection system capable of reliably detecting zero-day attacks," University of New South Wales, Canberra, Australia, 2014

[58] Creech G, Hu J (2014a) A semantic approach to host-based intrusion detection systems using Contiguousand Discontiguous system call patterns. IEEE Trans Comput 63(4):807–819

[59] Creech G, Hu J (2014b) A semantic approach to host-based intrusion detection systems using contiguous and Discontiguous system call patterns. IEEE Trans Comput 63(4):807–819

[60] Das, J. Bonneau, M. Caesar, N. Borisov, and X. Wang, "The tangled web of password reuse," in NDSS, 2014, vol. 14, pp. 23–26